flexera

AdminStudio 2022 R2

User Guide

Legal Information

Book Name: AdminStudio 2022 R2 User Guide

Part Number: ADS-2022 R2-UG00

Product Release Date: November 2022

Copyright Notice

Copyright © 2022 Flexera

This publication contains proprietary and confidential information and creative works owned by Flexera and its licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such publication in whole or in part in any form or by any means without the prior express written permission of Flexera is strictly prohibited. Except where expressly provided by Flexera in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera, must display this notice of copyright and ownership in full.

Intellectual Property

For a list of trademarks and patents that are owned by Flexera, see https://www.flexera.com/producer/company/about/intellectual-property/. All other brand and product names mentioned in Flexera products, product documentation, and marketing materials are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The Software is commercial computer software. If the user or licensee of the Software is an agency, department, or other entity of the United States Government, the use, duplication, reproduction, release, modification, disclosure, or transfer of the Software, or any related documentation of any kind, including technical data and manuals, is restricted by a license agreement or by the terms of this Agreement in accordance with Federal Acquisition Regulation 12.212 for civilian purposes and Defense Federal Acquisition Regulation Supplement 227.7202 for military purposes. The Software was developed fully at private expense. All other use is prohibited.

Third-Party Licenses

AdminStudio uses and redistributes the following third-party code:

Software	Links	
NCrawler	Website	http://ncrawler.codeplex.com
	License	http://ncrawler.codeplex.com/license
	Attribution	http://ncrawler.codeplex.com/team/view
	Download Link	http://saturn.installshield.com/product/adminstudio/115/NCrawler/NCrawler.zip
HTML Agility Pack	Website	http://htmlagilitypack.codeplex.com
	License	http://htmlagilitypack.codeplex.com/license
	Attribution	http://htmlagilitypack.codeplex.com/team/view

Contents

1 AdminStudio 2022 R2 Help Library 61

What's New in AdminStudio 2022 R2 | 24.0 63

AdminStudio Editions and Components 67

AdminStudio Full Editions 67

AdminStudio Limited Editions 76

About AdminStudio ZENworks Edition 76

How to Upgrade AdminStudio Limited Edition to Professional or Enterprise Editions 76

Activating AdminStudio 77

Deactivating AdminStudio to Enable Activation on a Different Machine 78

Evaluating AdminStudio 79

Upgrading Your Product Edition 81

Using Help 81

Contacting Us 82

2 Product Activation for AdminStudio 83

Licensing Options 84

Overview of Subscription License Life Cycle 85

Evaluating AdminStudio Before Activating It 89

Purchasing an AdminStudio License 89

Activating Through the Internet 90

Activating Through a Web Page 90

Activating the Package Feed Module 91

Subscription Expiration and Renewal 91

Automatic Activation on Product Upgrade 98

Uninstalling and Reinstalling AdminStudio 100

Returning a License to Your Account on the Activation Server 100

Specifying the License Server for Concurrent Licenses 101

Troubleshooting Activation Issues **102**
Activation Errors **103**
Activation FAQs **104**

3 Getting Started with AdminStudio **109**
Getting Started Tab **110**
Test for Application Compatibility Tab **110**
Import Packages and Mobile Apps 111
Select Tests to Run and Set Default Fix Option. 111
Perform Testing and View Results 112
Migrate to Application Virtualization Tab **112**
Identify Packages to Virtualize 113
Convert to Virtual Formats 113
Test and Distribute Converted Packages 114
Migrate to Windows Installer Tab **114**
Repackage Legacy Package 115
Import Into Application Catalog. 115
Test Repackaged Applications and Resolve Issues 115
Distribute Repackaged Applications 116
Set Up Infrastructure Tab **116**
Create/Connect to an Application Catalog 116
Configure Virtual Machines 117
Set E-Mail Notification Settings 117
Enter Server/Database Connection Settings. 118

4 Using the AdminStudio Interface **119**
Configuring the AdminStudio Interface **120**
Launching AdminStudio Applications 120
Specifying the AdminStudio Shared Location 120
Setting E-Mail Notification Settings 121
Setting the Workflow Task Help Page Location 122
Configuring How Often AdminStudio Checks for Updates. 123
Configuring AdminStudio to Stay on Top 123
Generating a Debug Log for AdminStudio 123
Workflows and Projects **124**
Creating and Editing Workflows. 124
Creating New Workflows. 125
Renaming Workflows. 125
Filtering Workflows 125
Deleting Workflows 126
Creating New Tasks 126
Modifying Task Properties 126
Creating Notes for a Task 127
Renaming Tasks 127
Reordering Tasks 127

Associating Help Files with Tasks . . . 128
Deleting Tasks . . . 128
Adding New Tools from the Process Template Editor . . . 128
Creating and Using Projects . . . 129
Creating Projects with the New Workflow Project Wizard . . . 129
Filtering Projects . . . 130
Executing Projects . . . 130
Running Associated Tools in Projects . . . 130
Deleting Projects . . . 130
Saving Workflow and Project Changes . . . 131
Workflow Project Example: Using the New Workflow Project Wizard . . . 131
Workflows, Projects, and Permissions . . . 134
Frequently Asked Questions . . . 134
AdminStudio Interface Reference . . . 135
AdminStudio Start Page . . . 136
Process Assistants Tab . . . 138
Process Template Editor . . . 138
Task Properties View . . . 139
AdminStudio Menus and Toolbar . . . 139
Dialog Boxes . . . 141
About Dialog Box . . . 141
Add New Tool Dialog Box . . . 142
Command Line Properties Dialog Box . . . 143
Options Dialog Box . . . 144
Locations Tab . . . 144
Updates Tab . . . 145
Quality Tab: Customer Experience Improvement Program . . . 145
Notification Settings Tab . . . 146
Tool Properties Dialog Box . . . 147
Properties Tab . . . 147
Configuration Tab . . . 148
Wizards . . . 150
Add Tool Wizard . . . 150
Welcome Panel . . . 150
Tool Properties Panel . . . 150
Command-Line Configurations Panel . . . 151
New Workflow Project Wizard . . . 151
Welcome Panel . . . 151
Workflow Selection Panel . . . 151
Source Package Panel . . . 152
Target Directory and File Name Panel . . . 152
Log Files . . . 152

5 Managing Accounts and Directory Services . . . 153
Managing Accounts . . . 153
Sample Workflow Manager Users . . . 154

Filtering by Account Status . . . 155
Creating a New Account . . . 155
Importing Directory Services Accounts and Groups . . . 156
Viewing or Changing an Existing Account . . . 158
Disabling an Account . . . 159
Deleting an Account . . . 159
Managing Directory Services Configuration . . . 160
Managing Directory Services Connections . . . 160
Creating a New Directory Service Connection . . . 160
Viewing or Changing an Existing Directory Service Connection . . . 162
Deleting an Existing Directory Service Connection . . . 162
Managing Directory Services Attributes . . . 163
Setting Up a New Directory Service Attribute . . . 163
Deleting an Existing Directory Service Attribute . . . 165
Managing Account Logins . . . 165
Login Methods . . . 165
Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On . . . 167
Using Account Login . . . 168
Using Domain Account Login . . . 168
Using Single Sign-On Login . . . 169
Using Guest Account Login . . . 170
Setting Up a Guest Account . . . 170
Logging in as a Guest . . . 171
Accounts and Directory Services Reference . . . 172
Account Administration Page . . . 173
Account Details Page . . . 174
Directory Services Import Page . . . 176
Directory Services Administration Page . . . 178
Add Directory Service Connection Page . . . 179
Directory Services Attributes Administration Page . . . 183
Add Directory Service Attributes Page . . . 184

6 Managing Roles and Permissions . . . 187

AdminStudio and Workflow Manager Roles and Permissions . . . 187
Role Permission Lists . . . 187
Administration and Reports Permissions . . . 188
Workflow Manager Permissions . . . 190
AdminStudio Client Tools Permissions . . . 193
System Roles . . . 195
Super User Role: AMSSuper . . . 196
Default System Roles . . . 196
Default System Accounts . . . 197
Role Management . . . 197
Creating a New Role . . . 198
Viewing or Changing an Existing Role . . . 198

Copying an Existing Role . . . 199
Deleting a Role . . . 200
Roles Reference . . . 201
Role Administration Page . . . 201
Role Copy Page . . . 202
Role Details Page . . . 203

7 Managing Applications and Application Catalog Databases . . . 207
About the AdminStudio Host Process . . . 210
Managing Application Catalogs . . . 210
About AdminStudio Application Catalogs . . . 211
Application Catalog Organization and Structure . . . 211
Overview of Application Catalogs . . . 213
Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog . . . 213
Required Permissions on Application Catalog Databases . . . 214
About the Application Manager Ribbon Interface . . . 216
Connecting to an Application Catalog for the First Time . . . 219
Connecting to an Existing Application Catalog . . . 220
Connecting AdminStudio Client Tools to a Standalone Application Catalog . . . 220
Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog . . . 221
Login Troubleshooting: Error 0x800A1518 . . . 222
Creating New Application Catalogs . . . 224
Creating New Application Catalogs Using the AdminStudio Interface . . . 224
Creating New Application Catalogs Using Scripts . . . 226
Upgrading an Existing Application Catalog . . . 229
Specifying a Default AdminStudio Application Catalog . . . 230
Creating Multiple Named Connections to Distribution Systems . . . 233
Creating a New Distribution System Connection Setting . . . 236
Editing an Existing Distribution System Connection Setting . . . 240
Setting Up a Distribution Connection for Microsoft Intune . . . 241
Integrating with Other Flexera Applications via the Flexera Service Gateway . . . 245
Overview of Unified Application Management Workflow . . . 245
Enabling Communication Between Products . . . 247
Setting Up AdminStudio Accounts . . . 248
Synchronizing Applications with App Portal and FlexNet Manager Suite . . . 249
Flexera Service Gateway Messages . . . 250
App Portal Only Integration . . . 250
Managing an Application's Flexera Identifier . . . 251
Searching an Application Catalog for Unrecognized Applications . . . 252
Performing a Manual Search for a Flexera Identifier . . . 254
Creating Local Flexera Identifier Entries for Internal or Repackaged Applications . . . 256
Entering Microsoft ACT Database Connection Settings . . . 259
Searching an Application Catalog . . . 259
Package Customization . . . 261
Disconnecting from an Application Catalog . . . 261
Organizing Your Application Catalog Using Groups . . . 261

Adding Groups . . . 262
Organizing Applications in Application Manager . . . 262
Deleting Application Manager Groups . . . 262
Editing Group Properties . . . 263
Copying and Sharing Packages in the Application Catalog . . . 263
Moving Applications, OS Snapshots, and Groups . . . 264
Deleting Packages and Applications . . . 264
Browsing to Package Location from Application Manager Tree . . . 265
Importing . . . 265
Package Types Supported By the Import Wizard . . . 266
Package Sources Supported by the Import Wizard . . . 271
Importing an Application Using The Package Feed Module . . . 271
Network Requirements for Package Feed Module . . . 283
Importing a Single Package File . . . 284
Associating a Virtual Package with its Source Windows Installer Package . . . 287
About Windows Installer Packages (.msi) . . . 289
About Microsoft MSIX Packages (.msix) . . . 289
About Transforms (.mst) . . . 289
About Patches (.msp) . . . 289
About Legacy Installer Packages . . . 290
Importing Links to Public Store Applications . . . 290
Importing a Folder of Multiple Applications . . . 296
Importing From ConfigMgr (Formerly called as System Center Configuration Manager) . . . 299
Importing Applications, Mobile Apps, and Packages from ConfigMgr (Formerly called as System Center Configuration Manager) 300
Package Information Imported from ConfigMgr (Formerly called as System Center Configuration Manager) . . . 303
Importing Merge Modules . . . 304
About Merge Modules (.msm) . . . 305
Importing OS Snapshots . . . 305
About OS Snapshots (.osc) . . . 306
Using Duplicate Package Identifiers . . . 306
Generating Software ID Tag Files During Package Import . . . 308
About Software ID Tag File Generation . . . 309
How Tag Files Are Named . . . 309
Output Files Created by Tag File Generation: .mst and .cab . . . 310
Sample Software ID Tag File . . . 310
Creation of Tag Files During Application Catalog Upgrade . . . 312
Support for Packages With Multiple Tag Files . . . 312
How Existing Tag Information is Incorporated Into the Software ID Tag File . . . 313
About Software Tagging RegIDs . . . 314
About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields . . . 315
Viewing and Editing Package Tag Information in Application Manager . . . 316
Viewing Bundled Packages of Complex Installer Executables . . . 317
Using Package Automation . . . 318
Adding a Single Package Request to the Backlog Tab . . . 318
Editing a Package Request in the Backlog Tab . . . 320

Customize Package . . . 321
Importing a List of Applications for Automation . . . 321
Matching the Applications . . . 325
Configuring Actions for Automation . . . 330
Configure Import Action for Automation . . . 330
Configure Test Action for Automation . . . 332
Configure Wrap Action for Automation . . . 333
Configure Convert Action for Automation . . . 335
Configure Publish Action for Automation . . . 336
Scheduling Automation . . . 338
Scheduling Automation for Once . . . 338
Scheduling Daily Automation . . . 340
Scheduling Weekly Automation . . . 341
Scheduling Monthly Automation . . . 343
Specifying Notification Settings for Automation . . . 345
Automate Package Feed Module Trial Packages . . . 347
Support for Publishing to Microsoft Intune and Converting to Intune Package Format . . . 348
Publishing to a Microsoft Intune Distribution System . . . 349
Setting up Intune Connection for the Distribution System . . . 349
Setting Up the Distribution Properties . . . 352
Publishing the Applications . . . 353
Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard . . . 354
Monitored Directory for Package Automation . . . 357
Identifying Vendor Command Line Arguments . . . 360
Viewing Application Testing and Analysis Reports on the Reports Tab . . . 361
Available Reports . . . 362
Viewing a Report . . . 366
Exporting a Report in PDF, Excel, or Word Format . . . 367
Creating Custom Reports . . . 367
Managing ConfigMgr (Formerly called as System Center Configuration Manager) Application Model Data . . . 371
Specifying General Application Information . . . 372
Specifying ConfigMgr (Formerly called as System Center Configuration Manager) Information . . . 372
Specifying Deployment Data for an Application's Packages . . . 373
Specifying Intune Deployment Data for an Application's Packages . . . 374
Viewing Reference Data: Dependencies and Supersedences . . . 375
Viewing Microsoft Configuration Manager Deployment Information . . . 375
Viewing an Application's Configuration Manager Deployment Status . . . 375
Viewing the Microsoft Configuration Manager Deployments Report . . . 376
Viewing Microsoft Intune Deployments Report . . . 377
Retiring or Reinstating an Application in System Center 2012 Configuration Manager . . . 378
Managing macOS Desktop Application Metadata . . . 379
Viewing Imported macOS Desktop Application Metadata . . . 379
Customizing Apple Installer Package PKG Installer Settings . . . 382
Viewing Bundled Packages of macOS PKG and DMG Files . . . 383
Managing Mobile App Metadata . . . 383

About Mobile App Metadata . . . 384
Viewing Imported Mobile App Metadata . . . 384
Specifying the Path to Local iOS and Android Public Store Apps . . . 386
iOS Property Files (Info.plist) and iOS Enterprise Policy Files (*.plist) . . . 387
Managing iOS Enterprise Policy Configuration Files . . . 388
About Enterprise Policy Configuration Files . . . 388
Importing Enterprise Policy Configuration Files . . . 389
Viewing Enterprise Policy Configuration File Settings . . . 390
Viewing Mobile App Reports . . . 391
Viewing Mobile App Analysis and Test Result Reports . . . 391
Viewing iOS Enterprise Policy Compatibility Reports . . . 397
Managing App Portal Application Information . . . 400
Enabling Automatic Creation of App Portal Catalog Item . . . 401
Setting Brief Description and Long Description . . . 403
Specifying Catalog Item Categories . . . 404
Selecting an App Portal Template . . . 406
Specifying Catalog Item Keywords . . . 407
Troubleshooting: App Portal Catalog Item Not Created Upon AdminStudio Publication . . . 409
Enabling Application Extended Attributes . . . 411
Enabling the Extended Attributes Tab of the Application View . . . 412
Defining Application Extended Attributes . . . 413
Viewing and Editing Application Extended Attributes . . . 417
Managing ConfigMgr (Formerly called as System Center Configuration Manager) Package Deployment Data . . . 418
Specifying Package Content Deployment Data . . . 420
Specifying Package Programs Deployment Data . . . 420
Specifying Package User Experience Deployment Data . . . 421
Specifying Package Detection Methods Deployment Data . . . 422
Viewing Detection Methods for a Windows Store Application or MSIX Package . . . 422
Viewing a Windows Store Application's Framework Customizations . . . 423
Specifying Package Requirements Deployment Data . . . 424
Creating Custom Requirements Containing Global Conditions . . . 424
Building Expressions When Creating Global Conditions . . . 424
Creating and Editing Global Conditions . . . 426
Creating Device Requirements . . . 427
Creating User Requirements . . . 429
Specifying Package Dependencies Deployment Data . . . 431
Specifying Package Supersedences Deployment Data . . . 433
Viewing and Editing Return Codes . . . 434
Changing the Priority of Deployment Types . . . 435
Setting Application Model Properties . . . 436
Setting Default Application Model Properties . . . 437
Setting Default Application Model Properties on the Options Dialog Box . . . 437
Setting Default Application Model Properties Using an SQL Script . . . 438
Setting Application Model Properties Using the AdminStudio PowerShell Cmdlets . . . 441
Managing App-V Package Deployment Data . . . 441
Specifying a Package's App-V Deployment Settings . . . 442

Specifying a Package's Advanced App-V Deployment Settings . . . 442
Managing Casper Package Deployment Data . . . 443
Managing Citrix XenApp Package Deployment Data . . . 447
Specifying a Package's XenApp Deployment Settings . . . 448
Specifying a Package's Advanced XenApp Deployment Settings . . . 450
Managing Altiris Package Deployment Data . . . 451
Specifying a Package's Altiris Deployment Settings . . . 452
Specifying a Package's Altiris Deployment Command Line Settings . . . 453
Managing Workspace ONE Package Deployment Data . . . 454
Specifying Package Details . . . 454
Specifying Package Files . . . 455
Specifying Package Deployment Options . . . 455
Managing Microsoft Intune Deployment Data . . . 456
Specifying App Information . . . 457
Specifying Package Programs . . . 457
Specifying Package Requirements . . . 458
Specifying Package Detection Rules . . . 459
Specifying Return Codes . . . 459
Managing App-V Virtual Environments . . . 460
Creating an App-V Server Virtual Environment . . . 461
Editing an Existing App-V Server Virtual Environment . . . 465
Creating a ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment . . . 465
Editing an Existing ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment . . . 470
Viewing a Package's ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environments . . . 470
Viewing Additional Package Data . . . 471
Viewing and Editing Package Extended Attributes . . . 472
Using Package Extended Attributes . . . 472
Package Extended Attribute Description File . . . 473
Integrating Package Extended Attribute Data with a Workflow Request . . . 475
Viewing Package Dependencies . . . 476
Viewing Windows Installer Package Dependencies . . . 476
Viewing App-V Package Dependencies . . . 477
Viewing Package Files, Components, and Directories . . . 478
Viewing Windows Installer Package Files and Components . . . 478
Viewing App-V Package Files and Directories . . . 478
Viewing Windows Installer Package INI File Changes . . . 480
Viewing Registry Information . . . 480
Viewing Windows Installer Package or Microsoft UWP App Package Registry Information . . . 480
Viewing App-V Package Registry Information . . . 481
Viewing Package Shortcuts . . . 482
Viewing Windows Installer Package Shortcuts . . . 482
Viewing App-V Package Shortcuts . . . 483
Viewing Windows Installer Package Merge Modules . . . 485
Viewing Package Catalog History . . . 485
Viewing App-V Package History . . . 486

Viewing Package Tables . . . 488
Viewing App-V Package File Type Associations . . . 488
Viewing App-V Package Environment Variables . . . 489
Creating, Importing, and Managing PowerShell-Wrapped Packages . . . 490
Setting Wrap Options . . . 491
PowerShell Script Template Files . . . 492
Customizing the PowerShell Installer . . . 494
Customizing the PowerShell Installer Banner Image . . . 494
Customizing Post-Installation Text Messages . . . 495
Wrapping a Windows Installer (.msi) or Installer Executable (.exe) . . . 495
Importing PowerShell Wrapped Packages . . . 499
Viewing a Power Shell Wrapped Package's Bundled Child Applications . . . 500
Testing PowerShell Wrapped Packages . . . 500
PowerShell Wrapped Packages Report . . . 501
Distributing PowerShell Wrapped Packages . . . 501
Virtualizing PowerShell Wrapped Packages . . . 502
Testing PowerShell Wrapped Packages on a Virtual Machine . . . 502
Editing a PowerShell-Wrapped Package . . . 502
AdminStudio PowerShell Cmdlet Support for Performing PowerShell Wrapping: Invoke-ASPowerShellWrap . . . 503
Updating PowerShell App Deploy Toolkit . . . 504
Using the Conversion Wizard to Perform Virtualization or Repackaging . . . 505
Setting Conversion Wizard Options . . . 506
Creating an Automated Application Converter Settings File . . . 506
Specifying the Default Automated Application Converter Settings File . . . 510
Editing the Default Automated Application Converter Settings File From Application Catalog . . . 511
Setting App-V 5.0 Conversion Options . . . 513
Setting Microsoft Intune Conversion Options . . . 514
Converting App-V 4.x Packages to App-V 5.0 Format . . . 515
Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging . . . 518
Using Test on Virtual Machine Wizard . . . 522
Software Repository Integration into Other AdminStudio Tools . . . 525
Taking OS Snapshots . . . 527
OS Snapshot Best Practices . . . 527
Configuring OS Snapshot Analysis Options . . . 528
Capturing an OS Snapshot . . . 528
Reference . . . 529
Application Manager Interface . . . 529
Application Manager Ribbon Interface . . . 530
Application Catalog Tab Menu . . . 530
Home Tab of Application Manager Ribbon . . . 532
Analyze Tab of Application Manager Ribbon . . . 534
Reports Tab of Application Manager Ribbon . . . 535
Backlog Tab of Application Manager Ribbon . . . 536
Support Tab of Application Manager Ribbon . . . 537
Application Manager Tree and Subnode Icons . . . 538
Shortcut Menus . . . 543

Output Window . . . 548
Application Manager Reports Tab . . . 549
Products Tree/Home Tab Views . . . 550
Group View . . . 550
Application View . . . 551
General Information Tab . . . 551
References Tab . . . 555
Deployment Status Tab . . . 556
ConfigMgr Settings Tab . . . 559
App Portal Information Tab . . . 560
XenApp Deployment Types Tab . . . 563
Altiris Deployment Types Tab . . . 564
Workspace ONE Deployment Types Tab . . . 564
Extended Attributes Tab . . . 565
Home Deployment Type View . . . 566
Package Information Tab . . . 567
Deployment Data Tab . . . 573
Deployment Data Tab / Content Subtab . . . 574
Deployment Data Tab / Programs Subtab . . . 576
Deployment Data Tab / User Experience Subtab . . . 578
Deployment Data Tab / Detection Method Subtab . . . 580
Deployment Data Tab / Requirements Subtab . . . 581
Deployment Data Tab / Dependencies Subtab . . . 582
Deployment Data Tab / Supersedence Subtab . . . 583
Deployment Data Tab / Return Codes Subtab . . . 584
Deployment Data Tab / Detection Method AppX, MSIX Subtab . . . 585
Deployment Data Tab / Framework Subtab . . . 585
Deployment Data Tab / Virtual Environments Subtab . . . 586
Deployment Data Tab / Install Behavior Subtab . . . 587
Deployment Data Tab / Publishing Subtab . . . 587
Bundled Packages Tab . . . 588
PKG Installer Choices Tab . . . 590
App-V Deployment Data Tab . . . 591
Casper Deployment Data Tab . . . 594
XenApp Deployment Data Tab . . . 597
XenApp Deployment Data Tab / XenApp Information Subtab . . . 598
XenApp Deployment Data Tab / Advanced Settings Subtab . . . 600
Software Identification Tag Tab . . . 604
Altiris Deployment Data Tab . . . 607
Workspace ONE Deployment Data Tab . . . 609
Microsoft Intune Deployment Data Tab . . . 616
Home Deployment Type View Subnode Views . . . 622
Extended Attributes View (Packages) . . . 623
Dependencies View . . . 624
Files View . . . 626
INI File Changes View . . . 627

Registry View 627
Shortcuts View 629
Merge Modules View 631
Catalog History View 631
App-V History View 632
Tables View 633
File Type Associations View 633
Environment Variables View 634
Merge Module Tree Views 634
All Merge Modules View 634
Merge Module View 634
Components View 635
Dependency View 635
Exclusion View 635
Files View 636
Products View 636
Environment Tree Views 637
Security Patches Group View 637
New Security Patches Group View 638
Group View of a Selected Group 638
Patch View 639
OS Snapshot View 640
Enterprise Policy View 642
Dialog Boxes 643
Add Applications Dialog Box 644
Add Connection Group Packages Dialog Box 645
Add/Edit Applications Dialog Box 647
Add/Edit Return Code Dialog Box 648
AdminStudio Host Process 649
AdminStudio Tools Dialog Box 650
App-V Server Connection Groups Dialog Box 650
App-V Virtual Environments Dialog Box 652
Application Search Results Dialog Box 653
Associate with Workflow Manager Workflow Request Dialog Box 653
Categories Dialog Box 654
Change Deployment Type Priority Dialog Box 655
Change Enterprise Server Password Dialog Box 656
Configure Connection Group Dialog Box 657
Connect Application Catalog Dialog Box 658
Enterprise Server Tab 659
Standalone Tab / Specify Database Information 659
Recent Tab 661
Create Global Condition Dialog Box 661
Create Virtual Environment / Properties Dialog Box 668
Default Application Catalog Dialog Box 669
Edit Keywords Dialog Box 670

Extended Attribute Property Dialog Box . . . 671
Find Dialog Box . . . 671
Flexera Identifier Dialog Box . . . 673
Flexera Local Identifier Dialog Box . . . 676
Global Conditions Dialog Box . . . 677
Keywords Dialog Box . . . 679
Login Required Dialog Box . . . 679
Properties Dialog Box . . . 680
Options Dialog Box . . . 680
General Tab . . . *681*
Import Options / General Tab . . . *682*
Import Options / Duplicate Package Tab . . . *683*
Import Options / Application Model Defaults Tab . . . *685*
Import Options / Software Tagging Tab . . . *700*
Analyze Tab . . . *702*
Windows Installer Validation Tab . . . *703*
ACE Tests Tab . . . *704*
Mobile Tests Tab . . . *705*
Plugin Options Tab . . . *707*
Package Feed Options Tab . . . *712*
Wrap Options Tab . . . *714*
Server Options / Distribution System Tab . . . *715*
Server Options / Microsoft ACT Tab . . . *720*
Package Automation Options / Monitored Directory . . . *721*
Package Automation Options / Configure Actions . . . *722*
Package Automation Options / Schedule Automation . . . *724*
Package Automation Options / Notifications . . . *725*
Software Repository Tab . . . *725*
Flexera Integration / Flexera Service Gateway (FSG) Tab . . . *727*
AdminStudio Services via FSG Tab . . . *732*
References Dialog Box . . . 734
SCCM Server Environment Dialog Box . . . 734
Select Application Catalog Dialog Box . . . 735
Select AdminStudio Enterprise Server URL Dialog Box . . . 735
Select Substitute Package Dialog Box . . . 735
Servers Dialog Box . . . 736
Specify Applications Dialog Box . . . 736
Users Dialog Box . . . 738
Virtual Package Association Dialog Box . . . 738
XML Namespaces Dialog Box . . . 739
Wizards . . . 740
Application Catalog Wizard . . . 740
Welcome Panel . . . *740*
Specify Database Information Panel . . . *741*
Select Software Repository Location Panel . . . *743*
Creating Application Catalog Panel . . . *744*

Conversion Wizard 744
Target Type Selection Panel 745
Package Output Location Panel 747
Select the Package(s) to Convert Panel 747
Automated Application Converter Settings Panel 748
Summary Panel 749
Converting the Packages Panel 749
Customization Wizard 750
Options Panel 751
Homepage Preferences Panel 751
Distribution Preferences Panel 752
Add or Remove Programs Panel 753
Release Notes Panel 754
Saving Customization Panel 755
Detection Method Wizard 756
Welcome Panel 756
File System Detection Panel 758
Registry Detection Panel 759
Windows Installer Detection Panel 761
Script Detection Panel 762
Summary Panel 763
Detection Rule Wizard of Intune Deployment Data Tab 764
Welcome Panel 764
Configure Manual Detection Rule Panel 765
Custom Detection Script Panel 768
Summary Panel 770
Dependency Wizard 771
Welcome Panel 771
Deployment Types in Application Catalog Panel 772
Configuration Manager Credentials Panel 773
Deployment Types in Configuration Manager 2012 Panel 774
Auto Detect Dependencies Panel 775
Scanning Progress Panel 775
Auto Scan Results Panel 776
System Requirements Panel 778
Summary Panel 778
Import Apps List Wizard 779
Select Apps List Source 779
Import Apps List from CSV 780
Summary 782
Running the Import 783
Import Wizard 784
Source Panel 785
Search for an Application in Package Feed Panel 787
Provide Vendor Setup File Panel 788
Silent Command Line Arguments 790

Package Feed Module - Customization Wizard Panels *791*
Package Type Selection Panel (Single Application) *795*
Package Type Selection Panel (Folder of Multiple Applications) *798*
Enterprise Policy File Selection Panel *800*
Security Patch File Selection Panel *800*
OS Snapshot Selection Panel *801*
Public Store Selection Panel *802*
Store Application Selection Panel *802*
Source Server Details Panel *805*
Package File Selection Panel *806*
Package Folder Selection Panel *807*
Select Applications (Folder of Multiple Applications) Panel *808*
Select Applications/Packages Panel *809*
Package Support Files Panel *811*
Destination Group Panel *813*
Summary Panel *814*
Running the Import Panel *814*
Merge Module Import Wizard 815
MSM Source Information Panel *815*
Summary Panel *816*
OS Snapshot Wizard 816
Welcome Panel *816*
Project Information Panel *816*
Analyzing Panel *817*
OS Snapshot Summary Panel *817*
Analysis Options Dialog Box *817*
ISSnapshot.ini File *818*
Wrap Package Wizard 818
Requirement Wizard 820
Welcome Panel *821*
Create Custom Requirements Panel *822*
Create User Requirements Panel *825*
Select the Device Requirements Type Panel *826*
Configuration Manager Credentials Panel *826*
Device Requirements from Configuration Manager Panel *827*
Create Device Requirements Panel *829*
Summary Panel *831*
Requirement Wizard of Intune Deployment Data Tab 832
Welcome Panel *832*
Device Requirements Panel *833*
Additional Requirements Panel *835*
Summary Panel *837*
Return Code Wizard of Intune Deployment Data Tab 838
Welcome Panel *838*
Summary Panel *839*
Supersedence Wizard 840

Welcome Panel *841*
Deployment Types in Application Catalog Panel *842*
Configuration Manager Credentials Panel *843*
Deployment Types in Configuration Manager 2012 Panel *844*
Summary Panel *845*
Test on Virtual Machine Wizard 845
Automated Application Converter Test Settings Panel *846*
Summary Panel *847*
Performing the Test Process Panel *848*
Upgrade Wizard 849
User Permissions in Application Manager 850
Database Server Permissions 850

8 Repackaging Legacy Installations Using the Repackaging Wizard 853

About Repackaging 854
Purpose of Repackaging Applications 854
Supported Legacy Installation Types 855
Repackaging 64-Bit Applications 855
Repackaging Options Comparison 856
Repackaging Wizard Best Practices 857
About Repackaging on Clean Systems 859
Alternate-Language Repackaging on Clean Machines 860
Including the InstallScript Engine With a Windows Installer Package 860
Repackaging Methods 860
Installation Monitoring Method 860
Snapshot Method 861
Configuring Repackager to Ensure Optimal Installation Capture 862
Launching Repackager Remotely 862
Installing Repackager on a Clean Machine 864
Repackaging Legacy Installations Using the Repackaging Wizard 865
Repackaging Using the Installation Monitoring Method 866
Step 1: Selecting the Repackaging Method 867
Step 2: Excluding Processes (Optional) 868
Step 3: Collecting Product Information 869
Step 4: Adding Additional Setup Programs (Optional) 870
Step 5: Set Target Project Information 872
Step 6: Set Capture Settings (Optional) 873
Step 7: Beginning the Repackaging Process 874
Repackaging Using the Snapshot Method 876
Performing Multiple Step Snapshot Repackaging 877
Step 1: Selecting the Repackaging Method *878*
Step 2: Initial Analysis *879*
Step 3: Install Setup and Make Manual System Changes *881*
Step 4: Entering Product Information *881*
Step 5: Set Target Project Information *882*
Step 6: Set Capture Settings (Optional) *883*

Step 7: Beginning the Repackaging Process ... *885*
Performing Single Step Snapshot Repackaging ... 887
Step 1: Selecting the Repackaging Method ... *887*
Step 2: Collecting Product Information ... *889*
Step 3: Set Target Project Information ... *890*
Step 4: Set Capture Settings (Optional) ... *891*
Step 5: Beginning the Repackaging Process ... *893*
Repackaging an InstallScript MSI Setup to a Basic MSI Setup ... 895
Running the Repackaging Wizard from the Command Line ... 897
Repackaging a Windows Installer (.msi) Package ... 898
Documenting Repackaging Steps Using the Microsoft Step Recorder Tool ... 902
Repackaging Wizard Reference ... 907
Repackaging Wizard ... 908
Welcome Panel ... 908
Method Selection Panel ... 909
Snapshot Method Panel ... 910
Collect Product Information Panel ... 912
InstallScript MSI Identified Panel ... 913
Set Target Project Information and Capture Settings Panel ... 914
InstallScript MSI Conversion Output Panel ... 915
Repackaging Panel ... 915
Summary Panel ... 917
Additional Repackaging Wizard Dialog Boxes ... 918
Additional Setup Programs Dialog Box ... 918
Setup Information Dialog Box ... 919
Excluded Processes Dialog Box ... 919
Analysis Options Dialog Box ... 920
Repackaging Wizard Command-Line Options ... 922
Reboot Handling in the Repackaging Wizard ... 924

9 Converting Legacy Installations Using the Repackager Interface ... 927
About the Repackager Interface ... 928
Launching the Repackager Interface ... 928
Setting Repackager Options ... 929
Selecting Data Display Colors ... 929
Specifying Additional Merge Module Directories ... 929
Controlling the Display of ICE Validation Warnings ... 930
Suppressing Build Output Folder Overwrite Warnings ... 931
Creating Repackager Projects ... 932
Converting Legacy Installations Using the Repackager Interface ... 933
Converting Repackager 3.x Output Files ... 933
Converting a Microsoft SMS Project to a Repackager Project ... 934
Converting Microfocus ZENworks Projects ... 934
Converting a Microfocus ZENworks Project Using the Repackager Interface ... *934*
Converting Multiple Microfocus ZENworks Projects Using the Command Line ... *935*
Converting WinINSTALL Projects ... 937

Converting Wise Installation Projects . . . 937
Converting InstallShield Professional Log Files . . . 937
Working With Repackager Projects . . . 938
Building an InstallShield Editor Project . . . 938
Building a Windows Installer Package . . . 941
About the Context.msi File . . . 945
Configuring Advanced Conversion Options . . . 946
Automatically Generating a Virtual Application During Repackager Project Build . . . 947
Viewing Repackager Project Properties . . . 949
Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project . . . 950
Creating a Setup Capture Report for a Project . . . 951
Generating Software ID Tag Files During Repackaging . . . 952
Enabling Software ID Tag Generation During Repackaging . . . 953
Viewing and Editing Software ID Tag Information in the Repackager Interface . . . 955
Saving Repackager Projects . . . 955
Opening InstallShield Editor from Repackager . . . 956
Isolating Windows Installer Packages . . . 956
About Application Isolation . . . 957
Isolating Windows Installer Packages Using Application Isolation Wizard . . . 957
About Assemblies . . . 958
About Manifests . . . 958
About Digital Certificates . . . 959
Setting Isolation Options . . . 960
Specifying Manifest Options . . . 961
Selecting the Assembly Type . . . 961
Specifying the Assembly Naming Conventions . . . 961
Setting Digital Signature Options for Shared Assemblies . . . 962
Building an Isolated Windows Installer Package . . . 963
Configuring Exclusions . . . 963
Configuring Exclusions Using Repackager . . . 964
Excluding Files . . . 964
Excluding All Files in a Directory . . . 965
Excluding Directories and Subdirectories . . . 965
Adding Files and Folders to the Global Exclusions List from the Files and Folders View . . . 965
Excluding Registry Keys . . . 966
Excluding Registry Values . . . 966
Excluding .ini Files . . . 966
Excluding .ini File Sections . . . 967
Excluding Shortcuts . . . 967
Excluding All Shortcuts in a Directory . . . 967
Excluding Shortcuts from Subdirectories . . . 967
Specifying the External Configuration File . . . 968
Modifying External Configuration Files . . . 969
Configuring Exclusions Using the Exclusions Editor . . . 969
Exclusions and Repackager . . . 970
Exclusions and the OS Snapshot Wizard . . . 970

Launching Exclusions Editor . . . 970
Excluding Files . . . 971
Excluding Files with Specific Extensions . . . 972
Excluding Directories . . . 973
Editing Existing File Exclusions . . . 973
Removing File Exclusions . . . 973
Excluding .ini Files . . . 974
Excluding Sections from .ini Files . . . 974
Editing Existing .ini File Exclusions . . . 975
Removing .ini File Exclusions . . . 976
Excluding Registry Data . . . 976
Editing Existing Registry Exclusions . . . 977
Removing Registry Exclusions . . . 977
Repackaging and Anti-Virus Software . . . 978
Creating an InstallShield Editor Template to Use Within Repackager . . . 978
Repackager Interface Reference . . . 983
Repackager Interface . . . 984
Repackager Start Page . . . 985
Menus and Toolbar . . . 986
Dialog Boxes . . . 988
About Repackager Dialog Box . . . 988
Create Report Dialog Box . . . 989
Isolation Options Dialog Box . . . 990
Options Dialog Box . . . 992
Project Properties Dialog Box . . . 995
WinINSTALL Conversion Dialog Box . . . 998
Repackager Views . . . 998
Captured Installation View . . . 999
Files and Folders View . . . 1001
Registry Entries View . . . 1003
Shortcuts View . . . 1004
INI Files View . . . 1005
Deleted Files View . . . 1006
Deleted Registry Entries View . . . 1007
Repackaged Output View . . . 1008
Package Information View . . . 1012
Software Identification Tag View . . . 1013
Advanced Package Settings View . . . 1015
Setup Intent Wizard . . . 1017
Welcome Panel . . . 1018
Scanning Project Panel . . . 1018
Results Panel . . . 1018
VMware Repackaging Wizard . . . 1019
Welcome Panel . . . 1020
VMware Virtual Machines Panel . . . 1020
Exclusions Editor Interface . . . 1020

Menus . . . 1020
Files Tab . . . 1021
.ini Files Tab . . . 1022
Registry Tab . . . 1023
File Exclusion Information Dialog Box . . . 1024
INI File Exclusion Information Dialog Box . . . 1025
Choose Registry Key Dialog Box . . . 1025
Edit Registry Key Dialog Box . . . 1026
About Exclusions Editor Dialog Box . . . 1026
Options.ini File . . . 1027
Files Associated with Repackager . . . 1033
Repack.ini File . . . 1035
Using InstallShield to Chain Multiple Windows Installer Packages Together . . . 1036
Troubleshooting . . . 1036
Troubleshooting Guidelines for WinINSTALL Conversion . . . 1036
Troubleshooting Guidelines for SMS Conversion . . . 1037
Resolving an “Error Building Table File” Error . . . 1037

10 Performing Virtualization and Repackaging Using the Automated Application Converter 1039

Getting Started With Application Virtualization . . . 1041
About Application Virtualization . . . 1044
About Microsoft Application Virtualization . . . 1045
About Microsoft Application Virtualization (App-V) . . . 1046
Components of an App-V Package . . . 1046
Comparison of the App-V 5.0 Conversion Methods . . . 1048
Support for App-V 5.0 SP2 Shell Extension and Runtime Features . . . 1050
Creating 64-Bit App-V Packages . . . 1050
Editing an OSD File to Make Advanced Changes to an App-V 4.x Package . . . 1051
How Windows Services Are Integrated into an App-V Package . . . 1052
About VMware ThinApp Virtual Packages . . . 1052
About ThinApp Applications . . . 1052
ThinApp Virtual Operating System . . . 1053
Components of a ThinApp Application . . . 1053
Benefits of Deploying ThinApp Applications . . . 1055
Prerequisites for Building a ThinApp Application . . . 1055
About Citrix XenApp Virtual Packages . . . 1055
About Citrix XenApp and Citrix Profiles . . . 1056
About Citrix XenApp . . . 1056
About Citrix Profiles (.profile) . . . 1057
Benefits of Deploying Citrix XenApp Profiles . . . 1058

About the Automated Application Converter . . . 1060
Benefits of Using the Automated Application Converter . . . 1061
Automated Application Converter Workflow Diagram . . . 1062
Supported Operating Systems . . . 1062
Supported Virtual Machines . . . 1063
Launching the Automated Application Converter . . . 1063

Getting Started With the Automated Application Converter 1064
Opening a Project 1065
Using the Application Conversion Project Wizard to Perform an End-to-End Conversion 1066
About Automated Application Converter Project Files 1079
Using Automated Application Converter in Evaluation Mode 1083
Managing Virtual Machines 1083
Virtual Machine System Requirements 1084
Preparing Your Virtual Machines for Use With the Automated Application Converter 1086
Preparing Virtual Machines 1086
Preparing a Snapshot for Repackaging 1086
Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer 1088
Preparing a Snapshot for App-V 5.0 Testing Using the App-V 5.0 Client 1090
Running the Virtual Machine Preparation Setup 1092
Taking a Snapshot 1092
VMware-Specific Snapshot Configuration Option 1094
VMware vSphere API Requirement on the AdminStudio Machine 1094
Adding Virtual Machines Using the Virtual Machine Import Wizard 1095
Editing Virtual Machine Properties on the Machines Tab 1098
Connecting to Active Virtual Machines 1102
Managing Packages to Convert 1103
Adding Packages from an AdminStudio Application Catalog 1103
Adding Packages from a Local Machine or Network 1107
Editing Package Properties on the Packages Tab 1111
Setting MSIX Signing Options 1111
Setting General Package Properties 1113
Specifying a Package's Repackaging Snapshot 1114
Editing the Installation Command Line 1114
Specifying a Package's Compression Setting 1114
Selecting the Repackaging Method 1115
Specifying Time Out Settings 1115
Enabling Manual Installation During Repackaging 1116
Documenting Interactive Repackaging Steps Using the Microsoft Step Recorder Tool 1116
Enabling Pre-Installation and Post-Installation Configuration 1120
Setting Package Properties for Conversion to App-V Format 1122
Overriding the Name of the App-V Package 1122
Selecting the App-V Conversion Method 1123
Specifying the App-V Package's Primary Application Directory 1123
Specifying the App-V Package's Supported Operating Systems 1124
Specifying How to Optimize the App-V Package 1125
Specifying Whether to Append the Version Number to the App-V Package File Name 1126
Specifying the Diagnostic Tools to Include With the App-V Package 1127
Choosing to Expand the App-V 5.0 Package Before Sequencing 1128
Entering Comments for an App-V Package 1128
Setting the App-V 4.x Package's Server Location 1129
Specifying the App-V Package's Root Folder Name 1130
Enabling Dynamic Suiting for an App-V 4.x Package 1130

Specifying an App-V 4.x Package's Compression Setting *1131*
Designating an App-V Package as an Upgrade *1131*
Specifying an App-V 4.x Package's Client Runtime Drive *1132*
Setting an App-V Package's VFS Options *1132*
About Repackaging Windows Installer Packages 1132
Using the Application Conversion Wizard to Perform Automated Package Conversion 1133
Performing a Conversion Using the Application Conversion Wizard 1134
Viewing Conversion Results 1136
Testing Packages 1137
Testing App-V Packages 1137
Performing Automated Testing of App-V Packages 1138
Performing Manual Testing of App-V Packages 1141
Testing VMware ThinApp Packages 1144
Testing Citrix XenApp Packages 1145
Testing Repackaged and Source Windows Installer Packages 1146
Importing Converted Packages into the Application Catalog 1148
Publishing Converted Packages to a Distribution System 1148
Setting Default Project Properties 1149
Capturing Virtualization Context 1150
Reference 1151
Automated Application Converter User Interface 1152
Packages Tab 1152
Adding Packages to the List *1153*
Viewing Package Information on the Packages Tab *1154*
Packages Tab Properties *1154*
Icons Used on the Packages Tab *1168*
Shortcut Menu Commands on Packages Tab *1170*
Machines Tab 1170
Adding Virtual Machines to the List *1171*
Viewing Virtual Machine Information on the Machines Tab *1172*
Machines Tab Properties *1173*
Shortcut Menu Commands on Machines Tab *1176*
Results Tab 1176
Results Tab Properties *1177*
Icons Used on the Results Tab *1178*
Shortcut Menu Commands on Results Tab *1179*
Menus & Toolbar Buttons 1180
Output Window 1182
Column Selector and Properties Windows 1183
AdminStudio Automated Application Converter Log Report 1183
Using List Features 1186
Sorting Lists *1187*
Changing Which List Columns Are Displayed *1187*
Changing Column Order *1188*
Resizing List Columns *1188*

Grouping Lists *1188*
Wizards 1191
Application Conversion Project Wizard 1191
Open Project Panel *1193*
Application Conversion Project Wizard Welcome *1193*
Select Package Source *1194*
Connect to an AdminStudio Application Catalog *1195*
Select Packages *1196*
Selected Package List *1198*
Select Virtual Machine Source *1202*
Select Virtual Machines from a Microsoft Hyper-V Server *1203*
Select Virtual Machines from a VMware ESX or ESXi Server *1204*
Select Virtual Machines *1205*
User Credentials *1207*
Initial Configuration Complete *1207*
Select Output Formats *1208*
Automated Repackaging on Virtual Machines *1209*
Application Conversion Project Wizard Complete Panel *1210*
Package Import Wizard 1211
Package Import Wizard Welcome *1211*
Package Import Wizard Complete *1212*
Virtual Machine Import Wizard 1213
Virtual Machine Import Wizard Welcome *1213*
Virtual Machine Import Wizard Complete *1214*
Application Conversion Wizard 1214
Application Conversion Wizard Welcome *1215*
Application Conversion Wizard Complete *1215*
Dialog Boxes 1216
About Automated Application Converter 1216
App-V 5.x Application Launcher 1216
Browse for Folder Dialog Box 1218
Guest Agent 1219
Open Dialog Box 1219
MST Dialog Box 1220
Project Options Dialog Box 1221
Select Package Installation File Dialog Box 1230
Select Transform Dialog Box 1231
Select Virtual Machine Dialog Box 1232
Select Virtual Machine Image File Dialog Box 1233
Command Line Support 1233
Specifying Global Default Virtual Conversion Settings 1236
Virtual Converter Table Documentation for Microsoft App-V and VMware ThinApp 1236
Troubleshooting 1247
First Things to Check 1247
Problems and Solutions 1250
Best Practices for Optimal Performance 1254

How to Test a Virtual Machine . . . 1255
Resolving Problems Connecting to a Hyper-V Image . . . 1256
Automated Application Converter Error Messages . . . 1257
Error -4308: VM failed to start up. . . . 1258
Error -4309: VM failed to shut down . . . 1258
Error -4310: Failed to connect to VM . . . 1259
Error -4312: Failed to prepare Repackager . . . 1259
Error -4313: Failed to access the package. . . . 1260
Error -4314: Failed to copy repackaged output from virtual machine . . . 1261
Error -4315: Failed to send command to VM . . . 1261
Error -4316: Failed getting response from VM . . . 1262
Error -4317: Failed running pre-snapshot . . . 1262
Error -4318: Failed running post-snapshot . . . 1263
Error -4319: Failed running package installation . . . 1263
Error -4320: Failed creating folder on VM . . . 1264
Error -4333: Preparing command-line... . . . 1264
Error -4380: Failed to prepare AppV . . . 1265
Error -4388: Failed preparing for pre-snapshot . . . 1265
Error -4389: Failed connecting to server . . . 1266
Error -4390: Failed connecting to image . . . 1266
Error -4391: Failed to reboot . . . 1267
Error -4395: Failed to create VM directory. . . . 1267
Error -4409: Failed to delete package cache folder . . . 1268
Virtualization Conversion Error Messages . . . 1268
Error -9000: Unknown Exception . . . 1268
Error -9001: Unknown COM . . . 1269
Error -9002: Error Opening Package. . . . 1269
Error -9003: Error Saving Package . . . 1269
Error -9004: Process Cancelled By User. . . . 1270
Error -9005: Error Creating Temporary Folder . . . 1270
Error -9006: Error Decompressing Package . . . 1271
Error -9007: File With Extension Not Found . . . 1271
Error -9008: Error Extracting Icon . . . 1272
Error -9009: Unknown Provider. . . . 1272
Error -9010: Invalid Target File Name . . . 1272
Error -9011: Error Reading MSI Table . . . 1273
Error -9012: Unexpected Error in Method . . . 1273
Error -9013: Type Library Not Found . . . 1274
Error -9014: ShellExecute Failed . . . 1274
Error -9015: Unable to Determine Full Path for Driver . . . 1274
Warning -9016: Contents of Table Ignored . . . 1275
Warning -9017: .NET 1.x Assembly Not Supported . . . 1276
Warning -9018: Custom Actions Ignored . . . 1276
Warning -9019: Conditionalized Components . . . 1277
Error -9020: Directory With Null Parent . . . 1278
Error -9021: Unable to Extract COM Data . . . 1278

Error -9022: Complus Table 1279
Error -9024: FileSFPCatalog 1279
Warning -9026: LaunchCondition Table 1279
Warning -9027: LockPermissions Table 1280
Error -9028: MoveFile Table 1281
Error -9029: MsiDriverPackages Table 1281
Warning -9030: ODBCTranslator Table 1282
Warning -9031: RemoveFile Table 1282
Warning -9032: RemoveIniFile Table 1283
Warning -9033: RemoveRegistry Table 1283
Error -9036: ISCEInstall Table 1284
Error -9037: ISComPlusApplication Table 1284
Error -9038: ISPalmApp Table 1285
Error -9039: ISSQLScriptFile Table 1285
Error -9040: ISVRoot Table 1286
Error -9041: ISXmlFile Table 1286
Error -9051: Package Decompression Canceled 1287
Error -9100: CreateInstance of Package Object Failed 1287
Error -9101: Create Operation of Package Object Failed 1287
Error -9102: Failed to Write Header Information 1288
Error -9103: Citrix Finalization Failed 1288
Error -9104: Citrix Save Failed 1289
Error -9105: Error Initializing Citrix Writer 1289
Error -9106: Error Initializing Citrix Package 1289
Error -9107: Error Writing Citrix File Entries 1290
Error -9108: Error Determining Source File Path 1290
Error -9109: Error Writing Citrix Folder Entries 1290
Error -9110: Error Writing Citrix Registry Entries 1291
Error -9113: Error Writing Citrix INI File Entries 1291
Error -9114: Error Writing Citrix Shortcuts 1292
Error -9115: Error Saving Citrix Profile 1292
Error -9116: Error Creating Empty Citrix Profile 1292
Error -9117: Error Creating Intermediate Folder 1293
Error -9118: Error Initializing Citrix Profile 1293
Error -9119: Error Creating Default Target in Citrix Profile 1293
Error -9120: Error Deleting File From Profile 1294
Error -9121: Failed to Copy File into Citrix Profile 1294
Error -9122: Target Does Not Exist in Citrix Profile 1294
Error -9124: No Shortcuts Created for this Profile 1295
Error -9125: Error Writing Citrix File Type Associations 1295
Error -9126: Failed to Sign Profile Using Certificate 1296
Error -9127: Could Not Create Script Execution 1296
Warning -9128: Duplicate Shortcut 1296
Warning -9129: Duplicate Shortcut Names 1297
Warning -9130: Duplicate Shortcut Targets 1297
Warning -9131: Unable to Resolve Installer Variable 1298

Warning -9132: 16 Color Shortcut Icon Not Found . . . 1298
Warning -9133: Shortcut Icon Not Found . . . 1298
Warning -9134: Failure to Extract Icon from Executable . . . 1299
Error -9135: Shortcut Target is 16-Bit. . . . 1299
Warning -9136: Some Files May Not Be Decompressed . . . 1300
Warning -9137: Destination Directory Cannot Be Found . . . 1300
Warning -9138: Ignoring a DuplicateFile Table Entry . . . 1301
Error -9139: 64-Bit Executables (XenApp) . . . 1301
Error -9200: ThinApp Must Be Installed . . . 1302
Warning -9201: Extension for Shortcut Files Must Be “.exe”. . . . 1302
Error -9202: No Applications Were Created . . . 1303
Error -9203: ThinApp Tool is Missing . . . 1303
Error -9204: Duplicate Shortcut. . . . 1303
Error -9205: Identically-Named Shortcut Already Exists, But With Different Parameters . . . 1304
Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target . . . 1304
Error -9207: Error During Build Process (vregtool.exe) . . . 1304
Error -9208: Error Occurred During Build Process (vftool.exe). . . . 1305
Error -9209: Error Occurred During ThinApp Build Process (tlink.exe). . . . 1305
Error -9210: 64-Bit Executables (ThinApp) . . . 1306
Error -9300: Unhandled Exception During AdviseFile Operation. . . . 1306
Error -9301: Unhandled Exception During AdviseRegistry Operation . . . 1306
Error -9302: Unhandled Exception During Command Action . . . 1307
Error -9303: Unhandled Exception During Alter File Action . . . 1307
Error -9304: Unhandled Exception During Alter Registry Action . . . 1307
Error -9305: Unhandled Exception During Create Action . . . 1308
Error -9306: Unhandled Exception During Execution of Rules Engine . . . 1308
Error -9401: Error Initializing App-V Writer . . . 1308
Error -9402: Error Initializing App-V Package . . . 1309
Error -9403: Error Writing App-V File Entries . . . 1309
Error -9404: Error Writing App-V Folder Entries . . . 1309
Error -9405: Error Writing App-V Registry Entries . . . 1310
Error -9406: Error Writing App-V INI File Entries . . . 1310
Error -9407: Error Writing App-V Shortcuts . . . 1310
Error -9408: Error Writing App-V File Type Data. . . . 1311
Error -9409: Error Saving App-V Data . . . 1311
Error -9410: Error Determining Source File Path . . . 1311
Error -9411: OSD File Template Could Not Be Extracted . . . 1312
Error -9412: OSD File Could Not Be Saved . . . 1312
Error -9413: App-V OSD Save . . . 1312
Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application . . . 1313
Error -9415: Dependency Application Was Not Found . . . 1313
Warning -9416: Invalid Primary Application Directory . . . 1313
Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value . . . 1314
Error -9418: Error While Privatizing Side-By-Side Assemblies . . . 1314
Error -9419: Error Inserting Watermark . . . 1315
Error -9420: Error During App-V Package Upgrade . . . 1315

Warning -9421: 16-Bit Application . . . 1315
Error -9422: Package Cannot Be Opened . . . 1316
Warning -9423: No Shortcuts Detected . . . 1316
Error -9424: Windows 8 or Windows 2012 OS Required . . . 1317
Warning -9500: Shortcut Missing . . . 1317
Error -9600: Error Initializing Symantec Writer . . . 1318
Error -9601: Error Writing Symantec Folder Entries . . . 1318
Error -9602: Error Writing Symantec Shortcuts . . . 1319
Error -9603: Error Creating Target File for Symantec Package . . . 1319
Error -9604: Error Writing Symantec File Entries . . . 1319
Error -9605: Error Writing Symantec Registry Entries . . . 1320
Error -9606: Error Writing Symantec INI File Entries . . . 1320
Error -10000: Process Cancelled By User . . . 1321
Warning -10001: Suite File Missing . . . 1321
Warning -10002: Suite File is Duplicate . . . 1321
Warning -10003: Application File Missing . . . 1322
Warning -10004: INI File Missing . . . 1322
Fix 11000: Excluding TCPIP Registry Entries . . . 1322
Fatal Error 11001: Fail on VMware . . . 1323
Warning 11003: Control Panel Applet - Citrix . . . 1323
Fix 11004: Control Panel Applet - ThinApp . . . 1323
Fatal Error 11005: QuickTime 7.4.1 Causes Fatal Error . . . 1324
Fix 11006: Adobe Distiller Exclude AdobePDFSettings . . . 1324
Fix 11007: Exclude URL Shortcut . . . 1324
Steps to Take Before Calling Technical Support . . . 1325
Application Features Requiring Pre- or Post-Conversion Actions . . . 1325

11 Using the MSIX Editor . . . 1329

About MSIX Editor . . . 1329
MSIX Package Format . . . 1330
Getting Started with the MSIX Editor . . . 1330
Launching MSIX Editor . . . 1330
Opening an MSIX Package . . . 1333
Working with MSIX Editor Interface . . . 1334
MSIX Editor Ribbon . . . 1334
Preferences . . . 1336
Output Window . . . 1337
Editing MSIX Packages . . . 1338
Editing MSIX Package Information . . . 1338
Editing Files and Folders . . . 1338
Adding a File or Folder to an MSIX Package . . . 1338
Deleting a File or Folder From an MSIX Package . . . 1340
Editing Registry . . . 1341
Importing a Registry File . . . 1341
Adding a Registry Key . . . 1342
Adding a Registry Value and Data to a Registry Key . . . 1343

Modifying a Registry Value and Data of a Registry Key . . . 1344
Deleting a Registry Key or Registry Value . . . 1344
Editing Dependencies . . . 1345
Adding Dependency . . . 1345
Modify Dependency . . . 1346
Delete Dependency . . . 1347
Editing Package Declarations . . . 1347
Add Package Declarations . . . 1347
Modify Package Declarations . . . 1348
Delete Package Declarations . . . 1348
Editing Capabilities . . . 1349
Editing Applications . . . 1349
Add a New Application . . . 1350
Add a New Non-EXE Application . . . 1351
Modify an Existing Application . . . 1351
Delete an Existing Application . . . 1352
Editing Visual Assets . . . 1353
Modify Visual Assets . . . 1354
Generate Visual Assets . . . 1354
Delete Visual Assets . . . 1356
Signing Package . . . 1356
Fix Runtime Issues of an MSIX Package . . . 1356
Tracing App . . . 1356
Applying Fixups . . . 1357
File Redirection Fixup . . . 1357
Package Relative . . . 1358
Package Drive Relative . . . 1359
Known Folders . . . 1361
PowerShell Script Fixup . . . 1363
File Not Found Fixup . . . 1366
Executable Arguments Fixup . . . 1367
Registry Fixup . . . 1368
Environment Variable Fixup . . . 1370
Custom Fixup . . . 1371
Applying Recommended Fixup . . . 1372
Create Modification Packages . . . 1375
Specifying MSIX Package Information . . . 1375
Including Files and Folders . . . 1376
Including Registry . . . 1376
Saving as Modification Package . . . 1376
Signing Modification Package . . . 1379
MSIX Editor Reference . . . 1379
Package Properties . . . 1379
Package Information View . . . 1379
Package Identity . . . 1380
Package Properties . . . 1380

Files and Folders View . . . 1381
Registry View . . . 1383
Dependencies View . . . 1383
Target Device Family . . . *1384*
Package Dependency . . . *1384*
Main Package Dependency . . . *1385*
Driver Dependency . . . *1385*
OS Package Dependency . . . *1385*
Package Declarations View . . . 1385
Firewall Rules . . . *1386*
Com Interface . . . *1386*
Loader Search Path Override . . . *1388*
Capabilities View . . . 1388
Device Capabilities . . . *1388*
General Capabilities . . . *1389*
Restricted Capabilities . . . *1391*
Application Properties . . . 1392
Applications View . . . 1393
Visual Assets View . . . 1394
App Declarations View . . . 1401
Content URIs View . . . 1402
Advanced Editor . . . 1402
Fixups View . . . 1402
Recommended Fixups View . . . 1403
App Manifest View . . . 1403

12 Using the Virtual Package Editor . . . 1405

Contacting Us . . . 1405
About Virtualization . . . 1406
About the Virtual Package Editor . . . 1407
Components of an App-V Package . . . 1408
Getting Started with the Virtual Package Editor . . . 1409
Starting the Virtual Package Editor . . . 1409
Opening an Existing Virtual Package . . . 1409
Saving a Virtual Package . . . 1410
Closing a Virtual Package . . . 1413
Working with the Virtual Package Editor Interface . . . 1413
Configuring the Value of a Setting for More Than One Item at a Time . . . 1413
Showing or Hiding the Start Page in the Virtual Package Editor . . . 1414
Rearranging the Start Page and Virtual Package Tabs . . . 1414
Showing or Hiding the Settings and Output Windows . . . 1415
Moving the Settings, Output, and Script Windows . . . 1415
Showing or Hiding Toolbars . . . 1415
Adding Buttons and Menus to a Toolbar . . . 1415
Removing Buttons and Menus from a Toolbar . . . 1416
Creating a Custom Toolbar . . . 1416

Editing Virtual Packages 1417
Specifying Virtual Package Information 1417
Viewing History for a Virtual Package 1417
Configuring General Information for a Virtual Package 1418
Specifying a Virtual Package's Dependencies 1418
Adding a Dependency to a Virtual Package 1419
Configuring a Dependency in a Virtual Package 1419
Associating a Package's Targets with a Dependency in a Virtual Package 1419
Specifying Whether a Dependency is Mandatory for a Target in a Virtual Package 1420
Removing a Target from a Dependency in a Virtual Package 1420
Removing a Dependency from a Virtual Package 1421
Configuring Asset Intelligence Information 1421
Organizing Virtual Application Data 1422
Including Files and Folders 1422
Adding a Predefined Folder to the VFS Folder in an App-V Package 1422
Adding a Folder to an App-V Package 1422
Adding a File to an App-V Package 1423
Configuring a File or Folder in an App-V Package 1423
Setting the VFS Path for the Contents of a Predefined Folder in an App-V Package 1424
Moving a File or Folder in an App-V Package 1424
Extracting Files and Folders from the App-V Package 1425
Removing a File or Folder in an App-V Package 1425
Editing the Virtual Registry 1426
Adding a Registry Key to a Virtual Package 1426
Configuring a Registry Key in a Virtual Package 1427
Configuring the Isolation Setting for All of the Subkeys Under One or More Keys 1427
Adding a Registry Value to a Registry Key in a Virtual Package 1428
Configuring a Registry Value and Its Value Data in a Virtual Package 1428
Removing a Registry Value from a Registry Key in a Virtual Package 1429
Removing a Registry Key from a Virtual Package 1429
Defining Targets in a Virtual Application 1429
Adding a Target to a Virtual Package 1429
Configuring a Target in a Virtual Package 1430
Removing a Target from a Virtual Package 1430
Creating Shortcuts to the Virtual Application on the Client System 1430
Adding a Shortcut for a Virtual Package 1430
Configuring a Shortcut in a Virtual Package 1431
Removing a Shortcut from a Virtual Package 1432
Using Environment Variables in a Virtual Environment 1432
Setting an Environment Variable in a Virtual Package 1432
Configuring an Environment Variable in a Virtual Package 1433
Removing an Environment Variable from a Virtual Package 1434
Configuring File Extension Associations for the Virtual Application 1434
Adding a File Extension to a Virtual Package 1434
Configuring a File Extension in a Virtual Package 1436
Adding a Verb to a File Extension in a Virtual Package 1436

Configuring a Verb for a File Extension in a Virtual Package ... *1437*
Removing a Verb from a File Extension in a Virtual Package ... *1438*
Removing a File Extension from a Virtual Package ... *1439*
Creating Scripts that Run Before or After the App-V Application Is Streamed or Launched ... 1439
Adding a Script to a Target in a Virtual Package ... *1440*
Configuring a Script in a Virtual Package ... *1440*
Causing the App-V Application to Close After a Script Failure ... *1441*
Removing a Script from a Virtual Package ... *1442*
Specifying the Application Path for a File in a Virtual Package ... 1442
Adding an Application Path to a Virtual Package ... *1443*
Configuring an Application Path in a Virtual Package ... *1443*
Removing an Application Path from a Virtual Package ... *1444*
Configuring Virtual Services ... 1444
Adding a Virtual Service to a Virtual Package ... 1444
Configuring a Virtual Service in a Virtual Package ... 1445
Removing a Virtual Service from a Virtual Package ... 1445
Testing and Troubleshooting Virtual Packages ... 1445
Using the App-V Application Launcher to Test the Virtual Package ... 1445
Using Debug Tools with a Virtual Package ... 1447
Using the Virtual Package Editor to Resolve Application Conflict Evaluators (ACEs) in App-V Packages ... 1448
Virtual Package Editor Reference ... 1453
Virtual Package Editor Start Page ... 1453
Virtual Package Editor Menu, Toolbar, and Window Reference ... 1454
Menus in the Virtual Package Editor ... 1454
File Menu in the Virtual Package Editor ... *1454*
Edit Menu in the Virtual Package Editor ... *1455*
View Menu in the Virtual Package Editor ... *1456*
Window Menu in the Virtual Package Editor ... *1456*
Help Menu in the Virtual Package Editor ... *1457*
Standard Toolbar in the Virtual Package Editor ... 1457
Script Window ... 1458
Settings Window ... 1458
Output Window ... 1458
Virtual Package Editor Dialog Box Reference ... 1458
Browse for Folder Dialog Box ... 1459
Edit Value Dialog Box ... 1459
Save As Dialog Box ... 1459
Select a File Dialog Box ... 1460
Select a Folder Dialog Box ... 1460
Select Files to Add to the Virtual Package Dialog Box ... 1460
Virtual Package Editor View Reference ... 1460
Application Paths View ... 1461
Asset Intelligence View ... 1461
Dependencies View ... 1463
Environment Variables View ... 1464
File Extensions View ... 1465

File Extension Settings *1466*
Verb Settings for a File Extension *1467*
Files and Folders View 1469
General Information View 1472
Registry View 1476
Shortcuts View 1478
Target Settings *1480*
Shortcut Settings *1481*
Virtual Services View 1484

13 Creating Customized Virtual Applications 1489

About Virtualization 1489
About the AdminStudio Virtualization Interface 1491
About the Virtualization Assistant Tabs 1492
Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant 1492
Navigating in a Virtualization Assistant 1493
Opening the Installation Designer 1493
Showing or Hiding the Virtualization Assistants 1493
Creating Microsoft App-V Packages 1493
Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant 1494
About Microsoft Application Virtualization (App-V) and the Microsoft App-V Assistant 1495
Components of an App-V Package 1497
About the Microsoft App-V Assistant 1498
Process for Authoring an App-V Package Using the Microsoft App-V Assistant *1498*
Supported InstallShield Project Types *1499*
How Transforms are Included in an App-V Package *1499*
How Windows Services Are Integrated into an App-V Package *1500*
Using the Microsoft App-V Assistant to Create an App-V Package 1500
Specifying Package Information and Deployment Options 1501
Specifying Package Information *1501*
Specifying Operating System Requirements *1501*
Specifying Upgrade Package Information *1502*
Specifying the Deployment Server *1502*
Including Diagnostic Tools in an App-V Package *1502*
Managing Files in an App-V Package 1503
Adding, Deleting, and Moving Files and Folders in an App-V Package *1504*
Controlling the Display of Predefined Folders *1506*
Specifying the Primary Application Directory *1506*
Setting Isolation Options for Folders and Files 1508
Inheritance of Isolation Options from Folders to Files *1508*
Modifying Shortcuts to the App-V Package's Executable Files 1509
App-V Packages and the Virtual Environment *1509*
App-V Shortcut Requirements *1509*
Creating a New App-V Package *1509*
Including an Existing App-V Shortcut *1510*
Excluding or Deleting an Existing App-V Package *1510*

Excluding vs. Deleting App-V Package Shortcuts *1511*
Renaming a Shortcut *1511*
Modifying App-V Package Registry Settings 1511
About the Windows Registry *1512*
Adding or Deleting Registry Keys and Values *1512*
Setting App-V Package Registry Isolation Options 1512
Inheritance of Isolation Options in the Registry *1513*
Performing Dynamic Suite Composition 1514
Modifying Build Options 1515
Selecting the Releases for Which You Want to Build App-V Packages *1515*
Enabling App-V Package Building When in Direct Edit Mode *1516*
Specifying Whether to Compress the Data Files in an App-V Package *1516*
Including Additional Windows Installer Packages in an App-V Package *1517*
Building a Windows Installer Package to Assist in the Distribution of an App-V Package *1517*
Specifying Package Feature Block Optimizations *1518*
Building an App-V Package 1519
Build Output for App-V Packages *1520*
Building App-V Packages Through the Command Line *1522*
Testing an App-V Package Using the App-V Launcher Tool 1522
Troubleshooting the Builds of App-V Packages 1523
Application Features that Require Pre- or Post-Conversion Actions 1523
Microsoft App-V Assistant Reference 1523
Microsoft App-V Assistant Pages 1523
Microsoft App-V Assistant Home Page *1523*
Package Information Page *1524*
Files Page *1527*
Applications Page *1527*
Registry Page *1527*
Dynamic Suite Composition Page *1528*
Build Options Page *1529*
Microsoft App-V Assistant Dialog Boxes 1531
Advanced Settings Dialog Box *1531*
App-V Diagnostic Tools Dialog Box *1533*
App-V Package Upgrade Settings Dialog Box *1533*
File Mapping Dialog Box *1534*
Isolation Options Dialog Box (for a Package) *1535*
Isolation Options Dialog Box (for Registry Keys) *1536*
Launch App-V Package Dialog Box *1536*
Options Dialog Box (for Configuring Isolation Options for a File) *1536*
Options Dialog Box (for Configuring Isolation Options for a Folder) *1537*
Package Optimizations Dialog Box *1538*
Advanced Table Settings for Conversion to Microsoft App-V 1539
Creating Citrix Profiles 1547
Overview of the Citrix Assistant 1547
About Citrix XenApp 1549
About the Citrix Assistant 1550

About Citrix Profiles . . . 1551
Benefits of Deploying Citrix Profiles . . . 1552
Supported InstallShield Project Types . . . 1554
How Transforms are Included in a Citrix Profile . . . 1554
Using the Citrix Assistant to Create a Citrix Profile . . . 1554
Specifying Citrix Profile Information . . . 1555
Specifying the Profile Name, Description, and Version . . . 1555
Specifying Whether Users Should Be Able to Update Applications . . . 1555
Including Diagnostic Tools With a Citrix Profile . . . 1556
Specifying Operating System and Language Requirements . . . 1557
Setting Operating System Requirements and Service Pack Levels . . . 1557
Setting Language Requirements . . . 1558
How Requirements are Applied at Runtime . . . 1559
Adding Pre-Launch and Post-Exit Scripts . . . 1560
Managing Files and Folders in a Citrix Profile . . . 1562
Managing Files and Folders in a Citrix Profile . . . 1562
Controlling the Display of Predefined Folders . . . 1566
Setting Isolation Options . . . 1567
Overview of Citrix Isolation Options . . . 1568
Setting Isolation Options for Folders and Files . . . 1569
Inheritance of Isolation Options from Folders to Files . . . 1570
Modifying Profile Shortcut Settings . . . 1570
Shortcuts and the Isolation Environment . . . 1571
Shortcut Requirements . . . 1573
Creating a New Profile Shortcut . . . 1573
Including an Existing Profile Shortcut . . . 1574
Excluding vs. Deleting a Profile Shortcut . . . 1574
Conditions When a Shortcut Should be Excluded or Deleted . . . 1575
Renaming a Shortcut . . . 1575
Modifying Profile Registry Settings . . . 1576
About the Windows Registry . . . 1576
Adding or Deleting Registry Keys and Values . . . 1577
Setting Registry Isolation Options . . . 1577
Inheritance of Isolation Options in the Registry . . . 1578
Modifying Build Settings . . . 1578
Selecting Releases to Build . . . 1579
Digitally Signing a Citrix Profile . . . 1579
Including Additional Windows Installer Packages in a Citrix Profile . . . 1580
Enabling Citrix Profile Building When in Direct Edit Mode . . . 1580
Building a Citrix Profile . . . 1580
Citrix Assistant Reference . . . 1582
Pages . . . 1582
Home Page . . . 1583
Profile Information Page . . . 1584
Profile Requirements Page . . . 1586
Profile Files Page . . . 1589

Profile Shortcuts Page *1591*
Profile Registry Page *1593*
Build Settings Page *1594*
Dialog Boxes 1597
Script Execution Dialog Box *1597*
Diagnostic Tools Dialog Box *1598*
File Isolation Options Dialog Box *1599*
Folder Isolation Options Dialog Box *1601*
Registry Isolation Options Dialog Box *1602*
Service Packs Requirement Dialog Box *1604*
Building Citrix Profiles Using the Command Line 1605
Citrix Profile Conversion Error and Warning Messages 1605
Application Features Requiring Pre- or Post-Conversion Actions 1605
Creating ThinApp Applications **1606**
Overview of the ThinApp Assistant 1606
About ThinApp Applications 1607
The ThinApp Virtual Operating System *1607*
Benefits of Deploying ThinApp Applications *1607*
About the ThinApp Assistant 1608
Process for Authoring a ThinApp Application Using the ThinApp Assistant *1608*
Components of a ThinApp Application *1611*
Supported InstallShield Project Types *1612*
How Transforms are Included in a ThinApp Application *1612*
About Sandboxes *1613*
Using the ThinApp Assistant to Create a ThinApp Application 1613
Specifying ThinApp General Settings 1614
Specifying Sandbox Information *1614*
Specifying Control Access via Active Directory *1615*
Prerequisites for Building a ThinApp Application *1616*
Including Diagnostic Tools With a ThinApp Application *1616*
Managing Files and Folders in a ThinApp Application 1617
Adding, Deleting, and Moving Files and Folders in a ThinApp Application *1617*
Controlling the Display of Predefined Folders *1621*
Setting ThinApp Isolation Options 1622
Overview of ThinApp Isolation Options *1623*
Setting Isolation Options for Folders *1625*
Inheritance of Isolation Options from Folders to Files *1626*
Modifying Shortcuts to the ThinApp Application's Executable Files 1626
ThinApp Applications and the Virtual Environment *1627*
Compressing a ThinApp Application *1627*
ThinApp Shortcut Requirements *1628*
Creating a New ThinApp Application *1628*
Including an Existing ThinApp Application *1629*
Excluding or Deleting an Existing ThinApp Application *1629*
Excluding vs. Deleting ThinApp Application Shortcuts *1630*
Renaming a ThinApp Application *1630*

Modifying ThinApp Application Registry Settings . . . 1631
About the Windows Registry . . . 1631
Adding or Deleting Registry Keys and Values . . . 1632
Setting ThinApp Isolation Options on Registry Keys . . . 1632
Inheritance of ThinApp Isolation Options in the Registry . . . 1633
Modifying Build Options . . . 1633
Selecting Releases to Build . . . 1634
Enabling ThinApp Application Building When in Direct Edit Mode . . . 1634
Including Additional Windows Installer Packages in a ThinApp Application . . . 1635
Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application . . . 1635
Setting ThinApp Log Monitor Tracing Options . . . 1635
Setting AppLink Options . . . 1637
Setting AppSync Options . . . 1639
Building a ThinApp Application . . . 1642
ThinApp Assistant Reference . . . 1643
Pages . . . 1644
ThinApp Assistant Home Page . . . 1644
General Settings Page . . . 1646
Files & Folders Page . . . 1648
Applications Page . . . 1650
Registry Page . . . 1651
Build Options Page . . . 1653
Dialog Boxes . . . 1659
ThinApp Diagnostic Tools Dialog Box . . . 1659
Folder Isolation Options Dialog Box . . . 1660
Registry Isolation Options Dialog Box . . . 1662
AppLink Settings Dialog Box . . . 1664
Add AppLink Reference Dialog Box . . . 1667
AppSync Settings Dialog Box . . . 1669
Building ThinApp Applications Using the Command Line . . . 1673
ThinApp Application Conversion Error and Warning Messages . . . 1673
Application Features Requiring Pre- or Post-Conversion Actions . . . 1673
ThinApp Not Found . . . 1673
ThinApp Application Configuration File: package.ini . . . 1673
[BuildOptions] . . . 1674
[Compression] . . . 1679
[Isolation] . . . 1679
[MainApp.exe] . . . 1680
[Test.exe] . . . 1682

14 Customizing and Authoring Installations Using InstallShield . . . 1683

AdminStudio-Specific Functionality in InstallShield Editor . . . 1684
InstallShield Editor Integration with Application Catalog and the Software Repository . . . 1684
InstallShield Integration with Application Catalog Software Repository . . . 1684
Quickly Opening Package in InstallShield Direct Edit Mode . . . 1687
Quickly Creating and Opening a Transform File in InstallShield Direct MST Mode . . . 1687

Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support . . . 1689
Differences Between InstallShield Editor and InstallShield Professional and Premier Editions . . . 1689
InstallShield Editor Help Library . . . 1690

15 Customizing Installations Using Tuner . . . 1691

Working with Transforms . . . 1692
Creating New Transform Files . . . 1693
Opening Existing Transforms . . . 1694
Opening Recently Accessed Transforms . . . 1695
Creating Generic Transforms . . . 1695
Using Response Transforms . . . 1695
Viewing Transform Properties . . . 1696
Validation . . . 1696
Prevalidating Windows Installer Packages . . . 1697
Handling Invalid Windows Installer Packages . . . 1698
Postvalidating Transforms . . . 1698
Evaluation Files and Internal Consistency Evaluators . . . 1699
Setup Organization . . . 1699
Changing a Feature's Visibility . . . 1700
Setting the Initial State of a Feature . . . 1700
Editing a Feature's Description . . . 1701
Setting the Default Destination . . . 1701
Setting the Default Organization . . . 1701
Changing the Destination Variable . . . 1702
Preventing Features from Displaying During Custom Installation . . . 1702
Setting Feature Properties . . . 1702
Using Feature Advertisement . . . 1703
Configuring Package Content . . . 1703
Files and Folders . . . 1704
Adding Files . . . 1704
Displaying Files from the Base Windows Installer Package . . . 1704
Preventing Installation of Files from the MSI . . . 1705
Removing Added Files . . . 1705
Storing Added Files . . . 1705
Registry Entries . . . 1706
Creating a Registry Key . . . 1706
Creating a Registry Value . . . 1707
Importing REG Files . . . 1708
Removing Registry Information . . . 1708
Shortcuts . . . 1708
Creating Shortcuts . . . 1709
Changing a Shortcut's Icon . . . 1709
Change a Shortcut's Location . . . 1710
Changing a Shortcut's Target . . . 1710
Creating a Hot Key . . . 1710

Removing Shortcuts . . . 1711
Determining the Path of Changed Shortcuts . . . 1711
INI Files . . . 1711
Adding INI Files . . . 1712
Importing Existing INI Files . . . 1712
Adding Sections to INI Files . . . 1713
Adding New Keys to INI File Sections . . . 1713
Modifying INI File Keys, Values, and Actions . . . 1713
Removing INI Files . . . 1714
Removing Sections from INI Files . . . 1714
Removing INI File Section Keys . . . 1715
ODBC Resources . . . 1715
Adding New Data Sources . . . 1715
Adding New ODBC Data Source Attributes . . . 1716
Adding New ODBC Driver Attributes . . . 1716
Editing ODBC Data Source Attributes . . . 1716
Editing ODBC Driver Attributes . . . 1716
Removing Existing ODBC Data Sources . . . 1717
Removing ODBC Driver Attributes . . . 1717
Removing ODBC Data Source Attributes . . . 1717
NT Services . . . 1717
Setting NT Service Arguments . . . 1718
Setting NT Service Dependencies . . . 1718
Setting the NT Service Description . . . 1718
Setting the NT Service Display Name . . . 1718
Setting the NT Service Error Control Level . . . 1719
Setting the NT Service Load Order Group . . . 1719
Setting the NT Service Overall Install Result . . . 1719
Setting the NT Service Start Type . . . 1720
Setting NT Service Start Name and Password . . . 1720
Setting the NT Service Type . . . 1720
Working with Dialogs . . . 1721
Hiding Dialogs During UI Sequences . . . 1721
Restoring Dialog Sequences . . . 1721
Suppressing the License Agreement Dialog Box . . . 1722
Disabling Custom Setups . . . 1722
Editing Dialog Properties . . . 1723
Dialogs View vs. Command-Line Options . . . 1723
Dialog Suppression Issues . . . 1723
Configuring Additional Server Locations . . . 1724
Adding Additional Server Locations . . . 1724
Modifying Server Locations . . . 1724
Removing Server Locations . . . 1725
Reordering Server Locations . . . 1725
Changing Add/Remove Program Settings . . . 1725
Changing Add/Remove Programs Properties . . . 1725

Disabling the Modify, Remove, or Repair Buttons . . . 1726
Customizing Setup Properties . . . 1726
Adding Custom Setup Properties . . . 1726
Adding and Editing Comments . . . 1726
Removing Custom Setup Properties . . . 1727
Modifying Setup Properties . . . 1727
Preparing Packages for Distribution . . . 1727
Copying the Installation to a Network Location . . . 1728
Copying the Installation to an FTP Server . . . 1728
Creating a Package Definition File (PDF) . . . 1728
Creating an SMS File . . . 1729
Instructing SMS to Create a Management Information Format File at Deployment Time . . . 1730
Deploying Windows Installer Setup Packages with Systems Management Server 2.0 . . . 1730
Creating a Setup.exe File for the Package and Transform . . . 1731
Additional Setup.ini Parameters . . . 1731
Directly Editing Packages . . . 1731
Adding a New Record Using the Direct Editor . . . 1732
Finding and Replacing Using the Direct Editor . . . 1732
Launching the Direct Editor from the Validation Tab . . . 1732
Documenting Response Transform Creation Using the Microsoft Step Recorder Tool . . . 1733
Tuner Reference . . . 1735
User Interface Reference . . . 1736
Menus and Toolbar . . . 1736
View Bar . . . 1739
Checklist . . . 1739
Customization Steps Checklist . . . 1740
Output Window . . . 1740
Customize Dialog Box . . . 1741
Properties Dialog Box . . . 1741
Options Dialog Box . . . 1741
Transform Summary Dialog Box . . . 1742
Dialog Properties Dialog Box . . . 1743
Tuner Views . . . 1744
Tuner Start Page . . . 1745
Welcome to Tuner View . . . 1745
Create a New Transform View . . . 1745
Open a Recent Transform View . . . 1747
Open an Existing Transform View . . . 1748
Help View . . . 1748
Package Validation View . . . 1748
Prevalidation View . . . 1748
Organization View . . . 1749
Product Properties View . . . 1749
Features View . . . 1750
System Configuration View . . . 1751
Files and Folders View . . . 1751

Registry View *1752*
Shortcuts View *1753*
Shortcuts View/Shortcut Properties *1753*
Shortcuts View/Shortcut Target *1755*
Shortcuts View/Shortcut Locations *1756*
INI Files View *1757*
ODBC Resources View *1758*
NT Services View *1759*
Application Configuration View 1761
Server Locations View *1761*
Setup Properties View *1761*
Dialogs View *1762*
Add/Remove Programs View *1762*
Package Preparation View 1763
Postvalidation View *1764*
Package View *1765*
Package View/Location View *1765*
Package View/Setup View *1765*
Package View/SMS View *1765*
Additional Tools View 1766
Direct Editor *1766*
Import INI File Wizard 1767
Welcome Panel 1768
Import INI File Panel 1768
Import Conflict Options Panel 1768
Finishing INI File Import Panel 1768
Import REG File Wizard 1768
Welcome Panel 1769
Import Registry File Panel 1769
Import Conflict Options Panel 1769
Finishing Registry Import Panel 1769
Packaging Wizard 1769
Location Panel 1770
Setup.exe Panel 1770
SMS Panel 1770
Packaging Summary Panel 1770

16 Using Analyze to Perform Package Testing 1771

Analyze Overview 1772
Benefits of Using Analyze 1774
Test Run Optimization 1774
About Mobile Application Testing 1775
About Microsoft Windows Application Compatibility Infrastructure Testing 1776
Configuring Testing Options on the Analyze Tab 1778
About Analyze Tests 1778
Selecting Tests to Execute 1780

Setting the Compliance Level for Operating System Compatibility Tests . . . 1782
Setting Automatic Fix Preferences for Operating System Compatibility Tests . . . 1786
Updating the Location of the Custom ACE Rule File . . . 1787
Changing the ICE Validation File . . . 1787
Creating Custom Mobile Tests Using the Mobile Test Wizard . . . 1788
Performing Compatibility, Best Practices, and Risk Assessment Testing . . . 1794
Performing MSIX Conversion Compatibility Testing . . . 1795
Testing Windows Installer Packages for MSIX Conversion Compatibility . . . 1795
Converting your Existing Packages to MSIX using Repackager Tool and Digital Signing . . . 1797
Signing the MSIX Package in the Application Catalog . . . 1799
Scanning and Checking Whether or Not the Package Has Shortcuts for MSIX Conversion . . . 1801
MSIX App Attach . . . 1802
About MSIX App Attach . . . 1802
Prerequisites for Creating App Attach . . . 1803
Creating MSIX App Attach . . . 1803
Wrapping MSI/EXE Packages to Wise Script Wrapped EXEs . . . 1805
Wrap MSI/EXE Packages Using the Wrap Package Wizard . . . 1806
Performing Application Conflict Testing . . . 1811
Testing for Conflicts Between Packages . . . 1812
Testing for Conflicts Between Packages and OS Snapshots . . . 1812
Integrating Analyze With Other Applications . . . 1813
Integrating with Microsoft Application Compatibility Toolkit (ACT) . . . 1813
Viewing and Filtering Test Results . . . 1814
About Status Icons . . . 1815
Viewing Summary Group/Application Test Results . . . 1817
Viewing Detailed Package Test Results . . . 1819
Viewing Summary Test Results . . . 1819
Viewing Operating System Compatibility Test Results . . . 1821
Viewing MSIX Conversion Compatibility Test Results . . . 1823
Viewing Application Virtualization Compatibility Test Results . . . 1824
Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter . . . 1827
Viewing Remote Application Publishing Compatibility Test Results . . . 1828
Viewing Best Practices Test Results . . . 1830
Viewing Risk Assessment Test Results . . . 1832
Viewing Application Conflicts Test Results . . . 1833
Viewing Combined Test Results of Bundled Packages . . . 1835
Viewing Combined Test Results of Child Windows Installer Packages of Complex Installer Executables . . . 1835
Viewing Combined Test Results of Child Applications of PKG and DMG Installers . . . 1836
Filtering Test Results by Suppressing Errors/Warnings . . . 1837
Resolving Issues . . . 1838
Performing Automatic Issue Resolution . . . 1839
Performing Manual Issue Resolution . . . 1841
Viewing Test Summary Reports on Reports Tab . . . 1842
Analyze Reference . . . 1842
Analyze Views . . . 1843

Analyze Group View . . . 1843
Analyze Application View . . . 1845
Analyze Deployment Type View . . . 1845
Supportability Risks . . . *1846*
Analyze Subnode Views . . . 1853
Patch Impact View . . . *1853*
Associated Patches View . . . *1854*
Analyze Dialog Boxes . . . 1854
About Application Catalog Dialog Box . . . 1854
ACE Rule Properties Dialog Box . . . 1855
General Information Tab . . . *1855*
Additional Information Tab . . . *1856*
Custom Options Tab . . . *1856*
Where Clause Tab . . . *1857*
DLL Information Tab . . . *1858*
Add Ignore Table Dialog Box . . . 1858
Expression Builder Dialog Box . . . 1858
Rules Viewer Dialog Box . . . 1859
Select Tests to Execute Dialog Box . . . 1860
Analyze Wizards . . . 1860
AdminStudio Test Configuration Wizard . . . 1861
Compliance Level Panel . . . *1861*
OS Snapshot(s) Panel . . . *1863*
Summary Panel . . . *1864*
Conflict Wizard . . . 1866
Target Information Panel . . . *1867*
Target OS Snapshot Information Panel . . . *1867*
Summary Panel . . . *1867*
Mobile Test Wizard . . . 1868
Select the Tests Panel . . . *1868*
Provide the Test Details Panel . . . *1869*
Summary Panel . . . *1872*
Rules Wizard . . . 1872
Welcome Panel . . . *1873*
General Information Panel . . . *1873*
Additional Information . . . *1874*
Custom Options Panel . . . *1875*
Token Grammar . . . *1876*
Where Clause Panel . . . *1877*
DLL-Based ACEs Panel . . . *1877*
Summary Panel . . . *1877*

17 Analyze Tests . . . 1879

Operating System Compatibility Tests . . . 1881
Windows Server and Desktop Tests . . . 1882
32-Bit Driver . . . 1884

32-Bit Shell Extensions . . . 1885
64-Bit Files . . . 1886
AdminUser or Privileged Launch Condition . . . 1887
Application Requires Specific Minimum OS Version . . . 1888
Application Requires VCLibs 12.0 . . . 1889
Application Requires VCLibs 14.0 . . . 1890
Application Requires WinJS 2.0 or Higher . . . 1890
Applications with Known Windows OS Compatibility Issues . . . 1891
Compatibility Issues with Known Issues at Startup . . . 1892
Conditions Using AdminUser Property . . . 1893
Conflicting Permission Tables . . . 1895
Deferred Execution Custom Action Context . . . 1897
Deprecated API Calls . . . 1898
Deprecated Cluster Automation Server Functionality . . . 1899
Deprecated Distributed File System Tool . . . 1900
Deprecated Nested Windows Installer Packages . . . 1901
Deprecated NETDDE Functionality . . . 1903
Deprecated Proxy Configuration Tools . . . 1904
Deprecated Server Manager Command-Line Tool . . . 1906
Deprecated Windows Library Feature . . . 1907
Drivers with Known Windows Compatibility Issues . . . 1908
Excluded .NET Framework Payload Files . . . 1909
ForceReboot Action . . . 1910
Hard-Coded Paths . . . 1912
Hard-Coded Paths in Script-Based Custom Actions . . . 1914
IIS VBScripting Configuration . . . 1915
Immediate Execution System-Context Custom Actions . . . 1916
Installation to Secure Location . . . 1918
Installers with Known Windows OS Compatibility Issues . . . 1919
Interactive Services in Session 0 . . . 1920
Invalid Component Identifiers . . . 1921
Junction Points . . . 1923
Manifest Files Using Operating System Identifier . . . 1925
Maximum Version of the OS Where This App Was Tested by the Developer . . . 1926
Microsoft Management Console (MMC) Snap-ins Data Execution Prevention . . . 1927
Mixed Per-User and Per-Machine Data . . . 1928
Nested SendTo Menus . . . 1932
Obsolete API Calls . . . 1933
Obsolete File Associations . . . 1935
Operating System Version Conditions . . . 1936
Operating System Version Launch Conditions . . . 1938
Quick Launch Bar . . . 1939
Reboot Pending Launch Condition . . . 1941
Reorganized Start Screen . . . 1942
Restart Manager FilesInUse Dialog . . . 1944
rundll32 Calls (User Account Control) . . . 1945

Self-Update Functionality (User Account Control) . . . 1947
Standard User Changes (User Account Control) . . . 1949
Unmanifested Control Panel (.cpl) Files (User Account Control) . . . 1950
Unmanifested Control Panel Applications (User Account Control) . . . 1952
Unsigned Drivers . . . 1953
Unsigned Executables . . . 1955
Unsigned Windows Installer Database . . . 1956
Unsupported .NET Framework 1.0/1.1 Applications . . . 1957
Unsupported 16-Bit Files . . . 1958
Unsupported 32-Bit Windows Help Files . . . 1960
Unsupported DHTML Editing Control . . . 1961
Unsupported GINA Functionality . . . 1963
Windows Desktop Gadgets . . . 1964
Windows Internet Explorer Protected Mode . . . 1966
Windows Resource Protection Files . . . 1967
Windows Resource Protection Registry Keys . . . 1969
Windows Phone Tests . . . 1970
Application Requires Specific Minimum OS Version (M3001 / M3003 / M3101) . . . 1970
Application Requires VCLibs 11.0 (M3005) . . . 1971
Application Requires VCLibs 12.0 (M3007, M3107) . . . 1971
Application Requires WinJS 1.0 (M3006) . . . 1971
Application Requires WinJS 2.0 or Higher (M3008 / M3108) . . . 1972
Maximum Version of the OS Where This App Was Tested by the Developer (M3002 / M3004 / M3102) . . . 1972
Apple iOS Tests . . . 1973
Application Requires Specific Minimum OS Version (M301 / M401 / M501 / M1001 / M1101 / M1301 / M1401 / M1501 / M1601) 1973
Apple OS X Tests . . . 1973
Application Requires Specific Minimum OS Version (MAC003 / MAC103) . . . 1974
Application Requires 64-bit Processor (MAC005 / MAC105) . . . 1974
Deprecated APIs (MAC004 / MAC104) . . . 1974
Deprecated Frameworks (MAC002 /MAC102) . . . 1975
Deprecated Property List Keys (MAC001 / MAC101) . . . 1975
Removed APIs (MAC007 / MAC107) . . . 1976
Removed Frameworks (MAC006 / MAC106) . . . 1976
Google Android Tests . . . 1976
Application Requires Specific Minimum OS Version (M601 / M701 / M801 / M901 / M1201 / M1701 / M1801) . . . 1977
MSIX Conversion Compatibility Tests . . . 1977
Device Driver (MSIX001) . . . 1977
Windows/NT Service (MSIX002) . . . 1978
Shortcuts (MSIX003) . . . 1978
Custom Action (MSIX004) . . . 1979
Conditionalized Components (MSIX005) . . . 1979
Unsupported .Net Framework Version (MSIX006) . . . 1980
Requires Elevated Privileges (MSIX007) . . . 1980
Application Virtualization Compatibility Tests . . . 1981
Installer Analysis Tests . . . 1981

Application Virtualization Compatibility Installer Analysis Tests . . . 1981
Choosing the Virtual Formats to Display in Test Results . . . 1987
Best Practices Tests . . . 1988
Windows Installer Internal Consistency Evaluators . . . 1988
About ICE43, ICE50, and ICE57 Tests for Shortcuts . . . 1989
Windows Installer Best Practices Tests . . . 1990
ACE04: Components Without Files or Key Paths . . . 1990
ACE05: More Than One Executable File Per Component . . . 1991
ACE06: Executable File Not Marked as Key File of Component . . . 1992
ACE25: Hard-Coded Paths for Custom Action Targets . . . 1992
ACE26: Merge Modules That Are Missing from the Application Catalog . . . 1993
ACE27: Duplicate File Data Without the Required Standard Actions . . . 1994
ACE28: Hard-Coded Paths for Environment Variable Values . . . 1994
ACE29: Hard-Coded Paths for INI File Changes . . . 1995
ACE31: MoveFile Data Without the Required Standard Actions . . . 1996
ACE32: Hard-Coded Paths in Registry Entries . . . 1996
ACE33: RemoveFile Data Without the Required Standard Actions . . . 1997
ACE34: RemoveIniFile Data Without the Required Standard Actions . . . 1997
ACE35: RemoveRegistry Data Without the Required Standard Actions . . . 1998
ACE36: Merge Module Dependencies That Are Missing from the Application Catalog . . . 1999
Microsoft App-V Best Practices Tests . . . 2000
ACE201: Shortcuts with Hard-Coded Paths for Targets . . . 2000
ACE202: Shortcuts with Hard-Coded Paths in Command-Line Arguments . . . 2001
ACE203: Shortcut Targets with Hard-Coded Paths for the Working Directory . . . 2002
ACE208: App-V Packages Without at Least One Shortcut . . . 2003
ACE209: App-V Packages with Shell Extensions . . . 2003
ACE210: App-V Packages with ClickOnce Support . . . 2004
ACE211: App-V Package with DLL Surrogates . . . 2005
ACE212: App-V Packages with Boot Services . . . 2005
ACE213: App-V Packages with OS Integrated Files . . . 2006
ACE214: App-V Packages with Drivers . . . 2006
ACE216: App-V Package with Long .sft File Names . . . 2007
ACE217: App-V Packages with WMI Providers . . . 2007
ACE218: App-V Package with a J2EE Application Server . . . 2008
ACE219: App-V Packages with ASP.NET or IIS Components . . . 2008
ACE220: App-V Packages with Unsupported Applications . . . 2009
Apple Best Practices Tests . . . 2010
M001: Recommended Policy Keys are Specified to Ensure Proper Classification of the Application (Info.plist) . . . 2010
M002: Default Policy Keys are Defined When Device-Specific Versions are Present . . . 2010
M003: Localization Resources are Present and Contain All Required Information . . . 2011
M004: Localization Resources are Present and Contain the Recommended Keys . . . 2011
MAC701: Recommended Property List Keys . . . 2011
MAC702: Code Signature . . . 2011
MAC703: Volume Purchase Program . . . 2012
MAC704: Allows In-app Purchases . . . 2012
Risk Assessment Tests . . . 2012

Mobile Risk Assessment Tests . . . 2012
Windows Mobile Risk Assessment Tests . . . 2013
M4001: Application Requires Telephony . . . 2014
M4002: Application Requires Wi-Fi . . . 2014
M4003: SMS Scheme . . . 2015
M4004: Application Uses a Camera . . . 2015
M4006: Application Uses a Front-Facing Camera . . . 2015
M4008: Application Uses a Video Camera . . . 2016
M4010: Application Uses a Gyroscope . . . 2016
M4011: Application Uses Location Services . . . 2016
M4013: Application Uses a Magnetometer . . . 2017
M4015: Application Uses the Microphone . . . 2017
M4020: Application Uses Peer-to-Peer via Bluetooth . . . 2017
M4021: Application Uses Bluetooth LE . . . 2018
M4030: Application Accesses the Address Book . . . 2018
M4031: Application Supports In-App Purchases . . . 2019
M4037: Application Uses the NFC Card Emulation Feature in the Device . . . 2019
M4040: Application Uses the Device Proximity Sensor . . . 2019
M4045: Application Uses USB Feature . . . 2020
M4046: Application Accesses the Calendar . . . 2020
M4050: Application Uses Internet Access . . . 2020
M4052: Application Uses External Storage . . . 2021
M4053: Application Uses HID . . . 2021
M4054: Application Uses POS . . . 2021
M4055: Application Accesses the Local Document Library . . . 2022
M4056: Application Accesses Local Picture Library . . . 2022
M4057: Application Accesses Local Video Library . . . 2022
M4058: Application Accesses Local Music Library . . . 2023
M4059: Application Supports Enterprise Authentication . . . 2023
M4060: Application Uses Shared User Certificates . . . 2024
M4061: Application Supports Private Network Access . . . 2024
M4062: Application Uses Web Camera . . . 2024
M4063: Application Uses Web Browser . . . 2025
M4064: Application Uses DirectX 11 . . . 2025
M4065: Application Uses Digital Compass . . . 2025
M4066: Application Uses Xbox Service . . . 2026
M4067: Application Uses Push Notification Service . . . 2026
M4068: Application Uses Speech Recognition . . . 2026
M4069: Application Uses Local Ring Tones . . . 2027
M4070: Other App Management . . . 2027
M4071: Wallet . . . 2027
M4072: AllJoyn . . . 2028
M4073: Supports User Profiles . . . 2028
M4074: VOIP Service . . . 2028
M4075: Screen Projection . . . 2029
M4076: Application Uses Device Unlock . . . 2029

M4077: VPN Features 2029
Android Mobile Risk Assessment Tests 2030
M201: Application Requires Telephony 2032
M202: Application Requires Wi-Fi 2032
M203: Application Requires SMS Scheme 2033
M204: Application Uses a Camera 2033
M205: Application Uses an Auto-Focus Camera 2034
M206: Application Uses a Front-Facing Camera 2034
M207: Application Uses a Camera Flash 2035
M208: Application Uses a Video Camera 2035
M209: Application Uses an Accelerometer 2036
M210: Application Uses a Gyroscope 2036
M211: Application Uses Location Services 2037
M212: Application Uses GPS 2037
M213: Application Uses a Magnetometer 2038
M215: Application Uses the Microphone 2038
M220: Application Uses Peer-to-Peer via Bluetooth 2039
M221: Application Uses Bluetooth LE 2039
M230: Application Accesses the Address Book 2040
M231: Application Supports In-App Purchases 2040
M232: Application Supports Social Networking 2041
M235: Application Uses a Low-Latency Audio Pipeline 2041
M236: Application Uses the Consumer IR Capabilities on the Device 2042
M237: Application Uses the NFC Card Emulation Feature in the Device 2042
M238: Application Uses the Barometer in the Device 2043
M239: Application Uses the Device Light Sensor 2043
M240: Application Uses the Device Proximity Sensor 2044
M241: Application Uses the Step Device Detector 2044
M242: Application Requires Landscape Orientation 2045
M243: Application Requires Portrait Orientation 2045
M244: Application is Designed for a Television User Experience 2046
M245: Application Uses USB Feature 2046
M246: Application Accesses the Calendar 2047
M247: Application Uses Device Admin 2047
M248: Application Uses Heart Rate Sensor 2048
M249: Application Uses Relative Humidity Sensor 2048
M250: Application Uses Internet Access 2049
M251: Application Accesses Bookmarks 2049
M252: Application Uses External Storage 2050
M253: Uses Account Manager 2050
M254: Application Uses Kill Background Processes 2051
M255: Application Uses Profile 2051
M256: Application Uses Manage Documents 2052
M257: Application Uses IRTransmitter 2052
M258: Application Uses Body Sensors 2053
M259: Application Accesses Voice Mail 2053

M260: Android Pay 2054
M261: Fingerprint Scanner 2054
M262: Direct Share 2055
M263: VR Mode 2055
Apple Mobile Risk Assessment Tests 2055
M101: Application Requires Telephony 2057
M102: Application Requires Wi-Fi 2058
M103: Application Requires SMS Scheme 2058
M104: Application Uses a Camera 2058
M105: Application Uses an Auto-Focus Camera 2059
M106: Application Uses a Front-Facing Camera 2059
M107: Application Uses a Camera Flash 2059
M108: Application Uses a Video Camera 2060
M109: Application Uses an Accelerometer 2060
M110: Application Uses a Gyroscope 2060
M111: Application Uses Location Services 2061
M112: Application Uses GPS 2061
M113: Application Uses a Magnetometer 2062
M114: Application Uses Gamekit 2062
M115: Application Uses the Microphone 2062
M116: Application Uses OpenGL ES 1.1 2063
M117: Application Uses OpenGL ES 2.0 2063
M118: Application Uses ARMv6 2063
M119: Application Uses ARMv7 2064
M120: Application Uses Peer-to-Peer via Bluetooth 2064
M121: Application Uses Bluetooth LE 2064
M122: Application Uses Safari 2065
M123: Application Runs Only on an iPad 2065
M124: Application Uses Persistent Wi-Fi 2065
M125: Application Runs Only on an iPhone or iPod 2066
M126: Application Can Share Files Through iTunes 2066
M127: Application Can Interface Enumerated External Devices 2067
M128: Application Can Open a Specific File Type 2067
M129: Application Can Save a Specific File Type 2067
M130: Application Can Copy/Paste a Specific File Type 2068
M131: Application Supports Location Tracking 2068
M132: Application Supports Ad Networks 2068
M133: Application Accesses the Address Book 2069
M134: Application Supports In-App Purchases 2069
M135: Application Supports Social Networking 2069
M136: Application Supports User Identity 2070
M137: Application Accesses Local Pictures 2070
M138: Application Accesses the Calendar 2070
M139: Application Uses OpenGL ES 3.0 2071
M140: Application Accesses HealthKit 2071
M141: Application Uses Metal 2072

M142: Application Uses Local Authentication (Touch ID) 2072
M143: Application Uses HomeKit 2072
M144: Application Uses CloudKit 2073
M145: Application Uses Barometer 2073
M146: Application Uses PassKit (ApplePay) 2073
M147: Application Uses App-Extension Custom Keyboard 2074
M148: Application Uses App-Extension Document Picker 2074
M149: Application Uses App-Extension File Provider 2074
M150: Application Uses App-Extension Photo Editing 2075
M151: Application Uses App-Extension Share 2075
M152: Application Uses App-Extension Today 2076
M153: 3D Touch 2076
M154: ARM 64 2076
M155: Siri Voice Assistance (SiriKit) 2077
Desktop Risk Assessment Tests 2077
Mac OS Risk Assessment 2077
D001: App Transport Security (ATS) 2077
D002: ATS Exemptions 2078
D003: Insecure HTTP Connections 2078
D004: Forward Secrecy Exemptions 2079
EOLMAC001: Java Versions Required 2079
EOLMAC002: Java Installer Versions 2080
Windows Risk Assessment 2080
EOLWIN001: Java Versions Required 2080
EOLWIN002: Java Installer Versions 2081
VRAWIN001: Checks for Log4j jar files 2081
Application Conflicts Tests 2082
Package Data Conflicts Tests 2082
ACE02: Identical Components with Different Destinations 2083
ACE03: New or Missing Files in Identical Components 2084
ACE07: Same File in Different Components 2085
ACE08: Identical Components with Different Versions of a File 2086
ACE09: Identical Merge Modules 2087
ACE10: Conflicts in Registry Root, Key, and Name Combinations 2088
ACE12: Files from Merge Modules 2090
ACE13: Shortcut Conflicts 2091
ACE14: Duplicate INI File in Different Components 2092
ACE15: Duplicate ODBC Entries in Different Components 2093
ACE16: Duplicate Services in Different Components 2094
ACE17: Duplicate File Extension-Verb Combinations in Different Components 2095
ACE18: Identical Package Codes for Different Packages 2096
ACE19: Identical Product Codes for Different Packages 2097
ACE20: Identical Upgrade Codes for Different Packages 2098
ACE21: Conflicts Between Entries in the IniFile and File Tables 2098
ACE22: IniFile and File Table Entries for the Same File 2099
ACE23: Duplicate Files with Different Sizes, Versions, or Languages 2100

ACE24: Duplicate Registry Entries with Different Data Types or Values . . . 2102
ACE30: Different Components that Install the Same Key File . . . 2103
Microsoft App-V Conflict Tests . . . 2104
ACE200: Shortcut Location Conflicts . . . 2104
ACE204: App-V Package ID Conflicts . . . 2105
ACE205: Package Name Conflicts . . . 2105
ACE206: File Extension and ProgID Conflicts . . . 2106
ACE207: App-V Conflicts in Root Folder Names . . . 2107
ACE215: App-V Shortcut Name and Version Conflicts . . . 2108
Remote Application Publishing Compatibility Tests . . . 2108
Azure Application Services Tests . . . 2109
MAS0001: Port Bindings . . . 2109
MAS0002: Authentication . . . 2110
MAS0003: Global Assembly Cache (GAC) . . . 2110
MAS0004: IIS5 Compatibility Mode . . . 2111
MAS0005: Application Pools . . . 2111
MAS0006: COM and COM+ Components . . . 2112
MAS0007: ISAPI Filters . . . 2112
MAS0008: Migration of Other Components Like SSL, FTP . . . 2113
Remote Desktop Services Tests . . . 2113
WTS01: Per-User ALLUSERS Property Value for Remote Desktop Services . . . 2114
WTS02: Registry Entries in Per-User Locations . . . 2115
WTS03: Files in Per-User Locations . . . 2115
WTS04: ODBC Data Source Entries in Per-User Locations . . . 2116
WTS05: Per-User Environment Variables . . . 2117
WTS06: Executable Files with Disabled TSAWARE Flags . . . 2118
WTS07: TerminalServer or RemoveAdminTS Conditions . . . 2118
WTS08: 16-Bit Binary Files . . . 2119
WTS09: Administrator Manifest for Binary Files . . . 2120
Analyze Tests Reference . . . 2120
Analyze Resolutions . . . 2121
Resolutions for Operating System Compatibility Tests . . . 2121
Conflict Application Resolution Definitions (CARDs) . . . 2121
Creating Your Own Custom ACE Tests . . . 2123
Types of User-Defined ACEs . . . 2123
Creating User-Defined ACEs . . . 2124
Creating a Custom/Source Only Packages ACE . . . 2124
Creating a Custom/Source and Target Packages ACE . . . 2126
Creating a User Provided DLL-Based ACE . . . 2129
Editing User-Defined ACEs . . . 2131
Deleting User-Defined ACEs . . . 2132
Viewing ACE Metrics . . . 2132
Location of ACE Files . . . 2134

18 Analyzing the Impact of Installing Microsoft Operating System Security Patches . . . 2135

About Microsoft Operating System Patch Files . . . 2136

Importing Microsoft OS Security Patch Files . . . 2137
Identifying and Downloading Microsoft Operating System Patch Files . . . 2137
Importing a Microsoft Operating System Security Patch Into the Application Catalog . . . 2141
Analyzing the Impact of Installing a Microsoft Operating System Patch . . . 2142
Performing Patch Impact Analysis . . . 2143
Viewing Patch Impact Analysis Results . . . 2143
Viewing Patch and Patch Impact Information in Application Catalog . . . 2145
Generating the Patch Report . . . 2148
Reference . . . 2148
Patch Impact Analysis Wizard . . . 2149
Welcome Panel . . . 2149
OS Snapshot Panel . . . 2149
Source Patches Panel . . . 2150
Target Products Panel . . . 2151
Summary Information Panel . . . 2151
Patch Properties Dialog Box . . . 2151
General Tab . . . 2151
Contents Tab . . . 2152
Products Tab . . . 2152

19 Isolating Applications Using Application Isolation Wizard . . . 2153
About Application Isolation Wizard . . . 2154
Isolating Repackaged Setups Using Repackager . . . 2155
Launching the Application Isolation Wizard . . . 2155
Isolation Methods . . . 2155
Assemblies . . . 2156
Manifests . . . 2157
Digital Signatures . . . 2157
Certificates . . . 2157
Import Certificate . . . 2158
Code Signing Technologies . . . 2160
Software Publishing Credentials . . . 2160
Certificate Store . . . 2161
Private Keys . . . 2161
Isolating Applications . . . 2161
Setting Assembly Naming Conventions . . . 2162
Modifying the Default Isolation Recommendations . . . 2162
Filtering File Listings when Manually Configuring Isolation . . . 2163
Servicing Published Shared Assemblies . . . 2164
Application Isolation Wizard Reference . . . 2164
Welcome Panel . . . 2164
Windows Installer File Selection Panel . . . 2165
Isolation Method Panel . . . 2165
Summary Information Panel . . . 2165

Application Isolation Progress Panel . . . 2166
Completing the Application Isolation Wizard Panel . . . 2166
Advanced Options Dialog Box . . . 2166
Manifest Options Tab . . . 2166
Digital Signature Tab . . . 2167
Manifest and Assembly Design Dialog Box . . . 2168
Isolated Components Design Dialog Box . . . 2169
Assembly Properties Dialog Box . . . 2169
Application Manifest Properties Dialog Box . . . 2170
Command-Line Options . . . 2170
Configuration Files . . . 2170
Manifest Examples . . . 2172

20 Ensuring Package Quality Using QualityMonitor . . . 2175

About QualityMonitor . . . 2176
Creating New QualityMonitor Project Files . . . 2176
Opening Existing QualityMonitor Project Files . . . 2177
Working with Test Cases . . . 2177
Running Individual Test Items . . . 2177
Running Multiple Test Items . . . 2178
Adding Test Item Comments . . . 2179
Adding Test Case Comments . . . 2179
Viewing Test Item Details . . . 2180
Clearing Test Case Results . . . 2181
Manually Setting Test Case Status . . . 2181
Manually Setting Test Item Status . . . 2182
Filtering Test Case Data . . . 2182
Deployment Testing . . . 2183
Checking Class IDs . . . 2184
Checking File Associations . . . 2185
Checking Help Files . . . 2185
Checking Prog IDs . . . 2186
Checking Services . . . 2186
Checking Shortcuts . . . 2187
Checking Type Libraries . . . 2187
Checking Manifests . . . 2188
Checking ODBC Data Sources . . . 2188
Checking ODBC Drivers . . . 2189
Specifying Exclusions for Deployment Testing . . . 2190
Lockdown and Runtime Testing . . . 2191
Performing Lockdown and Runtime Tests . . . 2192
Performing Lockdown and Runtime Tests Under a Different User Account . . . 2193
Running Lockdown and Runtime Tests in Restricted Environments . . . 2193
Performing Isolation Tests . . . 2194
Filtering Results of Lockdown and Runtime Tests . . . 2195

Using MSI Doctor to Verify Package Deployment Status 2195
View Product, Feature, or Component Deployment Status Properties 2197
Verify Product, Feature, or Component Data 2197
Install or Configure Products or Features 2198
Reinstall Features 2199
Reinstall Components 2200
Creating Custom Test Cases 2200
Test Reports 2202
Running QualityMonitor from the Command Line 2202
QualityMonitor Reference 2203
Menus and Toolbar 2204
QualityMonitor Interface 2205
Dialog Boxes 2205
About QualityMonitor Dialog Box 2206
Add Exclusions Dialog Box 2206
Component Properties Dialog Box 2206
Feature Properties Dialog Box 2207
Install or Configure Feature Dialog Box 2208
Install or Configure Product Dialog Box 2208
Installed Data Dialog Box 2209
Open QualityMonitor Project Dialog Box 2210
Options Dialog Box 2210
General Tab 2210
Exclusions Tab 2211
Product Properties Dialog Box 2212
Re-install Product/Feature Dialog Box 2213
Runtime Test Filters Dialog Box 2214
Test Item Information Dialog Box 2214
Test Progress Dialog Box 2214
Test Result Dialog Box 2215
Views 2215
Welcome to QualityMonitor View 2216
Product Information View 2216
Test Cycle Summary View 2216
Deployment Tests View 2217
Class IDs View 2218
File Associations View 2218
Help Files View 2218
Prog IDs View 2218
Shortcuts View 2218
Type Libraries View 2218
Manifests View 2219
ODBC Data Sources View 2219
ODBC Drivers View 2220
Services View 2220
Lockdown and Runtime Tests View 2220

Runtime Execution Details View *2221*
Files View *2221*
Folders View *2223*
Registry Entries View *2224*
Isolation Tests View *2225*
User-Defined Tests View 2225
Test Case View *2226*
Deployment Status View 2226

21 Distributing Applications and Packages 2229

Distributing Applications Using the Distribution Wizard 2231

Distributing Packages Using the Package Distribution Wizard 2240

Creating Administrative Installations for Packages 2241
Distributing Packages to FTP Servers 2242
Preparing for Altiris 6.5 Distribution 2242
Distributing Packages to Network Locations 2243
Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager) 2244
Preparing for ZENworks Configuration Management Distribution 2246
Deploying InstallScript MSI Installations 2250

Reference 2251

Distribution Wizard 2252
Choose Applications Panel 2252
Target Server Details Panel 2254
Destination Group Panel 2255
Summary Panel 2257
Distributing Panel 2258
Package Distribution Wizard 2258
Welcome Panel 2259
Distribution Type Panel 2259
Administrative Install Panel 2260
Connect to a ConfigMgr Server Panel 2260
Select Destination Folder 2261
Select Group 2262
FTP Location Panel 2263
Altiris Integration Panel 2263
Altiris XML Template *2265*
Network Location Panel 2266
Package Information Panel 2266
Distribution Summary Panel 2267
Distribution Output Panel 2267
Distribution Wizard for ZENworks Configuration Management 2268
Welcome Panel 2268
Login Panel 2268
Windows Installer Package Information Panel 2269
Bundle Creation Options Panel 2273
Bundle Information Panel 2274

Summary Panel . . . 2275
Publishing Process Panel . . . 2275

22 Generating and Viewing Reports . . . 2277

Generating and Viewing AdminStudio Reports . . . 2278
Viewing Package Reports . . . 2279
Searching for a Package on the Search Packages Page . . . 2279
Information Included in Package Reports . . . 2282
Package Summary Information View . . . 2283
Files View . . . 2285
Registry View . . . 2286
Shortcuts View . . . 2287
ODBC Drivers View . . . 2287
ODBC DS View . . . 2288
Extended Attributes View . . . 2289
Validation View . . . 2290
Conflicts View . . . 2291
History View . . . 2292
Dependencies View . . . 2293
Properties View . . . 2294
Navigating Through a Package Report . . . 2295
Archiving a Package Report . . . 2297
Exporting a Package Report . . . 2299
Generating a Custom SQL Query Report for AdminStudio . . . 2299
Generating a Custom Stored Procedure Report for AdminStudio . . . 2304
Specifying Predefined Parameters in a Custom Stored Procedure Report . . . 2306
Viewing AdminStudio Application Catalog Reports . . . 2308

Generating and Viewing Workflow Manager Reports . . . 2309
Generating Standard Reports . . . 2310
Generating a Projects Report . . . 2312
Generating a Workflow Requests Summary Report . . . 2312
Generating a Request Detail Report . . . 2313
Generating a Project SLA Report . . . 2313
Generating a Workflow Phases SLA Report . . . 2315
Generating a Workflow Steps SLA Report . . . 2316
Creating Custom Reports . . . 2316
Creating a Custom Report . . . 2317
Creating an Activity Report . . . 2318
Generating a Custom SQL Query Report . . . 2322
Generating a Custom Stored Procedure Report . . . 2323
Wildcard Support in Reports SQL Queries . . . 2325
Sample SQL Queries Used to Generate Project and Workflow Request Reports . . . 2326
Exporting Report Data from Reports . . . 2327

Reports Reference . . . 2327
All Reports Page . . . 2327
Standard Reports . . . 2330

Projects Report *2330*
Workflow Requests Summary Report *2332*
Request Detail Report *2333*
Project SLA Report *2335*
Workflow Phases SLA Report *2337*
Workflow Steps SLA Report *2338*
Search Packages Page 2339
Application Catalog Reports Page 2341
Viewing the AdminStudio Application Catalog Reports 2342
Exporting a Report in PDF, Excel, or Word Format 2343
Package Report 2343
Reports Wizard 2348
Select Stored Procedure Panel 2349
Select Report Objects Panel 2350
Select Report Fields Panel 2350
Define Report Filters Panel 2351
Select Template Data Panel 2353
Enter SQL Query Panel 2354
Specify General Information Panel 2355
Save and Preview Report Panel 2356

23 AdminStudio PowerShell Cmdlets 2359

About the AdminStudio PowerShell Cmdlets 2359
Setting Up AdminStudio Module in PowerShell 2360
Example Script to Create Application Catalog, Import Packages, and Perform Testing 2361
PowerShell Cmdlets Reference 2365
Add-ASKeywords 2368
Add-ASPackageForConversion 2369
Get-ASAppVShortcuts 2371
Get-ASApplicationDetails 2372
Get-ASApplicationID 2372
Get-ASAppPortalCategories 2373
Get-ASAppPortalTemplates 2374
Get-ASCatalogItem 2375
Get-ASConfigPlatform 2386
Get-ASApplicationDeploymentSummary 2387
Get-ASDeploymentSystemPackageTree 2388
Get-ASDistributionProperty 2389
Get-ASKeywords 2391
Get-ASPackage 2391
Get-ASPackageTestSummary 2392
Get-ASPackageFeedDetails 2393
Get-ASProperty 2394
Get-ASTestDetails 2395
Get-ASTestState 2395
Get-ASOptionProperty 2396

Get-ASVirtualReadiness . . . 2397
Invoke-ASAppVBulkUpgrade . . . 2400
Invoke-ASConvertFolder . . . 2401
Invoke-ASConvertPackage . . . 2402
Invoke-ASConvertPackageEx . . . 2403
Invoke-ASImportAppFromDeploymentSystem . . . 2404
Invoke-ASImportPackage . . . 2405
Invoke-ASWrapPackage . . . 2407
Invoke-ASPackageCustomize . . . 2408
Invoke-ASPackageFeedSearch . . . 2408
Invoke-ASPackageFeedDownload . . . 2409
Invoke-ASPackageFeedSync . . . 2410
Invoke-ASPublish . . . 2411
Invoke-ASUpdatePackage . . . 2412
Invoke-ASSignPackage . . . 2413
New-ASCatalog . . . 2414
New-ASDistributionConnection . . . 2416
New-ASDistributionConnectionEx . . . 2417
New-ASCreateSoftwareTag . . . 2421
Remove-ASApplication . . . 2422
Remove-ASGroup . . . 2423
Remove-ASPackage . . . 2424
Resolve-ASPackage . . . 2425
Set-ASAppVShortcutProperty . . . 2425
Set-ASCatalog . . . 2426
Set-ASConfigPlatform . . . 2427
Set-ASDistributionProperty . . . 2431
Set-ASProperty . . . 2432
Set-ASSoftwareRepositoryState . . . 2439
Set-ASTestState . . . 2441
Set-ASOptionProperty . . . 2442
Set-ASSoftwareTagProperties . . . 2443
Start-ASConversion . . . 2444
Test-ASConflicts . . . 2445
Test-ASPackage . . . 2446
New-ASPackageRequest . . . 2447

24 AdminStudio REST APIs . . . 2449

About the AdminStudio REST APIs . . . 2449

Setting Up Connection to Run REST APIs . . . 2450

AdminStudioHost.exe.config file . . . 2450

REST APIs Reference . . . 2450

Adding App Portal Catalog Item Keywords to the Application Catalog . . . 2452
Getting Package Deployment Type . . . 2453
Get-ASTestDetails . . . 2455
Get-ASKeywords . . . 2456

Get-ASTestState 2457
Get-ASApplicationID 2458
Get-ASApplicationDetails 2459
Get-ASVirtualReadiness 2461
Get-ASPackageTestSummary 2462
Get-ASProperty 2463
Get-ASCatalogItem 2465
Get-ASPackageCodes 2467
Get Status of a Request 2469
Invoke-ASPublish 2471
Invoke-ASImportPackage 2473
Invoke-ASAppVUpgrade 2474
New-ASPackageRequest 2475
New-ASCatalog 2478
New-ASDistributionConnection 2480
Remove-ASApplication 2482
Remove-ASGroup 2483
Remove-ASPackage 2484
Resolve-ASPackage 2485
Set-ASTestState 2487
Set-ASProperty 2487
Test-ASPackage 2489
Set-ASCatalog 2492

Index 2495

1

AdminStudio 2022 R2 Help Library

AdminStudio powers an enterprise's daily application readiness process for packaging, compatibility testing, and automated distribution. Using AdminStudio, systems administrators can rapidly prepare error-free applications to deploy into their enterprise environment through a structured Microfocus process built on application management best practices.

You can use AdminStudio tools to repackage applications and convert packages to virtual formats. Then, after importing all of your of physical, virtual, and mobile applications into the Application Manager, you can thoroughly test them to ensure that they are ready for deployment across your enterprise. The final step is to perform automated distribution directly from the Application Manager.

The AdminStudio user documentation contains information about the functionality and features of all of the components of AdminStudio, and is presented in the following sections:

Table 1-1 • AdminStudio Help Library

Topic	Content
Getting Started with AdminStudio	Describes how to use the AdminStudio Start Page tabs—which provide process information on how to perform key tasks using AdminStudio tools—to quickly get started evaluating and using AdminStudio.
Using the AdminStudio Interface	Describes the AdminStudio Interface, the central application for AdminStudio. From it, you can launch the AdminStudio tools, create process templates and projects, use AdminStudio Enterprise Server tools, and connect to and create Application Catalog.
Managing Accounts and Directory Services	Explains how to create an account for each person that you want to have access to AdminStudio, and how to import users or groups of users from a directory service. Also explains how to set up the AdminStudio account, domain account, single sign-on, and guest account login methods.
Managing Roles and Permissions	Explains how to create and edit roles to manage access to AdminStudio functionality.

Table 1-1 • AdminStudio Help Library (cont.)

Topic	Content
Managing Applications and Application Catalog Databases	Explains how to use **Application Catalog** to import applications into the Application Catalog, organize them, and set up automatic import. It also explains how to manage application data, upgrade App-V packages from v4.x to 5.0, and how to convert packages to virtual formats using default settings.
Repackaging Legacy Installations Using the Repackaging Wizard	Explains how to use Repackager's Repackaging Wizard to convert existing legacy installations into Windows Installer (MSI) packages.
Converting Legacy Installations Using the Repackager Interface	Explains how to use the Repackager interface to create and modify Repackager project files, and how to build those files into InstallShield Editor projects or Windows Installer packages.
Performing Virtualization and Repackaging Using the Automated Application Converter	Explains how to use the Automated Application Converter to examine a group of setups and perform automated virtualization of those setups (including performing automated repackaging of those setups that require it).
Using the Virtual Package Editor	Explains how to use the Virtual Package Editor to edit App-V packages and perform tasks such as customizing your App-V applications, resolving virtualization Best Practice issues and application conflicts, and fixing run-time problems.
Using the MSIX Editor	Explains how to use the MSIX Editor, a tool that enables you to edit MSIX packages, create Modification packages and perform various other tasks on MSIX packages.
Creating Customized Virtual Applications	Explains how to use the InstallShield Virtualization Assistants to create customized virtual applications in the Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual application formats.
Customizing and Authoring Installations Using InstallShield	Describes how to use **InstallShield Editor** to create setup packages that utilize Windows Installer technology, while harnessing the flexibility provided by InstallScript. Also explains how to use InstallShield to create virtual applications.
Customizing Installations Using Tuner	Explains how to use Tuner to create a transform file to add to, modify, or remove information from a Windows Installer package.
Using Analyze to Perform Package Testing	Describes how to perform operating system compatibility, best practices, risk assessment, application conflict, remote application publishing compatibility, and application virtualization compatibility testing on packages in the Application Catalog using Analyze.
Analyze Tests	Describes the Analyze tests used to perform operating system compatibility, best practices, risk assessment, application conflict, remote application publishing compatibility, and application virtualization compatibility testing.

Table 1-1 • AdminStudio Help Library (cont.)

Topic	Content
Analyzing the Impact of Installing Microsoft Operating System Security Patches	Explains how to import Microsoft application patches into the Application Catalog and thoroughly test the impact they will have on your environment before they are deployed.
Isolating Applications Using Application Isolation Wizard	Explains how to use the **Application Isolation Wizard** to solve component versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.
Ensuring Package Quality Using QualityMonitor	Explains how to use **QualityMonitor** to run a series of built-in tests to installed Windows Installer–based products, helping to ensure they run correctly, especially in a locked-down environment.
Distributing Applications and Packages	Explains how to use the Distribution Wizard to publish applications to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, and Workspace ONE Server. It also explains how to use the Package Distribution Wizard to distribute packages to a variety of distribution systems.
Generating and Viewing Reports	Explains how to use the **Reports** to report on or view all of the information regarding the applications in your Application Catalog from a single location.
Setting Up AdminStudio Module in PowerShell	Explains how to use the AdminStudio PowerShell Cmdlets to integrate your existing .NET applications or scripting environments like Microsoft PowerShell with AdminStudio.

What's New in AdminStudio 2022 R2 | 24.0

This section lists the new features that are included in AdminStudio 2022 R2:

- MSIX Editor Enhancements
- Host AdminStudio Catalog on Azure SQL
- Automate Package Feed Module Trial Packages
- Support for Microsoft ConfigMgr 2207
- InstallShield 2022 R1

Note • *Detailed information on the new features in AdminStudio 2022 R2 is also available in the AdminStudio Release Notes on the Flexera product documentation site: https://docs.flexera.com*

MSIX Editor Enhancements

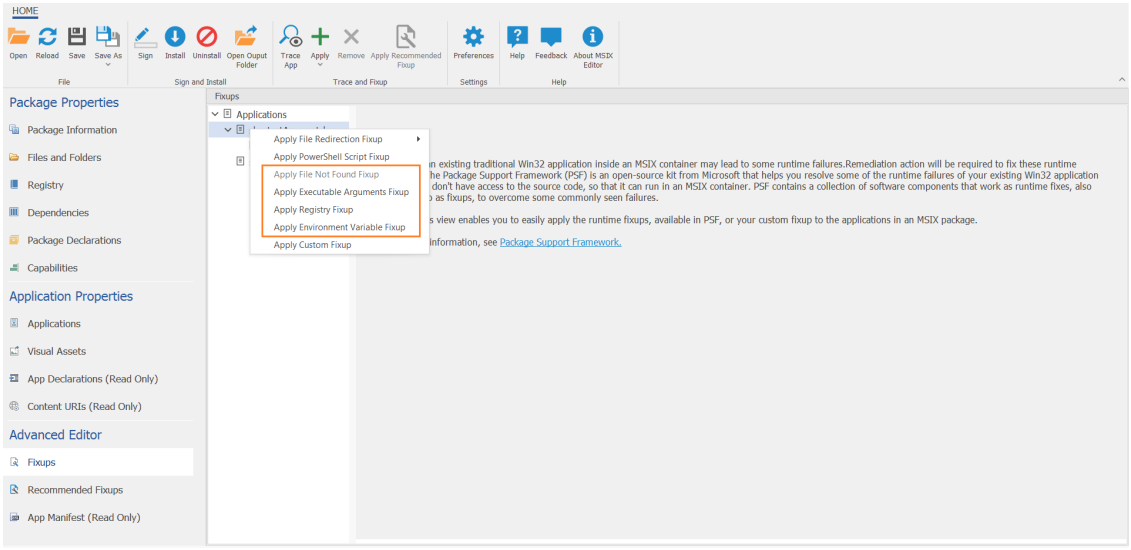
The following improvements have been added to the MSIX Editor.

- Support for New Additional Fixups
- Editing of Visual Assets
- Support for Non-EXE Applications
- Fixup Suggestions

Support for New Additional Fixups

In AdminStudio 2022 R2, you can apply the following new fixups to an application in an MSIX Package to resolve certain runtime failures.

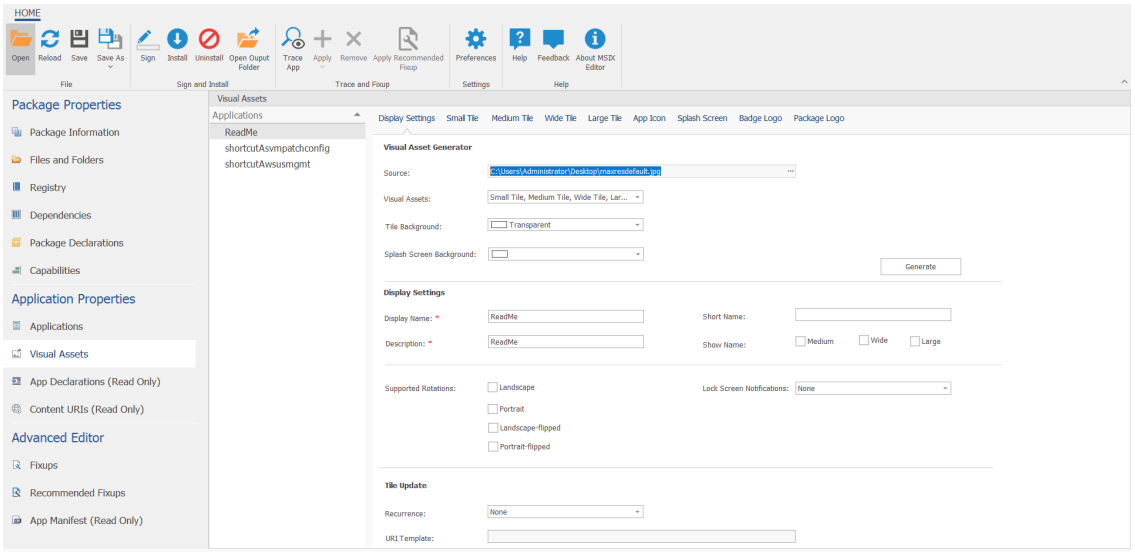
- File Not Found Fixup
- Executable Arguments Fixup
- Registry Fixup
- Environment Variable Fixup



For more information, see Applying Fixups.

Editing of Visual Assets

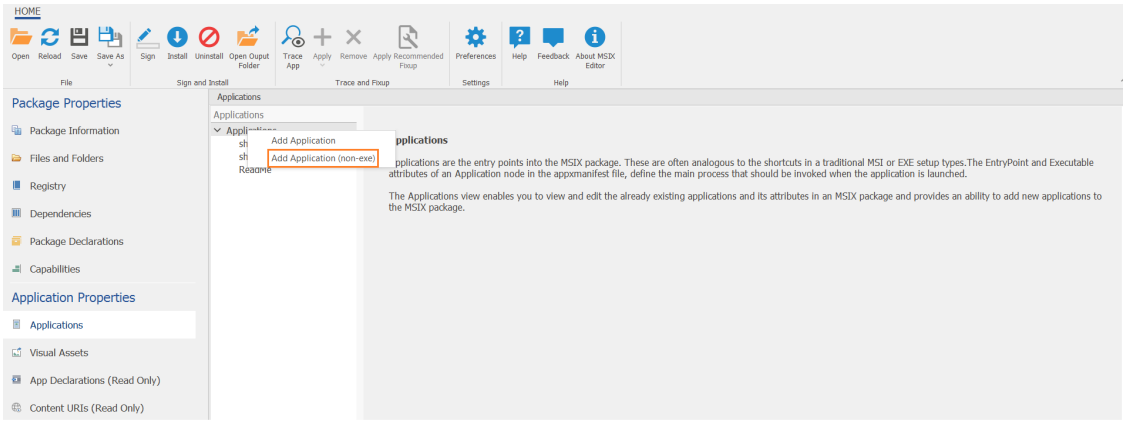
The **Visual Assets** view in MSIX Editor is editable now. You can modify the existing visual assets of an MSIX package or generate a new set of visual assets.



Support for Non-EXE Applications

You can create applications in an MSIX package pointing to a non-exe file. The following non-exe file format are supported to create an application: .bat, .cmd, .txt, .html and .docx.

To create a non-exe application, right-click on the **Applications** node and then select **Add Application (non-exe)** from the context menu.



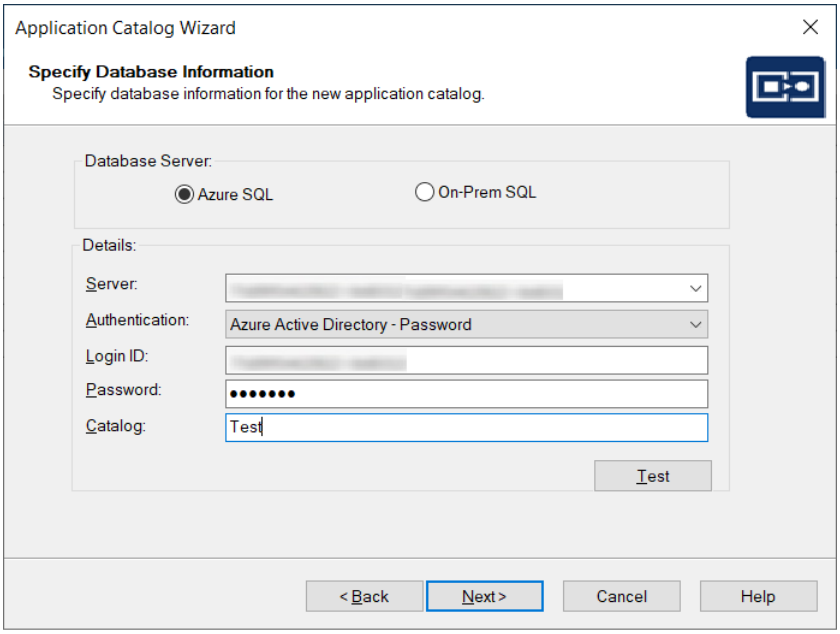
Fixup Suggestions

Tracing an MSIX package in the MSIX Editor, will now suggest fixups based on the runtime error encountered while tracing. The recommendation fixups will be displayed in the new **Recommended Fixups** view.

For more information, see Applying Recommended Fixup.

Host AdminStudio Catalog on Azure SQL

You can now create AdminStudio application catalog on Azure SQL. Select **Azure SQL** option in the Application Catalog Wizard and provide the required details to create a catalog on Azure SQL.



Automate Package Feed Module Trial Packages

The Package Feed Module trial packages are now added to the Backlog tab to enable automation. Subscribe the PFM trial packages for automation and configure actions and a schedule to automatically download and process packages when a new version is made available in the Package Feed Module.

For more information, see Automate Package Feed Module Trial Packages.

Support for Microsoft ConfigMgr 2207

In AdminStudio 2022 R2, you can distribute the following deployment types to Microsoft ConfigMgr 2207:

- Windows Installer (MSI)
- MSIX Package
- Legacy installer (EXE)
- PowerShell wrapped packages (PS1)
- App-V (4.x and 5.0)
- Apple iOS (binary file and public store)
- Google Android (binary file and public store)

InstallShield 2022 R1

AdminStudio 2022 R2 includes InstallShield 2022 R1, that has the following new features and enhancements:

- Support for Custom Declarations in MSIX
- Visual Studio 2022 Integration
- Minor Upgrade Support for Chained MSI
- Validation of MSI

For more information on these new features, see the InstallShield 2022 R1 Release Notes.

AdminStudio Editions and Components

AdminStudio is available multiple editions to meet the needs of every organization:

- AdminStudio Full Editions
- AdminStudio Limited Editions

AdminStudio Full Editions

AdminStudio 2022 R2 is available in Professional and Enterprise Editions.

- Feature Breakdown by Edition
- Documentation Edition Notes

Feature Breakdown by Edition

The following table lists the tools and features available in each of AdminStudio's editions.

Edition	Feature Type	Tools	Functionality
PROFESSIONAL	**General**	**Application Manager / Home Tab**	• Manage applications in an Application Manager database. • Manage a package's ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 or ConfigMgr 2002) and Symantec Altiris Client Management Suite deployment data. • View an application's ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 or ConfigMgr 2002) deployment status. • Perform advanced ISO tag file creation, editing, and storage. • **Package Automation with Package Feed Module**—You can use the Package Feed Module option of the Import Wizard to subscribe to up to 10 applications for bulk execution.
		Application Manager / Analyze Tab	Perform tests in the following categories: • Windows Installer Internal Consistency Evaluators • Windows Installer Best Practices • Application Conflicts Test and fix one package at a time
		Distribution Wizard	Publish applications to ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 or ConfigMgr 2002) and Symantec Altiris Management Server.
		OS Snapshot Wizard	Capture basic operating system configuration in an OS Snapshot, which can be imported into the Application Manager to check for potential OS conflicts
		QualityMonitor	Perform Windows Installer testing, including testing in a locked down environment

Edition	Feature Type	Tools	Functionality
PROFESSIONAL (Continued)	General (Continued)	**Automated Application Converter** (Single Application Version) **Conversion Wizard** (Single Application Version)	Automatically repackage a legacy package (.exe) into a Windows Installer package (.msi) Repackage one package at a time
		Test on Virtual Machine Wizard	Automatically launch a specified virtual machine and install a selected Windows Installer (.msi), App-V package (.appv), or installation executable (.exe) package for testing.
		Repackager	Repackage applications into Windows Installer format Perform basic ISO tagging, including creation of tag files
		Package Distribution Wizard	Prepare packages for distribution
		InstallShield 2022 R1 (Professional Edition)	Customize Windows Installer packages by either directly editing them or by creating transforms
		Tuner	Customize Windows Installer packages by creating transforms
		Application Isolation Wizard	Resolve component versioning conflicts
	Application Virtualization	Enhancements to **Application Manager / Home Tab**	Import virtual packages into Application Manager View virtual package data in Application Manager Manage ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 or ConfigMgr 2002) deployment data for App-V 4.x and 5.1 packages Manage Citrix XenApp Server deployment data for Citrix XenApp profiles and App-V 4.x packages Manage Symantec Altiris Client Management Suite deployment data for Symantec Workspace and VMware ThinApp packages Manage App-V Server deployment data for App-V 4.x and 5.1 packages

Edition	Feature Type	Tools	Functionality
PROFESSIONAL **(Continued)**	**Application Virtualization** (Continued)	Enhancements to Application Manager / Analyze Tab	Test packages for compatibility to be virtualized to App-V, ThinApp, and XenApp formats Test App-V packages for best practices Test App-V packages for conflicts with other packages
		Enhancements to **Distribution Wizard**	Publish applications containing App-V 4.x and 5.1 packages to Microsoft App-V Server Publish applications containing App-V packages to ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 or ConfigMgr 2002) and Citrix XenApp Server Publish applications containing Citrix XenApp profiles and App-V 4.x packages to Citrix XenApp Server Publish applications containing Symantec Workspace and VMware ThinApp packages to Symantec Altiris Client Management Suite Server
		Automated Application Converter (Single Application Version) **Conversion Wizard (Single Application Version**	Convert a package to a virtual application in the following formats: ● Microsoft App-V (4.x and 5.1) ● Citrix XenApp ● VMware ThinApp (4.x and 5.x) Convert one package at a time
		Virtual Package Editor	Edit App-V packages
		MSIX Editor	Edit MSIX packages
		Microsoft App-V Assistant **ThinApp Assistant** **Citrix Assistant**	Create a customized virtual package from an InstallShield project

Edition	Feature Type	Tools	Functionality
PROFESSIONAL (Continued)	**Application Compatibility Testing**	Enhancements to **Application Manager / Analyze Tab**	Test packages for compatibility with the following operating systems: ● Windows 8.1 (32-bit and 64-bit) ● Windows 10 1909 (32 bit and 64 bit) ● Windows 10 1809 (and 2019 LTSC) (32 bit and 64 bit) ● Windows 10 2004 (32 bit and 64 bit) ● Windows 10 20H2 (32 bit and 64 bit) ● Windows 10 21H1 (32 bit and 64 bit) ● Windows 10 21H2 (32 bit and 64 bit) ● Windows 11 21H2 (64 bit) ● Windows Server 2012 and 2012 R2 ● Windows Server 2016 ● Windows Server 2019 On the **Operating System Compatibility** tab of the Analyze **Deployment Type View**, you can see detailed data for only the last package tested; for all other packages in the Application Manager, this tab is blank (even if the package has been previously tested). Ability to display Microsoft Application Compatibility Toolkit (ACT) database test results on **ACT Summary** tab of the **Analyze Deployment Type View**.

	Mobile App and macOS Support	Enhancements to **Application Manager / Home Tab**	● Import of the following macOS desktop applications into the Application Manager:
			  ● Apple installer package (.pkg file)
			  ● Apple disk image (.dmg file)
			  ● Mac App Store app (public store link)
			● Import of the following mobile app formats into the Application Manager:
			  ● Apple iOS mobile apps (local and public store link)
			  ● Google Android mobile apps (local and public store link)
			  ● Microsoft Windows Store mobile apps (local and public store link)

Edition	Feature Type	Tools	Functionality
PROFESSIONAL (Continued)	**Mobile App and macOS Support** (Continued)	Enhancements to **Application Manager / Home Tab** (Continued)	• Ability to import iOS Enterprise Policy Configuration files, view their settings, and determine the policy compatibility of iOS mobile apps. • Ability to view iOS and Android mobile app reporting on feature use, device compatibility, and OS compatibility. • Ability to customize Apple Installer Package PKG installer settings • Ability to view deployment data for Windows Store mobile apps, including detection methods and framework customizations • Ability to manage AirWatch Server deployment data for both Apple iOS and Google Android mobile apps (local and public store link) • Ability to view and modify Casper deployment settings for macOS desktop applications
		Enhancements to **Application Manager / Analyze Tab**	• Test Apple iOS mobile apps for best practices • Test Apple iOS, Microsoft Windows, and Google Android mobile apps for risk assessment • Test Apple iOS, Microsoft Windows, and Google Android mobile apps for operating system compatibility • Test macOS desktop applications (.pkg, .dmg, and Mac App Store apps) for operating system compatibility and best practices

Edition	Feature Type	Tools	Functionality
PROFESSIONAL (Continued)	**Mobile App and macOS Support** (Continued)	Enhancements to **Distribution Wizard**	Ability to publish applications containing the following mobile app formats to ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 R2 or ConfigMgr 2002) and AirWatch Server: ● Apple iOS mobile apps (local and public store link) ● Google Android mobile apps (local and public store link) Ability to publish applications containing the following mobile app format to ConfigMgr (Formerly Called as Software Center Configuration Manager) (2012 R2 or ConfigMgr 2002): ● Windows Store (local and public store link) ● Microsoft UWP app packages (.appx) Ability to publish applications containing the following package formats to JAMF Casper Suite: ● Mac App Store app (public store link) ● Apple installer package (.pkg file) ● Apple disk image (.dmg file)

Edition	Feature Type	Tools	Functionality
ENTERPRISE	Same as **Professional Edition**, plus:		
	General	**InstallShield 2022 R1** (Premier Edition instead of Professional Edition)	Advanced customization of Windows Installer packages by either directly editing them or by creating transforms
		Enhancements to **Application Catalog** / **Home** Tab	**Package Automation with Package Feed Module**—You can use the Package Feed Module option of the Import Wizard to subscribe to an unlimited number of applications for bulk execution.
		Application Manager / Reports Tab	Advanced reports including detailed summary and dashboard reports on Analyze test results, package data, and deployment information
		Platform API	Use to integrate your existing .NET applications or scripting environments like Microsoft PowerShell with AdminStudio
		Software Repository	Secure storage system for AdminStudio package data, including version management
		Reports (Web Tool)	Generate reports on packages stored in the Application Manager, including reports using custom SQL queries
		Security Console (Web Tool)	Manage AdminStudio user accounts and directory services Manage AdminStudio roles and permissions
		Automated Application Converter (Multiple Application Version) **Conversion Wizard** (Multiple Application Version)	Automatically repackage legacy packages (.exe) into Windows Installer packages (.msi) Ability to perform automated repackaging of multiple packages at a time
	Application Virtualization	**Automated Application Converter** (Multiple Application Version) **Conversion Wizard** (Multiple Application Version)	Ability to perform automated conversion of multiple packages at a time

Edition	Feature Type	Tools	Functionality
ENTERPRISE (Continued)	**Application Virtualization** (Continued)	Enhancements to **Application Manager / Reports**	Includes the Application Virtualization Compatibility Dashboard report The Application Readiness Dashboard includes an Application Virtualization Compatibility summary chart and App-V Best Practices and App-V Conflicts test results summary charts.
	Application Compatibility Testing	Enhancements to **Application Manager / Analyze Tab**	Ability to test and fix multiple packages or groups of packages simultaneously Ability to view package-level test details for **Operating System Compatibility** tests for all packages in the Application Manager, not just the last one tested
		Enhancements to **Application Manager / Reports**	Display of Microsoft ACT database test results on the Reports tab
		Enhancements to **Platform API**	Ability to use the `Test-ASPackage` and `Resolve-ASPackage` Platform API commands to perform batch package testing and issue resolution.

Documentation Edition Notes

Documentation on features that are only available in specific Editions or add-on packs include the following notes:

Table 1-2 • Documentation Edition Notes

Edition/Add-On Pack	Note
Professional	***Edition*** • *This feature is included with AdminStudio Professional and Enterprise Editions.*
Enterprise	***Edition*** • *This feature is included with AdminStudio Enterprise Edition.*

AdminStudio Limited Editions

AdminStudio 2022 R2 | 24.0 is also available in the following limited editions.

Table 1-3 • AdminStudio Limited Editions

Edition	Description
Microfocus ZENworks Editions	The AdminStudio ZENworks Edition for Microfocus ZENworks Configuration Management (ZCM) 10 and 11 customers includes all of the features of AdminStudio Edition, the basic version of the AdminStudio solution. ***Note*** • *For more information, see About AdminStudio ZENworks Edition.*

About AdminStudio ZENworks Edition

Information about the ZENworks Edition is presented in the following sections:

- ZENworks Edition for ZENworks 10 and 11 Users
- ZENworks Server Connection Requirement

ZENworks Edition for ZENworks 10 and 11 Users

The AdminStudio 2022 R2 ZENworks Edition for Microfocus ZENworks Configuration Management (ZCM) 10 and 11 customers includes all of the features of AdminStudio Edition, the basic version of the AdminStudio solution. AdminStudio Edition provides a cost-effective way for software packagers to migrate applications to Windows Installer. It enables control over MSI packaging, customization, and distribution activities, helping organizations rapidly and reliably package and deploy applications.

ZENworks Server Connection Requirement

The first time you launch an AdminStudio ZENworks Edition tool, you will be prompted to log in to a ZENworks eDirectory server or a ZENworks Configuration Management server. After a successful login, you will not be prompted to login again.

How to Upgrade AdminStudio Limited Edition to Professional or Enterprise Editions

An upgrade feature has been built-in to AdminStudio that allows you to activate features in a higher edition or to add optional Packs without re-installing the application. You just need to enter a Activation Code for the upgrade that you purchased, and the features of that Edition are immediately unlocked and are available to you.

To upgrade AdminStudio, contact AdminStudio Sales and purchase a Activation Code for the desired Edition, and then follow the instructions in Upgrading Your Product Edition.

Activating AdminStudio

When you launch AdminStudio or one of its tools for the first time, you are notified that you are using a time-limited trial version, and you are given the opportunity to evaluate the product or to activate it by entering a valid Activation Code for an AdminStudio Edition.

Task ***To activate AdminStudio:***

1. Install AdminStudio, as described in the *AdminStudio Installation Guide*.
2. Launch AdminStudio or one of its tools. A dialog box opens, stating that you are using a time-limited trial version.
3. Select **Activate or Purchase AdminStudio** and click **Next**. The **AdminStudio Product Activation** dialog box opens, prompting you to enter a activation code.
4. Enter the activation code of the edition you purchased and click the **Activate** button. After a few seconds, you will receive a message that activation was successful.
5. Click **Finish**. AdminStudio will launch.

Ports Used in Activation

AdminStudio product activation uses ports 80 (HTTP), 443 (HTTPS), and 8443. If these ports are locked down or if you do not have an available internet connection, you can configure licensing for AdminStudio using one of the following alternative methods:

- **Offline activation**—You can perform offline activation using email.
- **Self-hosted licensing**—Your organization can choose to purchase self-hosted licenses of AdminStudio. A self-hosted license for AdminStudio requires a Flexera-generated license file for the machine on which you install AdminStudio, but it does not require activation.

Silent Activation

The AdminStudio installer was created using an InstallShield “Suite” project type, which means that the procedure for installing AdminStudio silently is different from previous releases, which were created using a “Basic MSI” project type.

To install AdminStudio 2022 R2 silently, you need to use the `ASCommandLine` property to pass MSI parameters to the AdminStudio installer (`AdminStudio2021R2SP1.exe`), along with the `/silent` switch:

```
AdminStudio2021R2SP1.exe /silent ASCommandLine="[Parameters]"
```

For example:

```
AdminStudio2021R2SP1.exe /silent ASCommandLine="TRANSFORMS=MyTransform.mst"
```

Note • *You cannot use the `ASCommandLine` property to pass the `ISInstallDir`, `SharedInstallDir`, or `PRODUCTID` command line parameters to the AdminStudio installer; these must be specified explicitly.*

For more information, see *Installing AdminStudio Silently Via Command Line* in the *AdminStudio 2022 R2 Installation Guide*.

Deactivating AdminStudio to Enable Activation on a Different Machine

AdminStudio node-locked licenses are single-machine, single-user licenses according to the end-user license agreement. This means that they can be activated on one machine and accessed by one user.

If moving a license to a new machine is required, the license must first be deactivated from the previous machine before it can be activated on the new machine.

This section includes steps for both online and offline methods of deactivation.

Online Deactivation

To perform deactivation when you have an Internet connection, perform the following steps:

Task ***To perform deactivation via internet:***

1. If you have already uninstalled AdminStudio from the previous machine, reinstall it.
2. Open the **Command Prompt** application.
3. Change the directory to the following path:

   `[AdminStudio_Installation_Directory]\Common`

   For example:

   `C:\Program Files (x86)\AdminStudio\2021\Common`
4. At the command prompt, execute the following command to deactivate the AdminStudio license on this machine:

   `TPSconfig /return`

   Note • *It is recommended that you allow incoming and outgoing traffic to and from `*.flexera.com` and `*.installshield.com`, as well as making sure that ports 80, 443, and 8443 are open. The Activation Server URL is:*

   `activationservice.installshield.com`
5. Install AdminStudio on the new machine. You will be permitted to activate it.

Offline Deactivation

To perform deactivation when you do not have an Internet connection, perform the following steps:

Task ***To perform deactivation via email:***

1. If you have already uninstalled AdminStudio from the previous machine, reinstall it.
2. Open the **Command Prompt** application.
3. Change the directory to the following path:

   `[AdminStudio_Installation_Directory]\Common`

For example:

```
C:\Program Files (x86)\AdminStudio\2021\Common
```

4. At the command prompt, execute the following command to generate a request code:

   ```
   TPSconfig /return /no_internet
   ```

5. E-mail the request code to Flexera Support in a text file attachment. A Technical Support Engineer will manually process the deactivation on your behalf.

Note • *For more information, see the Knowledge Base article: Deactivating AdminStudio:*

https://community.flexera.com/t5/AdminStudio-Knowledge-Base/Deactivation-of-AdminStudio-and-WiseScript-Editor/ta-p/5277

Evaluating AdminStudio

You can choose to evaluate AdminStudio for 21 days. By selecting the **Continue to Evaluate AdminStudio** option on the dialog box that opens when you launch AdminStudio, you can begin evaluating the AdminStudio 2022 R2 Enterprise Edition client tools.

Information about evaluating the AdminStudio client tools includes the following topics:

- AdminStudio Client Tools Evaluation Restrictions
- Evaluating AdminStudio Client Tools
- Evaluating AdminStudio's Microsoft App-V Support
- Evaluating the Automated Application Converter "Multiple Application" Option

AdminStudio Client Tools Evaluation Restrictions

When you run AdminStudio in trial/evaluation mode, all of its features are fully available, with the following restrictions:

- **Can create only one Application Catalog**—You are permitted to create only one Application Catalog, and it must be named `AdminStudio Evaluation Catalog`.
- **Ten package import limit**—Only 10 total packages (of one or more deployment types) can be imported into the Application Catalog.
- **Package deletion not permitted**—After you import a package into the Application Catalog, you are not permitted to delete it.
- **AdminStudio PowerShell Cmdlets support is disabled**—All platform support is disabled.
- **Cannot save changes in InstallShield Editor**—When using InstallShield Editor in evaluation mode, you will not be permitted to save any changes.
- **WiseScript Editor not available**—In AdminStudio Evaluation mode, WiseScript Editor is not available.

Evaluating AdminStudio Client Tools

To evaluate the Enterprise Edition client tools, perform the following steps.

Task ***To evaluate the AdminStudio Enterprise Edition client tools:***

1. Install AdminStudio, as described in the *AdminStudio Installation Guide*.
2. Launch AdminStudio. A dialog box opens, stating that you are using a time-limited trial version.
3. If you want to evaluate AdminStudio, select **Continue to Evaluate AdminStudio** and click **Next** (or just wait ten seconds). The product will launch.

   Each time you open AdminStudio while you are in evaluation mode, this dialog box shows you how many days are left in your trial period.
4. If you have five or fewer days left in your trial period, the dialog box remains open, requiring you to click before you can proceed. Do one of the following:

   a. If your trial period is not over, you can continue to use AdminStudio by selecting the **Continue to Evaluate AdminStudio** option and clicking **Next**.

   b. If you have already purchased a activation code or want to purchase one online, select **Activate or Purchase AdminStudio** and click **Next**.

Evaluating AdminStudio's Microsoft App-V Support

While evaluating the AdminStudio Enterprise Edition client tools, you will be able to convert a Windows Installer package to an App-V application using the Automated Application Converter, Conversion Wizard, Repackager, and the InstallShield App-V Assistant. However, an App-V application built using an evaluation version of AdminStudio will display the following message every time it is launched:

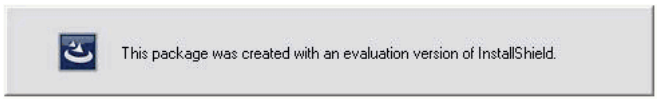


Figure 1-1: Evaluation Version Message

After purchasing the AdminStudio Application Virtualization, you will be able to remove this message by rebuilding the App-V application.

Evaluating the Automated Application Converter "Multiple Application" Option

The "multiple application" option of Automated Application Converter is only available when you purchase Application Virtualization with AdminStudio Enterprise Edition.

Note • *If you purchase Application Virtualization with AdminStudio Professional Editions, you will only be able to convert one package at a time, using one virtual machine.*

When using an evaluation version of AdminStudio, you will be able to use Automated Application Converter to convert a directory full of Windows Installer packages into individual virtual packages, but the conversion will be limited to three packages per run, using only one virtual machine. Therefore, only the first three packages that Automated Application Converter encounters will be converted to virtual applications.

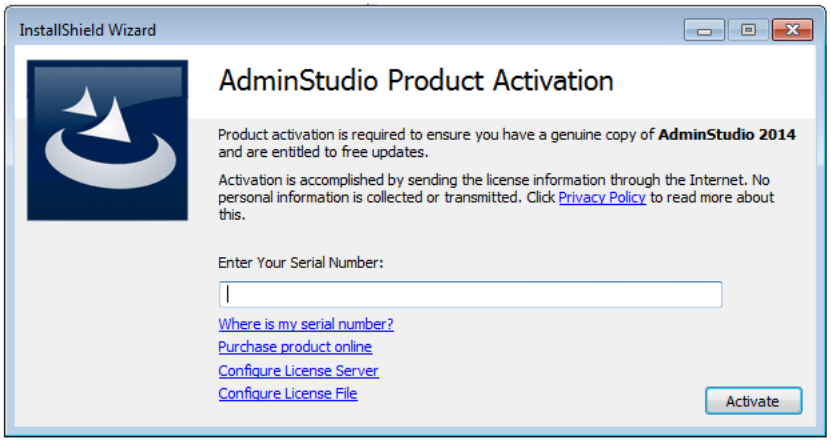
Upgrading Your Product Edition

An upgrade feature has been built-in to AdminStudio that allows you to activate features in a higher edition or in an additional add-on pack without re-installing the application. You just need to enter a Activation Code for the upgrade that you purchased, and the features of that Edition are immediately unlocked and are available to you.

To upgrade, perform the following steps:

Task ***To upgrade your AdminStudio Edition:***

1. Contact an AdminStudio Sales Representative and purchase a Activation Code for the desired edition and/or add-on pack.
2. Launch AdminStudio, Application Catalog, Automated Application Converter, Virtual Package Editor, or QualityMonitor.
3. On the **Help** menu, click **About**. The **About** dialog box opens.
4. Click the **Upgrade** button. The **AdminStudio Product Activation** dialog box opens, prompting you to enter the activation code of the edition that you want to upgrade to.



5. Enter the activation code of the edition you purchased and click the **Activate** button. After a few seconds, you will receive a message that activation was successful. The functionality of the upgraded edition and/or add-on pack is immediately available to you.

Using Help

Help is available both from the AdminStudio interface **Help** menu and directly from certain individual interface elements.

AdminStudio product documentation is also available to you online at:

https://docs.flexera.com/?product=AdminStudio

Contacting Us

Flexera is headquartered in Itasca, Illinois, and has offices worldwide. To contact us or to learn more about our products, visit our website at:

http://www.flexera.com

2

Product Activation for AdminStudio

Product activation confirms the authenticity of your AdminStudio software. This is done to protect you from the adverse effects of pirated software. The process also verifies that AdminStudio has not been activated on more machines than allowed by the AdminStudio End-User License Agreement (EULA).

- Licensing Options
- Overview of Subscription License Life Cycle
- Evaluating AdminStudio Before Activating It
- Purchasing an AdminStudio License
- Activating Through the Internet
- Activating Through a Web Page
- Activating the Package Feed Module
- Subscription Expiration and Renewal
- Automatic Activation on Product Upgrade
- Uninstalling and Reinstalling AdminStudio
- Returning a License to Your Account on the Activation Server
- Specifying the License Server for Concurrent Licenses
- Troubleshooting Activation Issues
- Activation Errors
- Activation FAQs

Licensing Options

AdminStudio 2022 R2 | 24.0 offers subscription licensing with two different types of licensing:

- **Node-locked subscription licensing**—With this model, the product license is tied to a specific user and machine for the subscription duration entitled. This model is the traditional option and the one that is most often purchased.

  Node-locked licensing has two types of activations:

  - **Activation code based**—This type requires a product activation code to activate the product.
  - **License file based (referred to as Self-Hosted)**—Self-hosted is used when AdminStudio is used in a locked-down environment with no access to the Internet. A license file needs to be generated in the Flexera Product and License Center using the MAC address of the AdminStudio machine.

  If you are using the node-locked type of license, it is your responsibility to maintain your license on your machine. Sharing this type of license between multiple users does not comply with Flexera products' end-user license agreements (EULAs).

- **Concurrent subscription licensing with License Server**—This model enables sharing or floating of AdminStudio licenses between multiple users for the subscription duration the concurrent license is entitled; it is also sometimes called floating licensing. Concurrent licensing can provide you with greater flexibility and cost efficiency than the traditional node-locked licensing. AdminStudio concurrent subscription licensing can be configured with the **FlexNet License Server**.

  Concurrent licensing also has two types of activations:

  - **Activation code based** (also referred to as Concurrent).
  - **License file based** (referred as Self-Hosted with License Server).

  If your organization purchased concurrent FlexNet License Server for AdminStudio, you need to identify the license server that you are using when you install AdminStudio on your machine. Subsequently, every time that you start AdminStudio on your machine, the server is queried to verify that the required license is available. If the license is available, you are granted access to AdminStudio. No product activation is required with concurrent licensing.

  Important • *AdminStudio concurrent licenses are available for Service Providers only.*

To learn more about these different licensing models and determine which option best fits your requirements, contact your AdminStudio sales representative.

Overview of Subscription License Life Cycle

The subscription licensing model requires that you activate AdminStudio for the subscribed duration on your machine. It also requires periodic renewal based on the subscription duration, which occurs automatically in most cases if the AdminStudio subscription is renewed on time. Activation and renewal verify that AdminStudio has not been activated on more machines than allowed by the AdminStudio EULA. If you are using the node-locked type of license, it is your responsibility to maintain your license on your machine.

The following information describes product activation and renewal, as well as different events that may occur for a license.

- Product Activation
- Product Renewal
- Product Upgrade
- Moving a License
- Permanently Transferring a License

Product Activation

After you first launch AdminStudio, the Activation Wizard opens. The wizard guides you through a series of steps to activate AdminStudio. You enter a product activation code, which is used to authenticate the AdminStudio license and unlock the product. The wizard first attempts an online activation. If online activation is unsuccessful, the wizard enables you to use the offline method (activation through a Web page that you can access from a different machine).

If you do not activate AdminStudio the first time that you launch it, you have a limited number of days to use it before activation is necessary. The Activation Wizard shows the number of days that are left in your trial period.

Sometimes activation is not successful. The most common reason is that the activation code was used to activate AdminStudio on another machine. The activation wizard protects the license in this case, preventing users from activating AdminStudio on more machines than allowed by the EULA.

Product Renewal

Once the product has been activated, renewal is periodically required based on the subscription duration. Renewal is the process by which an activated product is updated for the new subscription duration and checked to verify that it is still installed according to the AdminStudio EULA, and that it has not been activated on more machines than allowed.

AdminStudio displays the subscription info on the AdminStudio **About** dialog to remind you of the date when you will need to renew your subscription in order to continue using the product.

For more information on renewal of the product.

AdminStudio will show you warning messages when you get closer to the end of subscription based on various intervals.

If you have already renewed the subscription then follow the below two topics:

- Renewal for Activation Code License Type
- Renewal for Self-Hosted License Type

Renewal for Activation Code License Type

Renewal typically occurs automatically, without any interaction on your part if the AdminStudio subscription renewal is updated in the back office on time. After the new subscription date is updated in the back office, the product must be re-activated using the same activation code in the Activation Wizard. This re-activation will pick up the new expiration date and then product will start to function as usual after successful manual re-activation.

If renewal cannot be automatically performed when needed (for example, if you do not have an Internet connection), the Activation Wizard is displayed when you launch AdminStudio. At that point, the Activation Wizard lets you renew AdminStudio using the same wizard that is used for activation. The wizard first attempts an online renewal. If online renewal is unsuccessful, the wizard enables you to use the off-line method. The off-line activation process for renewal is the same as that for activation.

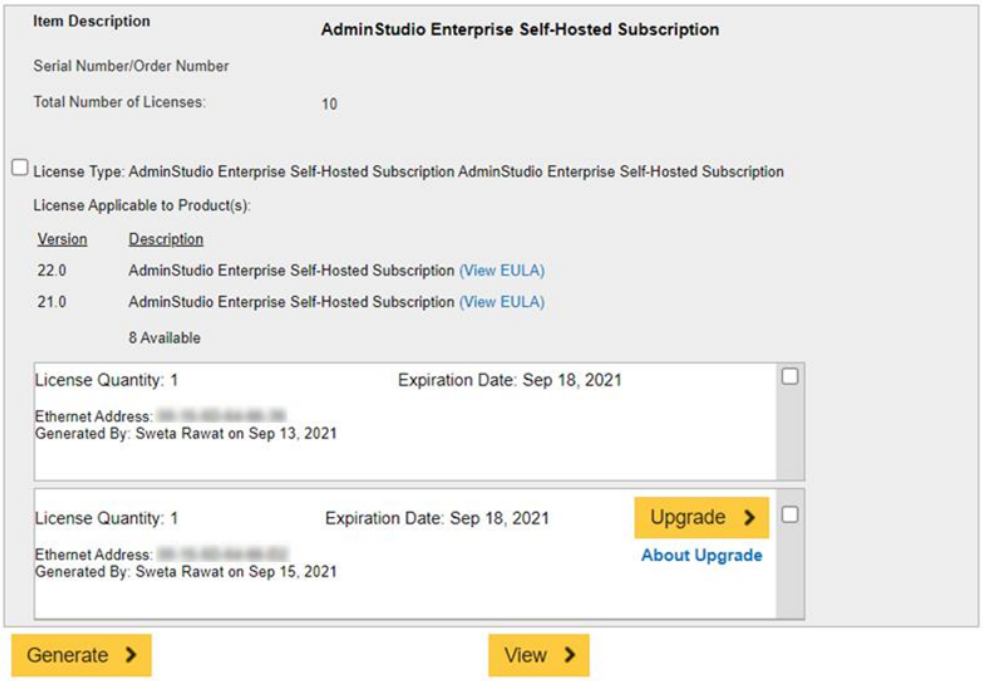
Sometimes renewal is not successful. The most common reason is that the activation code was used to activate AdminStudio on another machine. The Activation Wizard protects the license in this case, preventing users from activating AdminStudio on more machines than allowed by the EULA.

Renewal for Self-Hosted License Type

In the case of a Self-Hosted license type, a new license file must be generated in the Flexera Product and License Center.

Task ***To generate a new license file in the Product and License Center:***

1. Log on to the Flexera Community and select **Product and License Center** from the **Product Access** menu.
2. Click the **Let's Go** button under AdminStudio.
3. In the **Product List**, click **AdminStudio**.
4. Click on the AdminStudio Self-Hosted license type and version that you have. The **Product Download** page for that version opens.
5. Click on the **Licenses** tab. The **License Information** page opens.



6. Click the **Generate** button. A new license file for the specified MAC address with a new expiration date will be generated.

Product Upgrade

The method to perform a product upgrade depends upon license type:

- Product Upgrade for Activation Code License Type
- Product Upgrade for Self-Hosted License Type

Product Upgrade for Activation Code License Type

When a new version of AdminStudio 2021 R2 or later is released, you must use the same activation code to activate the new version.

After you run the setup to upgrade the AdminStudio installation to a new version (for example after AdminStudio 2021 R2 is upgraded to AdminStudio 2022 on the same machine), the same activation code that was used for AdminStudio 2021 R2 must be used in the activation wizard for AdminStudio 2022 to activate AdminStudio 2022.

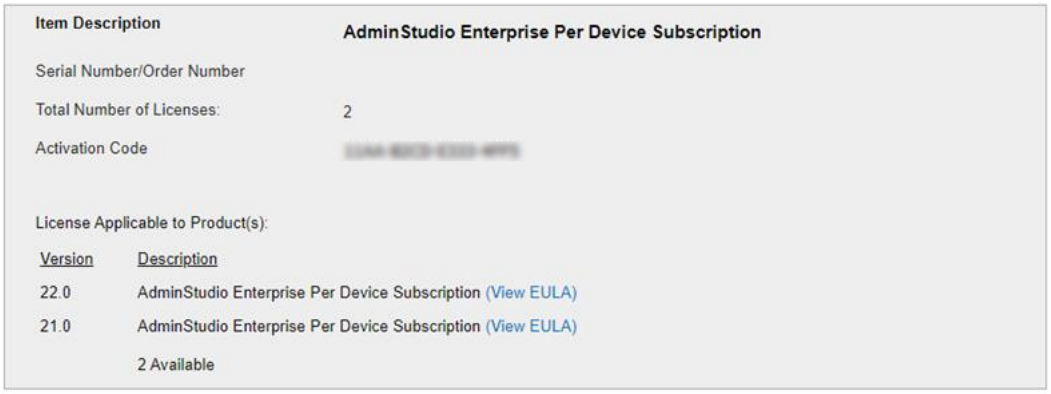


Figure 1: AdminStudio Enterprise Per Device Subscription

Product Upgrade for Self-Hosted License Type

When a new version of AdminStudio 2021 R2 or later is released, click on the **Upgrade** button in the Product and License Center to generate a new license file for the new version of the product installed in the same machine, upgrading the current version.

Task ***To generate an upgraded license file in the Product and License Center:***

1. Log on to the Flexera Community and select **Product and License Center** from the **Product Access** menu.
2. Click the **Let's Go** button under AdminStudio.
3. In the **Product List**, click **AdminStudio**.
4. Click on the AdminStudio Self-Hosted license type and version that you have. The **Product Download** page for that version opens.
5. Click on the **Licenses** tab. The **License Information** page opens.
6. Do one of the following:
   - **If you want to upgrade an installed version on a machine**—Click the **Upgrade** button. A new license file for the specified MAC address will be generated without consuming an additional license.
   - **If you want to install the new version on a different machine**—Click the **Generate** button to generate a new license file.

Moving a License

If you obtain a new replacement machine, you can move your license from your old machine to your new machine.

In order to move your license to your new machine, you must first return your license to your account on the activation server. This process is sometimes referred to as deactivation. Returning the license makes it available again so that you can use your activation code for activation on a different machine. To learn how to return your license, see Returning a License to Your Account on the Activation Server.

Once you have returned your license, you can use the same activation code to activate the product on your new machine.

Important • *There is a limit to the number of times that a license can be moved. AdminStudio allows you to return the license back to the activation server two times. If you want to move the license beyond two times, contact your AdminStudio sales or support representative.*

Permanently Transferring a License

In some cases, it may be necessary to permanently transfer your license to a different user and machine in an organization. For example, if your responsibilities are changing and someone else will be using AdminStudio, you may need to transfer your license to that employee. In order to transfer your license, you must first return your license to your account on the activation server. Returning the license makes it available again so that the new user can use your activation code for activation on a different machine. Note that the new user will need to activate AdminStudio on their machine after they have installed it.

If a license is being permanently transferred to you, ensure that you contact your AdminStudio sales or support representative and give them the updated registration information for the license. The registration information update is required in order to best serve you and to notify you about product updates and special offers.

Important • *There is a limit to the number of times that a license can be permanently transferred. AdminStudio allows you to return the license back to the activation server two times. If you want to permanently transfer the license beyond two times, contact your AdminStudio sales or support representative.*

Evaluating AdminStudio Before Activating It

If you have not purchased a license for AdminStudio, you can still install AdminStudio and use it for a limited number of days without activating it. The activation wizard that AdminStudio displays whenever you launch AdminStudio in trial mode shows you how many days are left in your trial period. In addition, the About AdminStudio dialog box in AdminStudio shows the number of days remaining. To access the About AdminStudio dialog box: On the Help menu in AdminStudio, click About AdminStudio.

If you do not activate AdminStudio within the trial period, AdminStudio will stop working when the trial period has ended. You can activate AdminStudio at any time before or after the trial period has ended.

To obtain a copy of AdminStudio that you can evaluate, visit the Flexera website.

Purchasing an AdminStudio License

You can purchase AdminStudio through several methods:

- For information on how to purchase AdminStudio, visit the Flexera website:

  https://www.flexera.com/about-us/contact-us.html

- Contact your AdminStudio sales representative.
- Purchase from a reseller. Visit the Flexera website for a list of resellers in your country.

When you purchase AdminStudio, you receive an activation code that you can use for activation.

Activating Through the Internet

Online activation through the Internet is a quick process. Online activation occurs when you enter your activation code in the activation wizard and click the Activate button.

Task ***To activate AdminStudio through the Internet:***

1. Launch AdminStudio. Before AdminStudio starts, the activation wizard opens. If you have more than 5 days left in your trial period, the wizard automatically disappears after a few seconds.

   If you have 5 or fewer days left in your trial period, the wizard remains, requiring you to click before you can proceed. If your trial period is not over, you can select the **Continue to Evaluate AdminStudio** option and then click the **Next** button on the wizard to use AdminStudio without activating it.

2. When you are ready to proceed with activation, select the **Activate or Purchase AdminStudio** option in the activation wizard and then click the **Next** button. AdminStudio displays a dialog that requests the activation code.

3. Enter your activation code, and then click the **Activate** button.

The activation wizard transmits the activation request to the activation server. When the server receives your activation request, it validates the request. If the activation request is valid, the server automatically transmits the activation response text to the activation wizard, which then activates AdminStudio.

Activating Through a Web Page

If you do not have an Internet connection on the machine that has AdminStudio or if you are having problems completing the online activation process, the activation wizard gives you the option of performing offline activation through a self-service Web page on a different machine.

Task ***To activate AdminStudio through a Web page:***

1. Attempt to active AdminStudio through the Internet as described in Activating Through the Internet. If it cannot be completed, the activation wizard displays a message explaining why it could not occur.

2. Click the **Proceed with offline activation** button. The **Offline Activation** dialog opens. The **Request text** box contains your request text. The request text starts with **<?xml version**, and it ends with **</Request>**.

3. To save the request text to a text file that you can upload from a machine with an Internet connection, click the **Save** button. The wizard lets you save the text as a `.request` file.

4. Visit the Offline Activation Web page—a part of the Flexera Product and License Center—and follow the instructions to browse to the `.request` file that you saved.

   https://flexerasoftware.flexnetoperations.com/control/inst/offlineActivation

When you click the button on the Offline Activation Web page to submit the activation request and obtain the activation response file (.xml), the Web page prompts you for a place to save the `.xml` file. Save it and make it available on the machine on which you initiated the activation process.

5. When you have the activation response file (.xml) and you are ready to complete the activation process, launch AdminStudio to open the activation wizard.
6. Proceed to the **Offline Activation** dialog, which has a **Response text** box.
7. Click the **Load** button. The Open dialog opens.
8. Browse to the activation response file (.xml), and then click the **Open** button. The **Open** dialog closes, and the wizard writes the response text in the **Response text** box.

   The response text starts with **<?xml version**, and it ends with **</Response>**.

   ***Note** • As an alternative for step 8, you can copy the response text to your Clipboard and then use the Paste button to paste the Clipboard contents into the **Response text** box.*
9. Click the **Activate** button.

The activation wizard activates AdminStudio.

***Tip** • The aforementioned procedure is also used to perform an offline return of a license.*

Activating the Package Feed Module

Once you have purchased a license of the Package Feed Module, follow the below steps to activate it in your AdminStudio Enterprise or Professional Edition:

Task ***To activate the Package Feed Module:***

1. Open Application Manager and on the **Support** tab, click **About Application Manager**. The **About Application Manager** dialog box opens.
2. Click **Upgrade**. The **AdminStudio Product Activation Wizard** opens.
3. In the **Enter Your Activation Code** field, enter your Package Feed Module activation code.
4. Click **Activate** to activate the Package Feed Module. You will then need to restart AdminStudio.

Subscription Expiration and Renewal

You will see warning messages when your subscription is about to expire and when it expires. The first message will appear when there are 84 days left for your subscription to expire. Subsequently, you will this message when there are 70, 42, 28, 7,6, 5, 4, 3, 2 and 1 days left for your subscription to expire.

- Subscription Warning Message

- Subscription Renewal
- Activating the Expired Subscription

Subscription Warning Message

The following warning message will be displayed showing the number of days left for your subscription to expire.

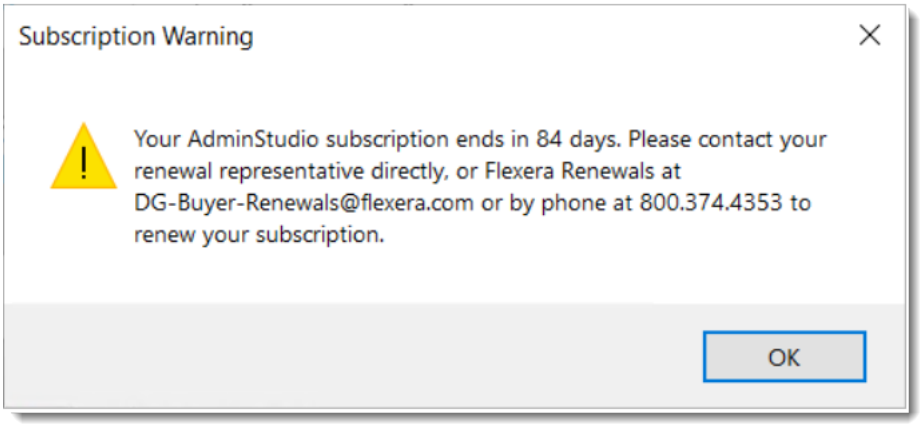


Figure 2: Subscription warning message for Self-hosted and Concurrent license types

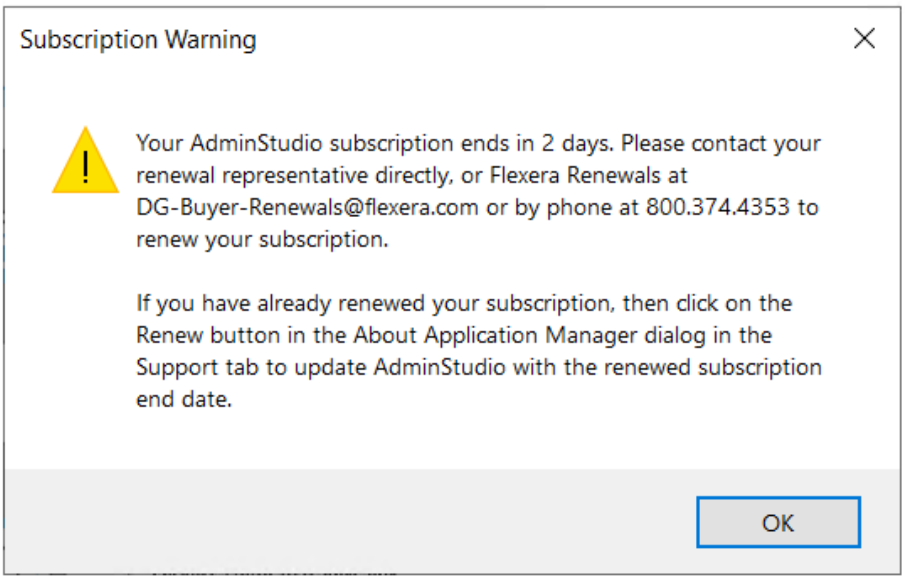


Figure 3: Subscription warning message for Activation Code license type

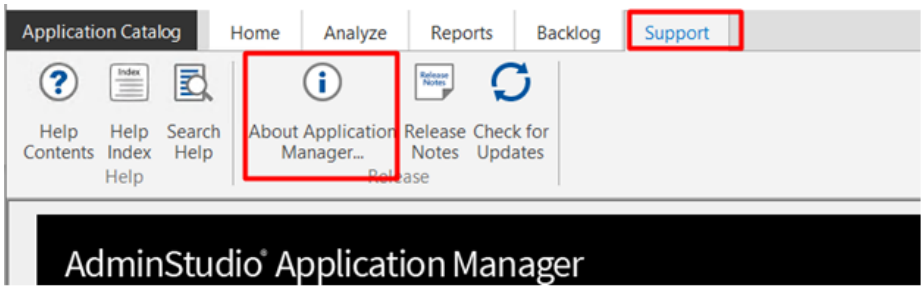
Subscription Renewal

If the AdminStudio subscription is renewed before the expiration, then you need to update the product with the new subscription end date.

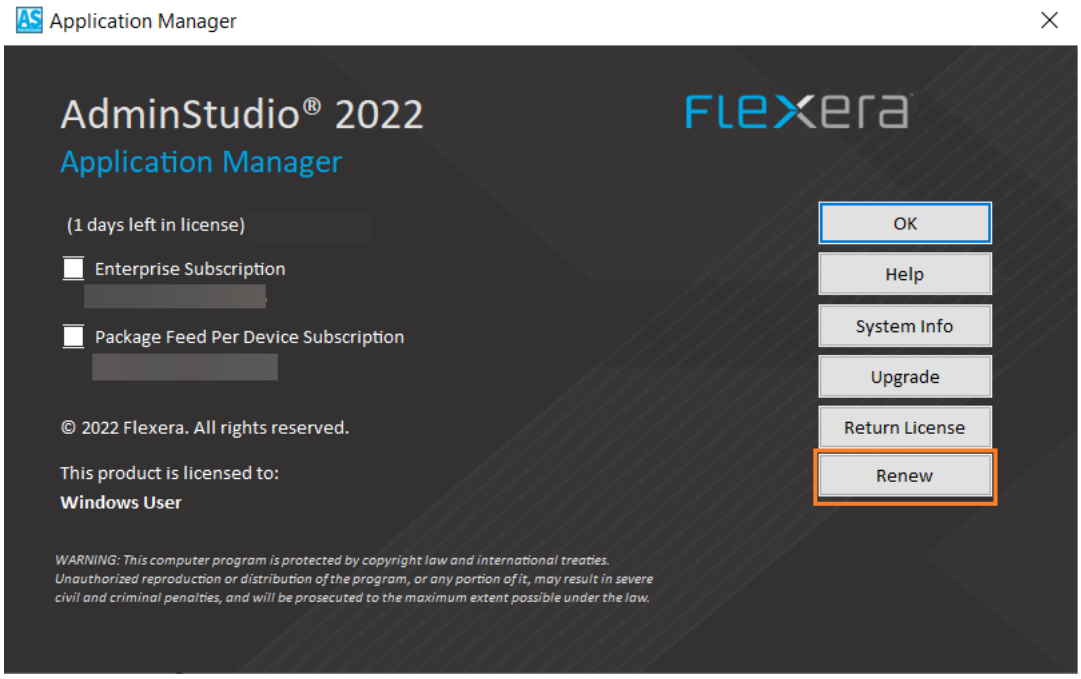
To update the product with the new subscription end date, perform the following steps:

Task ***To renew subscription in the product:***

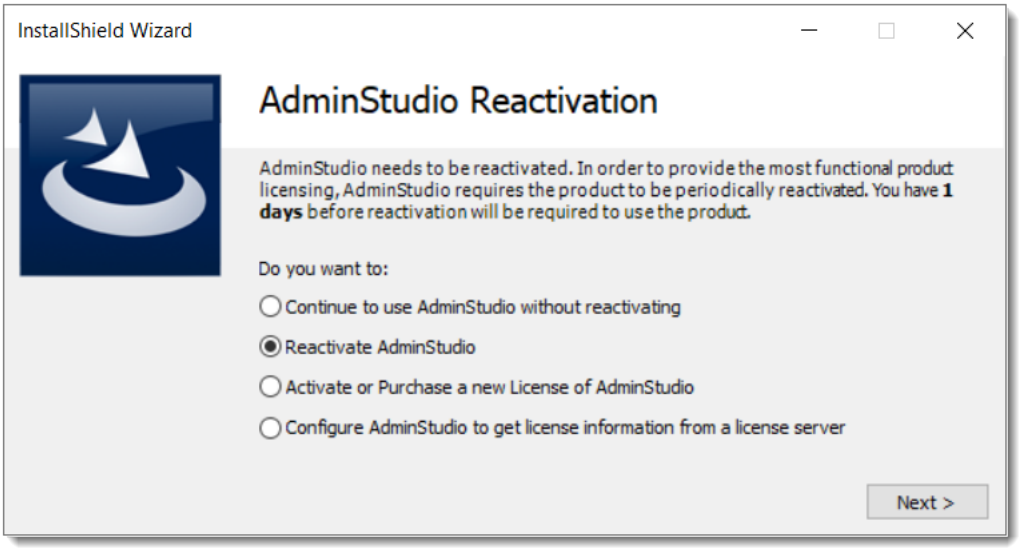
1. In Application Manager, click on the About Application Manage ribbon button in the Support tab to launch the About Application Manager dialog.



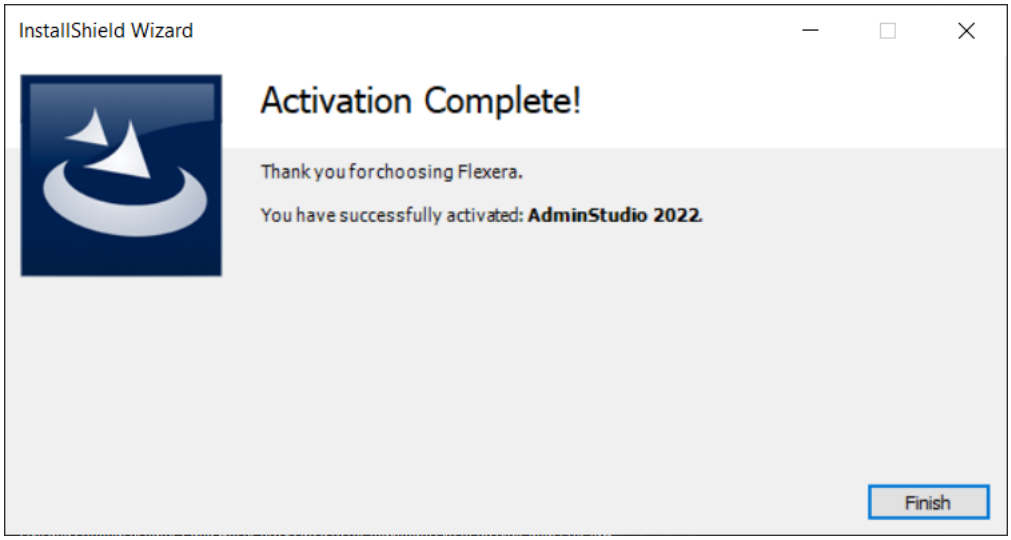
2. Select the one of the following two subscriptions you wish to renew by clicking on the check box
   - AdminStudio (Enterprise/Professional) Subscription
   - Package Feed Module Subscription
3. Click the Renew button to renew the selected subscription.



4. Click **Yes** to confirm and proceed with renewal if you have recently renewed AdminStudio or Package Feed Module subscription. The **AdminStudio Reactivation** wizard will appear.
5. Select **Reactivate AdminStudio** radio button and click **Next**.



6. The **Activation Complete** panel appears, click **Finish**.



7. . Upon successful renewal, you will see the updated number of days lefts in the About Application Manager dialog based on the new subscription end date.

Note • *Note: If a license file was previously used for activation, then a new license file needs to be generated for the new subscription end date from the Product and License Center.*

Activating the Expired Subscription

The following message will be displayed upon expiration of the subscription. Clicking OK will be close AdminStudio.

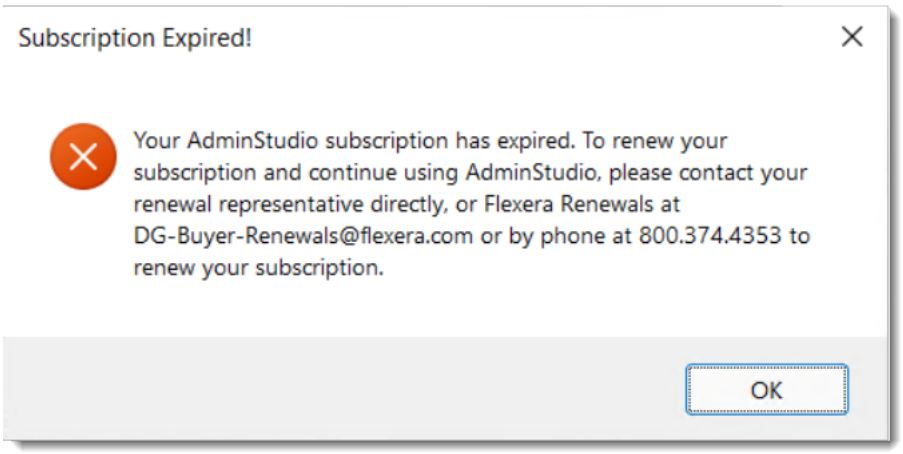


Figure 4: Subscription Expired Message (AdminStudio)

The following message will be displayed upon the expiration of the Package Feed Module subscription. Clicking OK button, will restart AdminStudio. Upon expiration Package Feed Module will go back to trial mode while AdminStudio will continue to be fully functional if the subscription on AdminStudio is active.

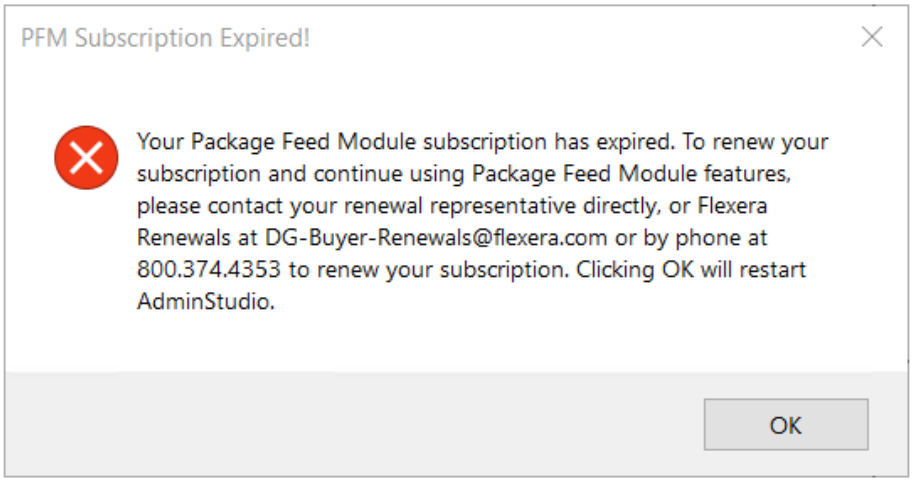


Figure 5: Subscription Expired Message (Package Feed Module)

After the subscription is expired, upon its first launch, AdminStudio will automatically attempt to reactivate by checking of the subscription is renewed. The automatic reactivation will fail if the subscription is not renewed. When the subscription is renewed later, perform the following steps to reactivate AdminStudio with the new subscription end date.

Task ***To reactivate an expired subscription:***

1. Launch AdminStudio to see the AdminStudio Reactivation wizard.

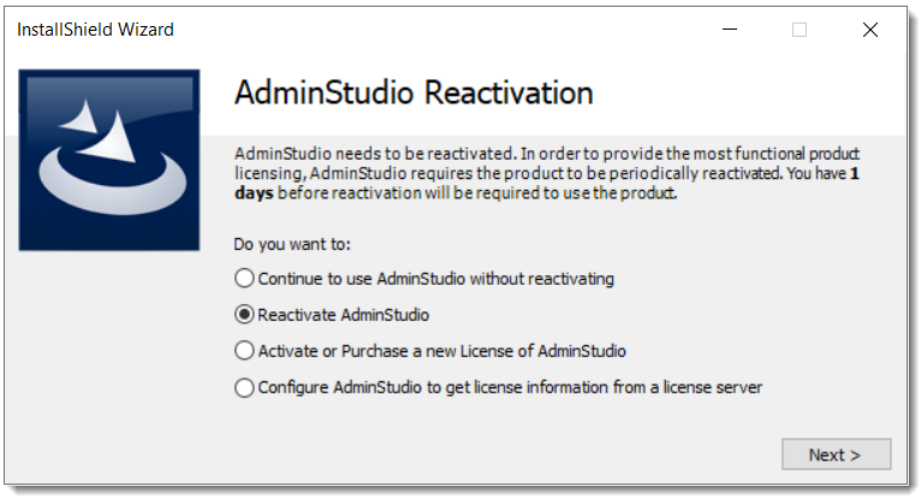
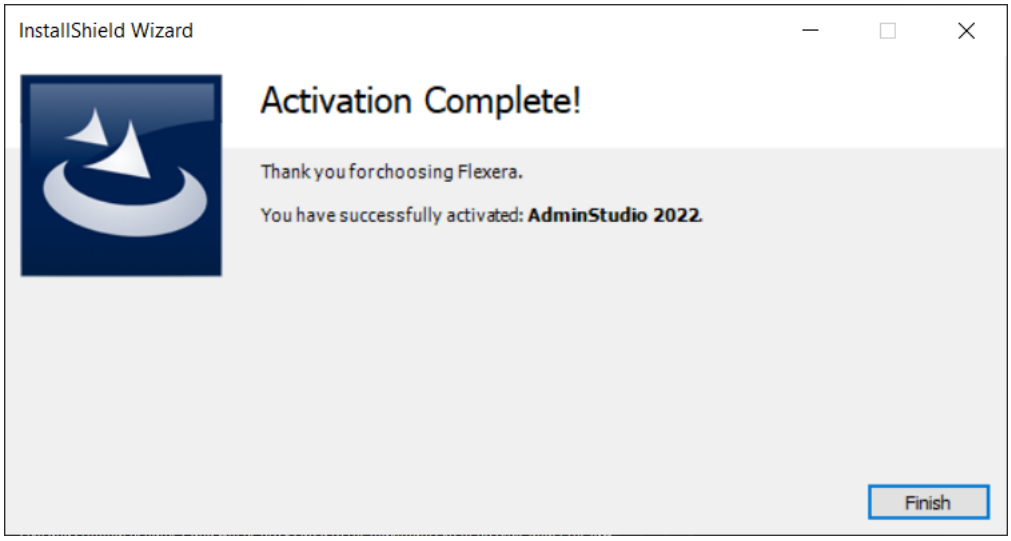
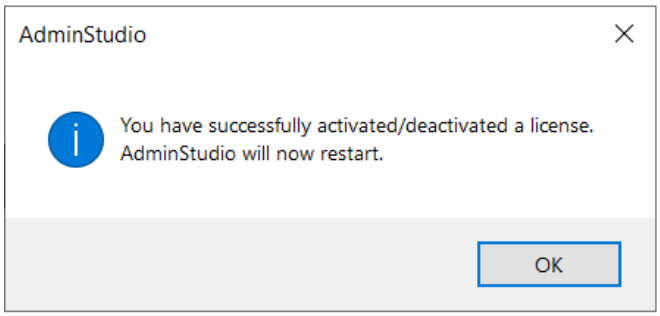


Figure 6: AdminStudio Reactivation Wizard Panel

2. Select Reactivate AdminStudio and click **Next**.
3. The **Activation Complete** panel appears, click **Finish**.



4. Click **OK** and confirm the popup for activation process. Upon clicking **OK** button, the AdminStudio will now restart.



5. Upon successful renewal, you will see the updated number of days lefts in the About Application Manager dialog based on the new subscription end date.

Note • *The automatic reactivation is applicable only if you have previously activated AdminStudio using an activation code. If a license file was previously used for activation, then a new license file needs to be generated for the new subscription end date from the Product and License Center.*

Automatic Activation on Product Upgrade

Starting from AdminStudio 2021 R2 (v22.0), when a certain version of a product on active subscription is upgraded to a higher version, for example from AdminStudio 2021 R2 to AdminStudio 2022, the setup upgrade process will attempt automatic activation of the higher version of the product after the setup upgrade is completed. If the automatic activation fails, then perform the following steps to activate the upgraded version

Task ***To activate an upgraded version of AdminStudio:***

1. Launch AdminStudio to see the AdminStudio Reactivation wizard.

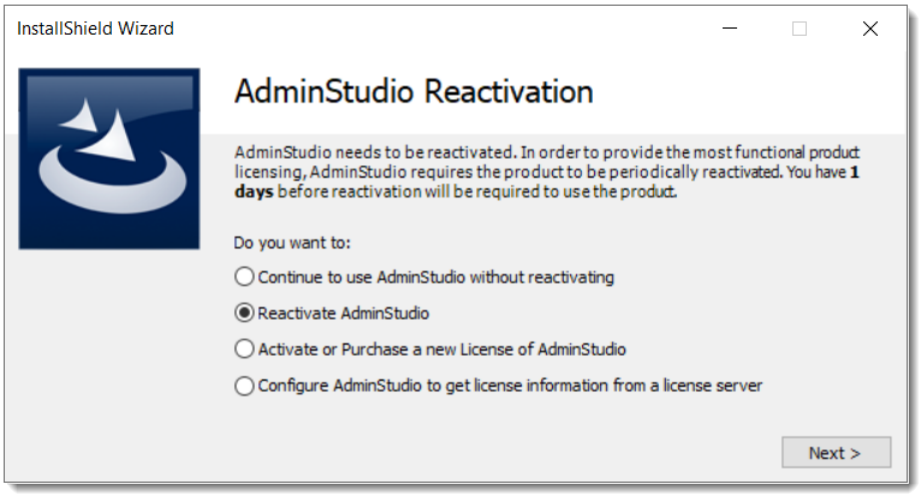
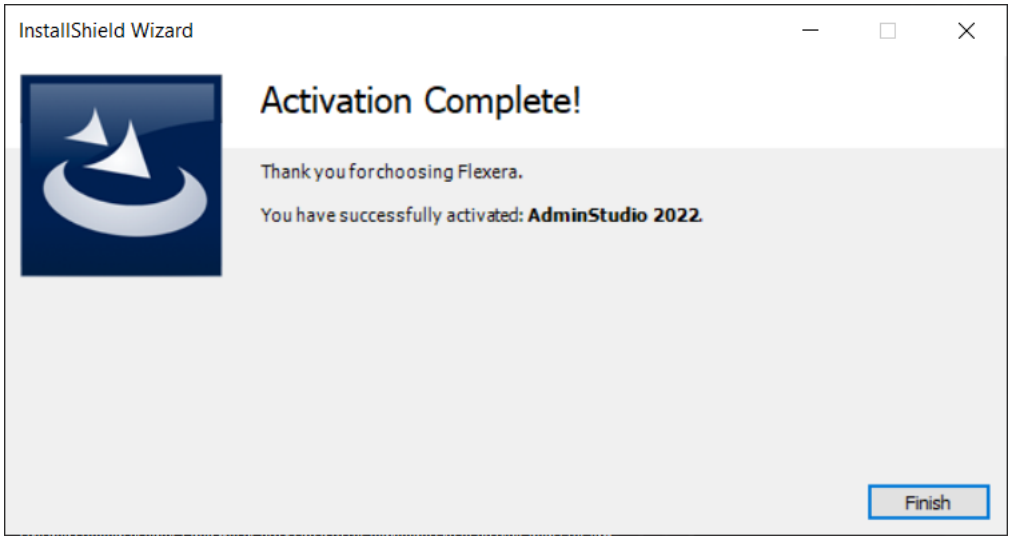
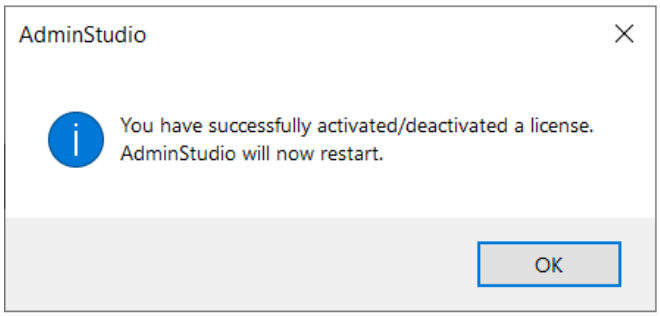


Figure 7: AdminStudio Reactivation Wizard Panel

2. Select **Reactivate AdminStudio** and click **Next**.
3. The **Activation Complete** panel appears, click **Finish**.



4. Click **OK** and confirm the popup for activation process. Upon clicking **OK** button, the AdminStudio will now restart.



5. Upon successful renewal, you will see the updated number of days lefts in the About Application Manager dialog based on the new subscription end date.

***Note** • The automatic reactivation during setup upgrade is applicable only if you have activated the previous version of AdminStudio using an activation code. If a license file was previously used for activation, then a new license file needs to be generated for the new subscription end date from the Product and License Center.*

Uninstalling and Reinstalling AdminStudio

If you need to move your AdminStudio license from one of your machines to another machine in your organization, or if you need to permanently transfer your license to a different user in your organization, you must first return your license to your account on the activation server. This process is sometimes referred to as deactivation. This typically occurs automatically if your machine is connected to the Internet when you uninstall AdminStudio from the original machine. Returning the license makes it available again so that you can use your activation code for activation on your other machine.

In some cases, it is not possible to automatically return a license during uninstallation. For example, if your machine is not connected to the Internet when you uninstall AdminStudio, your license cannot be returned. Therefore, if you want to return your license to make it available for activation on a different machine, the recommended method is to first return the license, as described in Returning a License to Your Account on the Activation Server, and then uninstall AdminStudio.

***Important** • Note that there is a limit to the number of times that you can return your license and then activate on another machine. AdminStudio allows you to return the license back to the activation server two times. If you want to move the license beyond two times, contact your AdminStudio sales or support representative.*

Returning a License to Your Account on the Activation Server

If you have activated AdminStudio on a machine but you no longer want it to be activated on that particular machine, you can return your license to your account on the activation server. Note that if you do this without uninstalling AdminStudio, AdminStudio reverts back to trial mode if any days remain in your trial period. If the trial period has ended, AdminStudio will stop working.

One example of when you may want to return a license is if you want to permanently transfer your license to another machine in your organization. You can first return the license on the initial machine, and then use your activation code to activate AdminStudio on a different machine.

Task ***To return a license to your account on the activation server:***

1. Launch AdminStudio.
2. On the **Help** menu, click **About AdminStudio**. The **About AdminStudio** dialog box opens.
3. Click the **Return License** button.

AdminStudio transmits the license return request to the activation server. When the server receives your request, it returns your license.

In case of subscription, the license will be returned to the activation server. So, if you click the **Return License** button on the **About AdminStudio** dialog box, then the subscription license will be returned, and the subscribed version installed on that machine revert to trial mode if any days remain in your trial period. If the trial period has ended, AdminStudio will stop working.

If the license cannot be returned—for example, if your machine is not connected to the Internet when you attempt to return the license—the Activation Wizard is displayed, enabling you to return your license through a Web page that you can access from a different machine. The procedure is essentially the same as off-line activation: the Activation Wizard creates an activation request file, you browse the request file from a Web page, and the Web page prompts you for a location for saving the corresponding response file. Next, you browse to the response file in the Activation Wizard. For more information, see Activating Through a Web Page.

Important • *Note that there is a limit to the number of times that you can return your license and then activate it on another machine. AdminStudio allows you to return the license back to the activation server two times. If you want to move the license beyond two times, contact your AdminStudio sales or support representative.*

Specifying the License Server for Concurrent Licenses

If your organization purchased concurrent licenses for AdminStudio, your organization must first set up the license server. Once that has been done, you need to identify the license server that you are using on the machine where you install AdminStudio.

Important • *AdminStudio concurrent licenses are available for Service Providers only.*

Task ***To specify the license server:***

1. Launch AdminStudio. Before AdminStudio starts, the activation wizard opens.
2. Select the **Configure AdminStudio to get license information from a license server** option and then click the **Next** button. AdminStudio displays the **Specify License Server** dialog.
3. In the **Server** box, enter the path to the license server, or click the **Browse** button to navigate to the server.
4. If the server is not configured to use the default port, specify the server port number in the **Port** box.
5. Click the **Test Connection** hyperlink.

The wizard connects your machine with the license server.

Tip • *For more information about the license server, see the documentation that is provided to you when you purchase your concurrent licenses.*

Troubleshooting Activation Issues

General Troubleshooting Tips

The following tips may help you resolve issues that may occur during the activation process.

- Ensure that you are entering the activation code correctly, and that it is in the format XXXX-XXXX-XXXX-XXXX (4 sets of 4 characters).
- If you or someone in your organization previously activated AdminStudio on another machine, you must first return your AdminStudio license on that machine through a full uninstallation before it can be activated on the new machine. To learn more, see Uninstalling and Reinstalling AdminStudio.

Activation Errors

If you encounter an activation error, see Activation Errors for help with resolving it. For the latest troubleshooting information, see the AdminStudio Knowledge Base:

https://community.flexera.com/t5/AdminStudio-Knowledge-Base/tkb-p/Admin-Studio-Knowledge

Offline Activation

If you are unable to activate AdminStudio through the automatic online method, offline activation is required. You can perform offline activation on a different machine that has Internet access. For more information, see Activating Through a Web Page.

Further Assistance

For more information about activating AdminStudio, visit the AdminStudio Knowledge Base in the Flexera Community.

If you have tried all of the aforementioned suggestions and you still have not been able to activate AdminStudio, log in to the Flexera Community, select **Open New Case** on the **Support** menu, and submit a Support case.

Activation Errors

The following table contains tips on how to resolve errors that may occur when your try to activate AdminStudio.

Tip • *For the latest troubleshooting information for AdminStudio 2022 R2 | 24.0, see the AdminStudio Knowledge Base.*

Table 1 • Activation Errors

Error Number	Description	Troubleshooting Information
20653	The number of activations allowed for this account has been exceeded.	The AdminStudio end-user license agreement allows you to install and activate AdminStudio a limited number of times. This error occurs if that limit has been exceeded. To resolve this error, try one of the following solutions: • Uninstall AdminStudio on one machine; doing so returns your license to your account on the activation server, allowing you to activate on a different machine. • Contact your AdminStudio sales representative to purchase an additional AdminStudio license, or to purchase concurrent licensing. • Ensure that no one else in your organization is already using the same activation code for their activated copy of AdminStudio.
20660	The Activation Code that you entered has been disabled.	The activation code that you entered has been disabled. One example of when this may occur is if you returned your copy of AdminStudio and then later tried to use your activation code to activate AdminStudio. If you encounter this error, ensure that you entered the activation code correctly. If you did enter it correctly, contact AdminStudio Support so that they can re-enable your activation code if it is allowed. To contact support, log in to the Flexera Community, select **Open New Case** on the **Support** menu, and submit a Support case. Once the activation code has been re-enabled, you can proceed with activation.
20676	The license has been transferred between computers more times than is allowed.	There is a limit to the number of times that you can transfer your AdminStudio license from one machine to another in your organization. This error occurs if that limit has been exceeded. To resolve this error, contact your AdminStudio sales representative to purchase an additional AdminStudio license, or to purchase concurrent licensing.

Table 1 • Activation Errors

Error Number	Description	Troubleshooting Information

50020	The response text you entered is invalid.	This error occurs if you are attempting offline (email) activation and the response text that you entered in the activation wizard is incorrect. Ensure that you correctly entered the response text. The response text starts with the following string:
		`<?xml version`
		The following string marks the end of the response text:
		`</Response>`

50040	An unexpected error occurred.	This error may occur if an online activation attempt fails. For example, if your machine does not have an Internet connection, online activation cannot occur. If you do not have an Internet connection or if you are having problems completing the online activation process, the activation wizard gives you the option of performing offline activation through email. For information on how to activate offline through email, see Activating Through a Web Page.

Activation FAQs

Following are frequently asked questions and answers about the activation process for AdminStudio.

Questions

- What is product activation?
- What happens during activation?
- What site is accessed for activation?
- What is product reactivation?
- What happens during reactivation?
- How do activation and reactivation work?
- What information is required for activation and reactivation?
- Does activation affect my software or my computer?
- How do I activate AdminStudio?
- How long does it take to activate AdminStudio?
- How soon must I activate AdminStudio?
- Can I install AdminStudio without activating it?

- What happens if I do not activate AdminStudio?
- How can I obtain an activation code for activation?
- What is the difference between product activation and product registration?
- Can I uninstall my copy of AdminStudio on one machine and reinstall it on my other machine?
- What if I upgrade or get a new machine, and I forget to return my license on my old machine?
- Can I share my copy of AdminStudio with others?
- Will I be able to reinstall and reactivate AdminStudio if my hard drive crashes?
- Do I always need to be online to use AdminStudio?
- What does Flexera do with the AdminStudio activation information?
- What is product renewal?

Answers

What is product activation?

Product activation is a quick and anonymous process that confirms the authenticity of your software. This is done to protect you from the adverse effects of pirated software. The process also verifies that AdminStudio has not been activated on more machines than allowed by the AdminStudio End-User License Agreement (EULA).

After you first launch AdminStudio, the activation wizard opens. After a few seconds, the activation wizard disappears if you have not clicked on it, and AdminStudio is launched as a trial product. If you want to activate AdminStudio right away, you can select the Activate or Purchase AdminStudio option, and then click the Next button. The wizard guides you through the activation process, and in seconds, AdminStudio is activated.

Back to Top

What happens during activation?

You go through a series of easy steps to activate AdminStudio, usually through the Internet (or offline, through a Web site that you can access on a different machine). You enter a product activation code, which is used to authenticate the AdminStudio license, thus unlocking the product. The entire process takes only a few seconds.

Back to Top

What site is accessed for activation?

To activate AdminStudio and check for updates, your environment needs to have access to the following site:

https://flexerasoftware.flexnetoperations.com

Back to Top

What is product reactivation?

Product reactivation is the process by which an activated product is checked to verify that it is still installed according to the AdminStudio EULA, and that it has not been activated on more machines than allowed.

Reactivation is required on each machine on which AdminStudio has been successfully installed and activated. Reactivation must occur a certain number of months after the last activation or reactivation occurred; otherwise, AdminStudio stops working.

Reactivation offers more flexibility than licensing models that do not require it. With the reactivation model that is used for AdminStudio, a license is not permanently tied to a specific machine; this allows users to periodically update and replace hardware and the machines on which AdminStudio is installed.

Back to Top

What happens during reactivation?

Reactivation typically occurs automatically, without any interaction on your part. Several weeks before reactivation is required, the activation wizard sends a reactivation request for your activation code to the activation server silently (that is, without displaying any user interface). The server receives the reactivation request, and it validates the request. If the reactivation request is successful, the server automatically notifies the activation wizard about the status, enabling the wizard to reactivate AdminStudio for you.

If reactivation cannot be automatically performed when needed (for example, if you do not have an Internet connection), the activation wizard is displayed when you launch AdminStudio. At that point, the activation wizard lets you reactivate AdminStudio using the same wizard that is used for activation. The wizard first attempts an online reactivation. If online reactivation is unsuccessful, the wizard enables you to use the offline method.

Back to Top

How do activation and reactivation work?

For an explanation, see Overview of Subscription License Life Cycle.

Back to Top

What information is required for activation and reactivation?

Activation and reactivation require your AdminStudio activation code. No personal information is needed.

Back to Top

Does activation affect my software or my computer?

No. Activation has no effect on the performance of your computer or software.

Back to Top

How do I activate AdminStudio?

After purchasing AdminStudio (and subsequently receiving your activation code), you simply enter the activation code in the designated field that is displayed in the activation wizard when you launch AdminStudio, and then click the Activate button.

Typically, activation is completed in just a few seconds through the Internet (online activation). In some cases, offline activation is required. If so, this is accomplished through a Web site that you can access from a different machine. To learn more, see Activating Through a Web Page.

Back to Top

How long does it take to activate AdminStudio?

Internet activation (online activation) typically takes seconds to complete. It is dependent on the type of Internet technology that you are using. The amount of data being transmitted is very small, so high-speed connections are not required.

Back to Top

How soon must I activate AdminStudio?

You have a limited number of days to activate AdminStudio after the first launch. The activation wizard shows the number of days that are left in your trial period. The activation wizard is displayed every time that you launch AdminStudio during the trial period (before you have activated AdminStudio). In addition, the About AdminStudio dialog box in AdminStudio shows the number of days remaining. To access the About AdminStudio dialog box: On the Help menu in AdminStudio, click About AdminStudio.

Back to Top

Can I install AdminStudio without activating it?

Yes. After installation, you can use AdminStudio for a limited number of days without activating it.

After that trial period has ended, you need to activate AdminStudio in order to use it.

Back to Top

What happens if I do not activate AdminStudio?

AdminStudio will stop working at the end of the trial period if you do not activate it.

Back to Top

How can I obtain an activation code for activation?

If you have already purchased AdminStudio, then an activation code will be sent to you in an email. This activation code will also be available in the Flexera Product and License Center. If you cannot find your activation code, contact your AdminStudio sales representative.

Back to Top

What is the difference between product activation and product registration?

Product activation is a mandatory, anonymous process that verifies the activation code for your copy of AdminStudio. Product registration is a process that entitles you to product updates and special offers.

Back to Top

Can I uninstall my copy of AdminStudio on one machine and reinstall it on my other machine?

Yes. The recommended method is to first return your license on the current machine and install the product on a new machine. Once you have installed it on the new machine, you must activate AdminStudio on the new machine.

For more information, see Returning a License to Your Account on the Activation Server.

Back to Top

What if I upgrade or get a new machine, and I forget to return my license on my old machine?

If you are planning to upgrade or get a new machine, it is important that you first return your license. If you do not do this, your account on the activation server still reflects that your license is activated on your old machine. As a result, when you install the product on your new machine, you will not be able to activate it, and you will need to contact AdminStudio Support. If this situation has occurred frequently, you may be denied another activation.

Back to Top

Can I share my copy of AdminStudio with others?

No, AdminStudio should not be shared with other users. Do not share the activation code that you used for activation, since it is required to reactivate your product.

Back to Top

Will I be able to reinstall and reactivate AdminStudio if my hard drive crashes?

Yes. However, in most cases, the AdminStudio license will still be activated. If you attempt to reactivate but it fails, contact AdminStudio support by logging into the Flexera Community and making a selection on the **Support** menu.

Back to Top

Do I always need to be online to use AdminStudio?

Once you have activated AdminStudio, you do not need to be online to use it. When AdminStudio needs to be reactivated, the activation wizard attempts to reactivate silently (that is, without displaying any user interface). If your machine does not have an Internet connection when reactivation is attempted, the activation wizard is displayed, allowing you to perform an offline (email) reactivation.

Note • *There are certain functionalities of AdminStudio, such as the Package Feed Module and publishing packages to end point management systems, which require the AdminStudio machine to be online.*

Back to Top

What does Flexera do with the AdminStudio activation information?

The information that is used to activate AdminStudio is used within the capacity outlined by the AdminStudio End-User License Agreement (EULA). For additional information, review the privacy policy on the Flexera website:

https://www.flexera.com/legal/privacy-policy.html

Back to Top

What is product renewal?

Product renewal is the process by which the subscribed product is renewed for the new subscription duration in the backend. AdminStudio updates the new subscription duration on the expiry of the current subscription.

Back to Top

3

Getting Started with AdminStudio

The AdminStudio Start Page, which is designed to help you quickly get started evaluating and using AdminStudio, provides process information on how to perform key tasks using AdminStudio tools. Information is organized into the following tabs:

Table 3-1 • AdminStudio Start Page Organization

Icon	Start Page Tab	Description
	Getting Started	Describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. See Getting Started Tab.
	Test for Application Compatibility	Provides a flowchart that outlines how to use Application Catalog to test applications for compatibility with the latest versions of Microsoft Windows, Windows Server, macOS, Apple iOS, Google Android, and Windows Phone operating systems. See Test for Application Compatibility Tab.
	Migrate to Application Virtualization	Provides a flowchart that outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise. See Migrate to Application Virtualization Tab.
	Migrate to Windows Installer	Provides a flowchart that outlines the steps required to migrate legacy setups (such as `.exe` files) to deployable Windows Installer packages (`.msi`). See Migrate to Windows Installer Tab.
	Set Up Infrastructure	Lists the infrastructure setup steps that you need to perform prior to using AdminStudio for the first time: connect to an Application Catalog, configure virtual machines, set e-mail notification settings, and specify server/database connection settings. See Set Up Infrastructure Tab.

Opening the AdminStudio Start Page

To open the AdminStudio **Start Page**, perform the following steps.

Task ***To open the AdminStudio Start Page:***

1. Open Application Catalog.
2. On the **Home** tab, click on the **AdminStudio Tools** button and then select **Process Assistants** from the menu. The AdminStudio interface is launched, with the **Start Page** tab selected.

Getting Started Tab

The **Getting Started** tab of the AdminStudio Start Page describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. It also provides a link to information on setting up AdminStudio infrastructure.

To quickly get started evaluating and using AdminStudio, click on the link of the task you want to accomplish:

Table 3-2 • Getting Started Tab

Link	Description
Test for Application Compatibility	Test applications for compatibility with the latest versions of Microsoft Windows, Windows Server, macOS, Apple iOS, Google Android, and Windows Phone operating systems.
Migrate to Application Virtualization	Automatically repackage and convert Windows Installer packages, as well as setups in other formats, into virtual applications in Microsoft App-V, VMware ThinApp, Citrix XenApp.
Migrate to Windows Installer	Capture, repackage, and customize installations, analyze packages for conflicts with target applications, and prepare packages for distribution.
Set Up Infrastructure	Create/connect to a Microsoft SQL Server Application Catalog database. Prepare virtual machines for use in automated repackaging and testing.

Test for Application Compatibility Tab

The flowchart on this tab outlines how to use Application Catalog to test applications for compatibility with the latest versions of Microsoft Windows, Windows Server, macOS, Apple iOS, Google Android, and Windows Phone operating systems.

To test applications for operating system, perform the following steps:

- Import Packages and Mobile Apps
- Select Tests to Run and Set Default Fix Option
- Perform Testing and View Results

Import Packages and Mobile Apps

For instructions on how to import packages and mobile apps, see the following help topics:

Table 3-3 • Import Packages and Mobile Apps

#	Step	Description
1	**Import Windows Installer packages into Application Catalog.**	For instructions on how to import Windows Installer packages into the Application Catalog, see: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager)
2	**Import mobile apps into Application Catalog.**	For instructions on how to import mobile apps into the Application Catalog, see: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) • Importing Links to Public Store Applications

Select Tests to Run and Set Default Fix Option

For instructions on how to select the tests to run and set the default fix option, see the following help topics:

Table 3-4 • Select Tests to Run and Set Default Fix Option

#	Step	Description
4	**Select the Analyze tests that you want to run.**	For instructions on how to select Analyze tests to run, see: • Selecting Tests to Execute
5	**Set the default fix option for selected tests: basic fix, advanced fix, or do not fix.**	For instructions on how to set the default fix option, see: • Setting Automatic Fix Preferences for Operating System Compatibility Tests

Perform Testing and View Results

For instructions on how to perform testing and view results, see the following help topics:

Table 3-5 • Perform Testing and View Results

#	Step	Description
6	**Click Execute Tests to test Windows Installer packages, and mobile apps.**	For information on executing tests, see: • Performing Compatibility, Best Practices, and Risk Assessment Testing
8	**View test results.**	For information on viewing test results, see: • Viewing and Filtering Test Results • Filtering Test Results by Suppressing Errors/Warnings
9	**Click Resolve Issues to automatically resolve issues.**	For information on resolving issues, see: • Resolving Issues

Migrate to Application Virtualization Tab

The flowchart on this tab outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise.

Note • *Prior to performing these steps, you should have already set up infrastructure as outlined in the Set Up Infrastructure Tab.*

To migrate a Windows Installer or legacy application to a virtual package, perform the following steps:

- Identify Packages to Virtualize
- Convert to Virtual Formats
- Test and Distribute Converted Packages

Identify Packages to Virtualize

For instructions on how to identify the packages to virtualize, see the following help topics:

Table 3-6 • Identify Packages to Virtualize

#	Step	Description
1	**Import packages into Application Catalog.**	For instructions on how to import Windows Installer and legacy applications into the Application Catalog, see: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) • About Legacy Installer Packages
2	**View each package's Application Virtualization Compatibility test results.**	For instructions on how to view a package's Application Virtualization Compatibility test results, see Viewing Application Virtualization Compatibility Test Results.
3	**Identify candidates for virtualization.**	Based upon the virtualization readiness information reported, decide which of the imported applications to select for virtualization.

Convert to Virtual Formats

For instructions on how to convert a package to a virtual format, see the following help topics:

Table 3-7 • Convert to Virtual Formats

#	Step	Description
4	**Import candidate packages into Automated Application Converter.**	For instructions on how to import candidate packages into Automated Application Converter, see: • Adding Packages from an AdminStudio Application Catalog • Adding Packages from a Local Machine or Network
5	**Convert to virtual packages.**	For instructions on convert selected packages to virtual packages, see Performing a Conversion Using the Application Conversion Wizard.
6	**Test launch virtual packages.**	For instructions on how to test the virtual packages that you have just created, see Testing Packages.
7	**Publish virtual packages to Application Catalog.**	For instructions on how to publish converted packages to the Application Catalog, see Importing Converted Packages into the Application Catalog.

Test and Distribute Converted Packages

For instructions on how to test and distribute converted virtual packages, see the following help topics:

Table 3-8 • Test and Distribute Converted Packages

#	Step	Description
8	**Perform virtualization best practice testing.**	For instructions on how to validate App-V packages against virtualization best practice rules, see Performing Compatibility, Best Practices, and Risk Assessment Testing.
9	**Perform conflict testing.**	For instructions on how to perform conflict analysis of an App-V packages against other packages, see Performing Application Conflict Testing.
10	**Edit App-V packages (if necessary).**	To resolve any warnings or errors that were found during testing, you can edit the App-V package in the Virtual Package Editor, as described in Using the Virtual Package Editor.
11	**Distribute to enterprise for user acceptance testing and production.**	For instructions on how to distribute an App-V package to your enterprise, see Distributing Applications and Packages.

Migrate to Windows Installer Tab

The flowchart on this tab outlines the steps required to migrate legacy setups (such as `.exe` files) to deployable Windows Installer packages (`.msi`).

Note • *Prior to performing these steps, you should have already set up infrastructure as outlined in the Set Up Infrastructure Tab.*

To migrate a legacy setup to a Windows Installer package, perform the following steps:

- Repackage Legacy Package
- Import Into Application Catalog
- Test Repackaged Applications and Resolve Issues
- Distribute Repackaged Applications

Repackage Legacy Package

For instructions on how to repackage a legacy package, see the following help topics:

Table 3-9 • Repackage Legacy Package

#	Step	Description
1	**Select legacy packages (.exe).**	For instructions about how to get started with repackaging, see About Repackaging.
2	**Repackage to Windows Installer package (.msi).**	For instructions on how to repackage a Windows Installer package, see Repackaging Legacy Installations Using the Repackaging Wizard.
3	**Edit packages in Repackager.**	For instructions on how to edit a Repackager project, see Working With Repackager Projects.

Import Into Application Catalog

For instructions on how to import a package into the Application Catalog, see the following help topics:

Table 3-10 • Repackage Legacy Package

#	Step	Description
4	**Import Windows Installer package into Application Catalog.**	For instructions on how to import a Windows Installer package into the Application Catalog, see Importing a Single Package File.

Test Repackaged Applications and Resolve Issues

For instructions on how to test a repackaged application and resolve issues, see the following help topics:

Table 3-11 • Test Repackaged Applications and Resolve Issues

#	Step	Description
5	**Perform Windows Installer best practices and OS compatibility testing.**	For instructions on how to perform Windows Installer best practice and OS compatibility testing, see Performing Compatibility, Best Practices, and Risk Assessment Testing.
6	**Perform application conflict testing.**	For instructions on how to perform conflict testing, see Performing Application Conflict Testing.
7	**Review test results in Analyze.**	For information on viewing test results, see: • Viewing and Filtering Test Results • Filtering Test Results by Suppressing Errors/Warnings

Table 3-11 • Test Repackaged Applications and Resolve Issues

#	Step	Description
8	**Perform automatic issue resolution.**	For information on resolving issues, see: • Resolving Issues

Distribute Repackaged Applications

For instructions on how to distribute a repackaged application, see the following help topics:

Table 3-12 • Distribute Repackaged Applications

#	Step	Description
9	**Distribute repackaged application via ConfigMgr (Formerly called as System Center Configuration Manager) or using another distribution method.**	For instructions on how to distribute a package, see Distributing Applications and Packages.

Set Up Infrastructure Tab

To get started using AdminStudio, you need to connect to a Microsoft SQL Server Application Catalog database, and prepare virtual machines for use in automated repackaging and testing.

- Create/Connect to an Application Catalog
- Configure Virtual Machines
- Set E-Mail Notification Settings
- Enter Server/Database Connection Settings

Create/Connect to an Application Catalog

For instructions on create a new Application Catalog or connect to an existing one, see the following help topics:

Table 3-13 • Create/Connect to an Application Catalog

Step	Description
Create/Connect to an Application Catalog	For instructions on how to create or connect to an Application Catalog, see Creating New Application Catalogs or Connecting to an Existing Application Catalog.
Enable Software Repository	For instructions on enabling the Software Repository, see Software Repository Integration into Other AdminStudio Tools.

Table 3-13 • Create/Connect to an Application Catalog

Step	Description
Set Default Application Catalog	For instructions on setting the default Application Catalog, see Specifying a Default AdminStudio Application Catalog.

Configure Virtual Machines

For instructions on how to configure virtual machines for use with Automated Application Converter, see the following help topics:

Table 3-14 • Configure Virtual Machines

Step	Description
Run Virtual Machine Preparation Setup to Enable Auto Login	For instructions on how to run the virtual machine preparation setup, see Preparing Your Virtual Machines for Use With the Automated Application Converter and Running the Virtual Machine Preparation Setup.
Create a Snapshot	For instructions on how to create a snapshot on your virtual machine, see Taking a Snapshot.
Install VMware Vsphere	For instructions on how to install VMware Vsphere, see VMware vSphere API Requirement on the AdminStudio Machine.
Test Virtual Machine Setup by Converting a Simple Package	For instructions on how to test a virtual machine setup, see Using the Application Conversion Wizard to Perform Automated Package Conversion.

Set E-Mail Notification Settings

For instructions on how to set e-mail notification settings, see the following help topics:

Table 3-15 • Set E-Mail Notification Settings

Step	Description
Set SMTP Notification Settings	For instructions on setting SMTP notification settings, see Setting E-Mail Notification Settings.

Enter Server/Database Connection Settings

For instructions on how to enter connection settings to enable AdminStudio to communicate with your distribution systems and the Flexera Service Gateway, see the following help topics:

Table 3-16 • Enter Server/Database Connection Settings

Step	Description
Configure Multiple Named Connections to Distribution Systems	For information on entering connection information for ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp, Symantec Altiris, and Workspace ONE distribution systems, see Creating Multiple Named Connections to Distribution Systems.
Connecting to the Flexera Service Gateway	For information on connecting to the Flexera Service Gateway—which enables communication with FlexNet Manager Platform, Workflow Manager, and App Portal—see Integrating with Other Flexera Applications via the Flexera Service Gateway.

4

Using the AdminStudio Interface

The main application of the AdminStudio tool set is Application Manager, from which all other tools can be launched, as described in Managing Applications and Application Catalog Databases. However, from the AdminStudio interface (AdminStudio.exe), you can perform the following tasks:

- **Getting started**—You can learn about AdminStudio tools using the **Start Page** tab, as described in Getting Started with AdminStudio.
- **Process Assistants**—On the **Process Assistants** tab, you can use workflow functionality to define repeatable processes to accomplish your goals, as described in Workflows and Projects.
- **Connect to/Create Application Catalog databases**—You can connect to an existing or create a new Application Catalog database, as described in Managing Applications and Application Catalog Databases.
- **Launch tools**—In addition to being able to launch AdminStudio tools from the Windows Start menu or from the Application Catalog **AdminStudio Tools** menu, you can also launch tools from the **AdminStudio Tools** button.

Information about the AdminStudio interface is presented in the following sections:

Table 4-1 • AdminStudio Interface Documentation

Section	Description
Configuring the AdminStudio Interface	Includes information on customizing the AdminStudio Interface, including setting shared directories.
Workflows and Projects	Explains how to use workflow functionality to define repeatable processes to accomplish your goals.
Frequently Asked Questions	A list of questions frequently asked by AdminStudio users, with links to the appropriate help topics.
AdminStudio Interface Reference	This section contains a description of each dialog box, Wizard, and UI element in AdminStudio.

Opening the AdminStudio Interface

To open the AdminStudio interface, perform the following steps.

Task ***To open the AdminStudio interface:***

1. Open Application Catalog.
2. On the **Home** tab, click on the **AdminStudio Tools** button and then select **Process Assistants** from the menu. The AdminStudio interface is launched, with the **Start Page** tab selected.

Configuring the AdminStudio Interface

This section explains how to configure and customize the AdminStudio interface. The following topics are discussed:

Table 4-2 • Topics on AdminStudio Interface Configuration

Category	Topics
Launching Applications	• Launching AdminStudio Applications
Configuring Application Catalog Settings	• Specifying the AdminStudio Shared Location
Setting Interface Preferences	• Setting E-Mail Notification Settings • Setting the Workflow Task Help Page Location • Configuring How Often AdminStudio Checks for Updates • Configuring AdminStudio to Stay on Top • Generating a Debug Log for AdminStudio

Launching AdminStudio Applications

Individual AdminStudio applications, such as InstallShield Editor or Repackager, can be launched from the Application Manager ribbon.

Note • *You can also launch AdminStudio tools from the **AdminStudio Tools** menu on the **Home** tab of Application Catalog.*

For more information on AdminStudio Applications, see AdminStudio Tools Dialog Box.

Specifying the AdminStudio Shared Location

The AdminStudio Shared directory (also referred to as the AdminStudio Shared location) contains shared information for repackaging and conflict identification, and other AdminStudio functions. The AdminStudio Shared Directory contains the following:

- The Shared `AdminStudio.ini` file, which specifies default Application Catalog database settings
- Application Catalog duplicate package identifier options
- Repackager `isrepackager.ini` exclusion list
- OS Snapshot `issnapshot.ini` file
- User-defined ACEs used in conflict analysis
- Distribution Wizard Distribution Type templates and `.ini` files

If you are working in a team environment, the AdminStudio Shared Directory should be set to a centralized network location, accessible by all AdminStudio users at your organization, rather than on your local machine. Follow the steps below to specify the location of the AdminStudio Shared Directory.

Note • *To maintain consistency when creating workflows, it is recommended that you set the AdminStudio Shared Directory the same for each AdminStudio user.*

Task ***To specify the location of the AdminStudio Shared Directory:***

1. Launch the AdminStudio Interface.
2. From the **Tools** menu, select Options. The **Options** dialog box opens.
3. In the **Options** dialog box, select the **Locations** tab.
4. Enter or browse to the directory for the **AdminStudio Shared Location**.
5. Click **OK** to close the **Options** dialog box.

Setting E-Mail Notification Settings

To enable AdminStudio to send you e-mail notifications during various processes, you need to configure your SMTP notification settings.

Note • *Currently, e-mail notifications are sent when soft time-outs are encountered while using Automated Application Converter to repackage an application on a virtual machine.*

To set your e-mail notification settings, perform the following steps:

Task ***To set e-mail notification settings:***

1. Launch the AdminStudio interface.
2. From the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. Select the **Notification Settings** tab.

4. On the **Notification Settings** tab, enter the following information:

Option	Description
SMTP Server	Enter the address of your e-mail server, such as: `smtp.yourcompany.com`
Authentication	Specify how your e-mail is authenticated by selecting one of the following options: • **Server Authentication**—Select this option if you want to perform server authentication on your AdminStudio e-mail. • **Anonymous**—Select this option if you do not want to perform authentication on your AdminStudio e-mail.
Domain	Enter the Domain of the user account listed in the **User Name** field.
User Name	Enter the name of an existing user account in the Domain specified in the **Domain** field. This user must have permission to send e-mail.
Password	Enter the password of the user account defined in the **User Name** and **Domain** fields. ***Note*** • *If your network domain requires that user passwords are changed periodically, you will have to open this dialog box again to update this account's password. To avoid this, try to obtain a user account that has a password that does not expire.*
From E-Mail ID	Enter the e-mail address to serve as the identity of AdminStudio. All e-mails sent by AdminStudio will have this e-mail address in the **From** field.
To E-Mail ID(s)	Enter the e-mail address to serve as the system account for AdminStudio e-mail. All e-mails sent to AdminStudio will be sent to this address. ***Note*** • *Use a semi-colon to separate multiple e-mail addresses in the **To E-Mail ID(s)** field.*
SMTP Server Port	Enter the port of your SMTP server.
Use SSL	Select this option if you want to use SSL security for the AdminStudio e-mail account.

5. Click **OK**.

Setting the Workflow Task Help Page Location

The **Task Help Page Location** is the directory where you want to store all HTML pages that serve as workflow task instructions.

Task ***To specify the location of task help pages:***

1. Launch the AdminStudio interface.
2. From the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. Select the **Locations** tab.
4. Enter or browse to the **Task Help Page Location**, the location where workflow task help pages are stored.
5. Click **OK** to close the **Options** dialog box.

Configuring How Often AdminStudio Checks for Updates

You specify how often you want AdminStudio to check for updates on the **Updates** tab of the **Options** dialog box.

Task ***To configure how often AdminStudio checks for updates:***

1. Open AdminStudio.
2. On the **Tools** menu, click **Options**. The **Options** dialog box opens.
3. On the **Updates** tab, select the frequency that AdminStudio will check for software updates:
   - Once every 15 days
   - Once every 30 days
   - Once every 60 days
   - Never
4. Click **OK** to close the **Options** dialog box.

Configuring AdminStudio to Stay on Top

When you launch AdminStudio tools, you can specify whether they open in front of or behind the AdminStudio interface. If you select **Always on Top** from the **View** menu, application will always open behind the AdminStudio interface.

Task ***To configure AdminStudio to stay on top of other applications:***

1. Launch the AdminStudio interface.
2. From the **View** menu, select **Always on Top**.

Generating a Debug Log for AdminStudio

To create a debug log for AdminStudio, perform the following steps.

Task ***To generate a debug log for AdminStudio:***

Use the following registry value to turn debugging on. Once this debugging is turned on, a log file will be created in the same location as the `.exe` file.

```
[HKLM\Software\InstallShield\AdminStudio] DebugLogLevel="3"
```

Levels 0, 1, 2, 3, 4, and 5 are supported with 5 being the highest. Default is level 0.

Workflows and Projects

Workflows, which can be created and modified using the **Process Template Editor**, are the basis for all projects in AdminStudio. These workflows consist of defined tasks, with which instructions (in the form of HTML files) and tools can be associated. Users can then create projects based on these workflows, and execute them—following the specific steps defined in the workflow. This allows you to create specific, repeatable procedures to accomplish your application migration goals.

Important • *Starting with AdminStudio 10, the* ***Workflow Templates*** *tab of the AdminStudio interface has been moved into its own tool named* ***Process Template Editor****, which can be launched from the* ***AdminStudio Tools*** *button. All functionality remains the same. Also, the former* ***Projects*** *tab of the AdminStudio interface has been renamed to* ***Process Assistants****.*

Tip • *If you update a workflow, all projects based on that workflow will reflect the changes made to the workflow.*

The following topics relate to workflows and projects:

- Creating and Editing Workflows
- Creating and Using Projects
- Saving Workflow and Project Changes
- Workflow Project Example: Using the New Workflow Project Wizard
- Workflows, Projects, and Permissions

Creating and Editing Workflows

Workflows serve as templates upon which projects are based. Typically, only a few individuals create workflows, while others create projects and execute the projects to accomplish the workflow goal.

The following topics relate to creating and executing workflows:

- Creating New Workflows
- Renaming Workflows
- Filtering Workflows
- Deleting Workflows

- Creating New Tasks
- Modifying Task Properties
- Creating Notes for a Task
- Renaming Tasks
- Reordering Tasks
- Associating Help Files with Tasks
- Deleting Tasks
- Adding New Tools from the Process Template Editor

Creating New Workflows

To create a new workflow, perform the following steps.

Task ***To create a new workflow:***

1. Open the **Process Template Editor** by clicking its icon on the **AdminStudio Tools** button and then selecting **Process Template Editor** from the menu. You are prompted to connect to an Application Catalog.
2. Enter the Application Catalog connection information and click OK. The Process Template Editor interface opens.
3. Right-click in the Workflows tree pane and select New Workflow. A new Workflow is listed.
4. Provide a name for the workflow.

Renaming Workflows

To rename a workflow, perform the following steps.

Task ***To rename an existing workflow:***

1. Open the **Process Template Editor**.
2. Right-click the workflow you want to rename and select **Rename** from the shortcut menu.
3. Provide a new name for the workflow.

Filtering Workflows

To display a specific workflow, perform the following steps.

Task ***To display a specific workflow:***

1. Open the **Process Template Editor**.
2. From the drop-down menu above the Workflows tree, select the workflow you want to display.

Deleting Workflows

To delete a workflow, perform the following steps.

Task ***To delete an existing workflow:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, right-click the workflow you want to delete and select **Delete** from the shortcut menu.
3. Confirm the deletion by clicking Yes in the resulting dialog box.

Creating New Tasks

To create a new task, perform the following steps.

Task ***To create a new task:***

1. Open the **Process Template Editor**.
2. Right-click the workflow to which you want to add the task and select New Task. Alternatively, right-click on a task and select New Task to create a subtask.

   A new task appears named NewTask*nn*, and the Task Properties view for that task is displayed.
3. Enter a name for the new task.
4. Modify properties for the task.

Modifying Task Properties

To modify task properties, perform the following steps.

Task ***To modify properties for an existing task:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task you want to modify. The Task Properties view for the selected task is displayed.
3. Change Task Properties as necessary for the task.

Creating Notes for a Task

To create notes for a task, perform the following steps.

Task ***To create notes for a task:***

1. Select the **Process Assistants** tab in the Interface.
2. From the Projects tree, select the task to which you want to add notes. The Project Task Properties view appears for the selected task.
3. Enter notes in the Notes field.

Tip • *You can also add notes to a task in the **Process Template Editor**. If you do, all projects based on that workflow will use the notes you enter as the default notes for the specific task.*

Note • *There is a 255 character limit on notes.*

Renaming Tasks

To rename a task, perform the following steps.

Task ***To rename an existing task:***

1. Open the **Process Template Editor**.
2. Right-click the task you want to rename and select **Rename** from the shortcut menu.
3. Provide a new name for the task.

Reordering Tasks

To reorder a task, perform the following steps.

Task ***To change the task order:***

1. Open the **Process Template Editor**.
2. In the Workflows tree, select the task you want to move.
3. From the toolbar, click Move Up or Move Down to change the order in which tasks are performed. Click Move Right to make a task a subtask of another task; click Move Left to promote a task.
4. Repeat the previous steps as necessary.

Associating Help Files with Tasks

To associate a help file with a task, perform the following steps.

Task ***To associate a help file with a task:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task with which you want to associate the help file. The Task Properties view appears for the selected task.
3. In the Help File field, enter the name and location of the help file, or click Browse and navigate to it.
   - The help file can either be local, or you can use a URL (for example, http://www.mycompany.com/myURL.htm).
   - You can also click the Edit HTML button to the right of the Browse button to open a default HTML page in an HTML editor as a starting point.

Note • *Help files must be in HTML format.*

Deleting Tasks

To delete a task, perform the following steps.

Task ***To delete an existing task:***

1. Open the **Process Template Editor**.
2. In the Workflows tree, right-click the task you want to remove and select Delete.
3. From the resulting dialog box, click Yes to confirm the deletion.

Adding New Tools from the Process Template Editor

To add a new tool to the Tools list from the Process Template Editor, perform the following steps.

Task ***To add a new tool to the Tools list from the Process Template Editor:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task with which you want to associate the new tool. The Task Properties view appears for the selected task.
3. From the Tool list in the Task Properties view, select <Add Tool ...>. The Add New Tool dialog box opens.
4. In the **Add New Tool** dialog box, enter properties about the tool.

5. Click OK.

Creating and Using Projects

Projects, which are based on existing workflows, are the procedures followed to accomplish a set goal. Projects may include instructions describing what to do, and perhaps links to tools necessary to perform tasks. They also allow you to provide notes to help document issues that may arise during a project.

Tip • *If you update a workflow, all projects based on that workflow will reflect the changes made to the workflow.*

The following topics relate to creating and using projects:

- Creating Projects with the New Workflow Project Wizard
- Filtering Projects
- Executing Projects
- Running Associated Tools in Projects
- Deleting Projects

Creating Projects with the New Workflow Project Wizard

To create a new project, perform the following steps.

Task ***To create a project using the New Workflow Project Wizard:***

1. Launch AdminStudio.
2. Click the **Process Assistants** tab.
3. Right-click in the Projects tree and select **New Project**. The **New Workflow Project Wizard** launches.
4. From the **Welcome Panel**, click **Next**. The **Workflow Selection Panel** appears.
5. From the **Workflow Selection Panel**, select the workflow on which you want to base the new project.
6. Provide a name for the new project and click **Next**. The **Source Package** panel appears.
7. From the **Source Package Panel**, specify the name and location of the source package used in this project. Alternatively, click Browse to navigate to it.
8. Click **Next**. The **Target Directory and Filename** panel appears.
9. From the **Target Directory and File Name Panel**, specify the Target Directory in which you want to store all files associated with this project.
10. In the **Target File Name** field, provide a name for the output file. Depending on the task being executed, the appropriate extension will be added to the file name.
11. Click **Finish**. The new Project is now listed.

Filtering Projects

To display a specific project, perform the following steps.

Task ***To display a specific project:***

1. From the Interface, click the **Process Assistants** tab.
2. From the drop-down menu above the Projects tree, select the project you want to display.

Executing Projects

To execute a project, perform the following steps.

Task ***To execute a project:***

1. From the Interface, click the **Process Assistants** tab.
2. Display the project you want to execute. If you want to only display that project, use the filter above the Projects tree.
3. Click the first task in the project.
4. Perform the task.
5. When finished with the task, click the box to the left of the task.
6. Repeat for subsequent tasks in the project.

Running Associated Tools in Projects

To run associated tools in projects, perform the following steps.

Task ***To run a tool associated with a task:***

1. Select the **Process Assistants** tab from the Interface.
2. Expand a Project in the Projects tree to display all of its tasks.
3. Right-click the task with which the tool is associated and select **Run Task** from the shortcut menu.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Deleting Projects

To delete a project, perform the following steps.

Task ***To delete an existing project:***

1. Click the **Process Assistants** tab in the Interface.
2. From the **Projects** tree, right-click the project you want to delete and select **Delete Project** from the shortcut menu.
3. Confirm the deletion by clicking **Yes** in the resulting dialog box.

Saving Workflow and Project Changes

Because AdminStudio uses a database (the Application Catalog) to store information involving Workflows and Projects, all changes are stored immediately. There is no need to "save" your modifications; AdminStudio performs this automatically.

Workflow Project Example: Using the New Workflow Project Wizard

The following basic example covers creating a workflow and project which takes advantage of command-line functionality available in AdminStudio.

Prior to creating projects, you must create a workflow on which to base the project. This workflow might involve few steps, or it might cover as broad of a task as repackaging a legacy installation, editing it in InstallShield Editor, customizing it in Tuner, performing application isolation, identifying and resolving conflicts, distributing it, and entering information about it into a third-party tracking system.

In this example, you are going to create a basic workflow involving three steps: repackaging a legacy installation, building a Windows Installer package in Repackager, and then importing that package into Application Catalog.

Creating a New Workflow

To create a new workflow, perform the following steps.

Task ***To create a new workflow:***

1. Open the **Process Template Editor**.
2. Right-click in the **Workflows** tree pane and select **New Workflow**. A new workflow is created.
3. Name the workflow `My Workflow Example`.
4. Right-click **My Workflow Example** and select **New Task**, and name the task `Repackage a legacy setup`.
5. With the **Repackage a Legacy Setup** task selected, make the following selections in the **Task Properties** pane:
   a. From the **Tool** list, select **Repackaging Wizard**.
   b. From the **Tool Configuration** list, select **Repackage a legacy setup**.

Note • *Selecting this tool configuration associates the following predefined command lines with this task:*

```
-app "[SourcePackage]" -pp "[TargetFileName]" -o "[TargetDir]"
-sb -app "[SourcePackage]" -pp "[TargetFileName]" -o "[TargetDir]"
-sn -app "[SourcePackage]" -pp "[TargetFileName]" -o "[TargetDir]"
```

6. Right-click **My Workflow Example** and select **New Task**, and name the task **`Build a Windows Installer package`**.
7. Right-click **My Workflow Example** and select **New Task**, and name the task **`Import package into Application Catalog`**.
8. With the **Import package into Application Catalog** task selected, make the following selections in the **Task Properties** pane:
   a. From the **Tool** list, select **Application Catalog**.
   b. From the **Tool Configuration** list, select **Import Package**.

Note • *Selecting this tool configuration associates the following predefined command lines with this task:*

```
-app -iwiz "[SourcePackage]"
```

Creating a Project Based on the Workflow

To create a project based on a workflow, perform the following steps.

Task ***To create a project based on your new workflow:***

1. Open AdminStudio and open the **Process Assistants** tab.
2. In the **All Projects** list, right-click and select **New Project**. The **Welcome** panel of the **New Workflow Project Wizard** opens.
3. Click **Next**. The **Workflow Selection** panel opens.
4. In the **Provide a name for the new project** field, enter **`My Sample Project`** and click **Next**. The **Source Package** panel opens.
5. Select **My Workflow Example** and click **Next**. The **Source Package** panel opens.
6. From the **Source Package Panel**, click **Browse** and select a legacy (`.exe`) installation program.

Note • *This value (the directory and package name) are stored in the* `SourcePackage` *variable, which is used by the command line in Repackager set when you created the workflow.*

7. Click **Next**. The **Target Directory and Filename** panel opens.
8. In the **Target Directory** field, browse to the directory where you want to store files associated with your project. For this example, use **`C:\AdminStudio Shared\Test\WorkflowExample`**.

Note • *This value is written to the* `TargetDir` *variable used in the command line set for InstallShield Editor when creating the workflow.*

9. Set the **Target File Name** to **`WorkflowProjectEx`**.

***Note** • This value is written to the `TargetFileName` variable used in the command line set for InstallShield Editor when creating the workflow.*

10. Click **Finish**. The new Workflow Project is now listed.

Running the Workflow

To run the workflow, perform the following steps.

Task ***To run the workflow:***

1. Expand the workflow **My Workflow Example** in the **Projects** tree.
2. Right-click on the **Repackage a Legacy Setup** task and select **Run Task** on the shortcut menu. The Repackaging Wizard opens.

   ***Note** • When Repackager launches, it reads the value* `SourcePackage` *to determine the file to repackage. It also reads* `TargetDir` *and* `TargetFileName` *to determine where to place the output and what to call the output file.*
3. On the **Welcome** panel, click **Next**.
4. On the **Method Selection** panel, select **Installation Monitoring** and click **Next**.
5. On the **Collect Product Information** panel, if a company name is not yet listed, enter a **Company Name**, such as **`My Company`** and click **Next**.
6. On the **Set Target Project Information and Capture Settings** panel, click **Start**. The **Repackaging** panel opens and repackaging begins.
7. When repackaging is complete, the new `WorkflowProjectEx.irp` file opens in the Repackager interface.
8. Return to the **Process Assistants** tab and mark the **Repackage a legacy setup** step complete.
9. Return to the Repackager interface and select **Repackaged Output** in the tree to open the **Repackaged Output** view.
10. Click **Build**. The Repackager project is built into a Windows Installer package.
11. Return to the **Process Assistants** tab and mark the **Build a Windows Installer package** step complete.
12. Right-click on the **Import package into Application Catalog** task and select **Run Task** on the shortcut menu. Application Catalog opens and the Import Wizard is launched.
13. Proceed with the steps in the Import Wizard to import the following package into the Application Catalog:

    `AdminStudio Shared\Test\WorkflowExample\MSI_Package\WorkflowProjectEx.msi`

    The **WorkflowProjectEx** application is now listed in the Application Catalog tree.
14. Return to the **Process Assistants** tab and mark the **Import package into Application Catalog** step complete.

Summary

This is just a brief example of how AdminStudio tools can be made aware of each other during a project. When crafting workflows, create command lines to streamline your projects.

Workflows, Projects, and Permissions

AdminStudio interface functionality (including workflows, projects, and the Tools Gallery) is directly influenced by user authorization and permissions. For example, Administrators can see all users and projects assigned to those users in AdminStudio. In the case of NT Groups, Administrators can see individual members of those groups in the **Process Assistants** tab. Further, Administrators can assign projects to users when running the **New Workflow Project Wizard**.

An example of how permissions affect workflows and projects is the availability of the **Process Template Editor**, which requires the **View Workflow Tab** permission. Likewise, only users with the **Create Project** permission can create projects. Even if you have permission to view and create workflows, you can only associate tools which you are permitted to use with tasks you create. If you are executing projects, you can only launch tools you have permissions to use, regardless of whether they are associated with a task in the workflow.

For more information, see Managing Roles and Permissions.

Frequently Asked Questions

The following is a list of questions frequently asked by AdminStudio users, including a link to the appropriate help topic.

General & Workflow

- How do I specify a default Application Catalog? See Specifying a Default AdminStudio Application Catalog.

Application Isolation Wizard

- How do I isolate applications? See Isolating Applications Using Application Isolation Wizard.
- How do I modify the default isolation recommendations? See Modifying the Default Isolation Recommendations.

Application Catalog

- What types of conflicts can the Conflict Wizard detect? See Application Conflicts Tests.
- How do I change which conflicts are checked? See Selecting Tests to Execute.
- How do I perform best practices testing? See Performing Compatibility, Best Practices, and Risk Assessment Testing.
- How do I identify conflicts? See Performing Application Conflict Testing.
- How do I automatically resolve issues? See Resolving Issues.

Tuner

- What should I do If MSI prevalidation fails? See Handling Invalid Windows Installer Packages.
- How do I prevent a feature from displaying during custom installation? See Changing a Feature's Visibility.
- When should I use the Dialogs View instead of MSI command-line options? See Dialogs View vs. Command-Line Options.
- How do I create a `setup.exe` file for my package and transform? Creating a Setup.exe File for the Package and Transform.
- When do I use Tuner vs. InstallShield Editor? See Customizing Installations Using Tuner.

Repackager

- Why do people use a Repackager? See Repackaging Legacy Installations Using the Repackaging Wizard.
- Why is a “clean” system important for repackaging? See About Repackaging on Clean Systems.
- Should I repackage a Windows Installer (.msi) setup? See Repackaging Wizard Best Practices.
- How do I repackage a non-Windows Installer setup? Repackaging Legacy Installations Using the Repackaging Wizard and Converting Legacy Installations Using the Repackager Interface.
- How can I speed up repackaging? See Repackaging Wizard Best Practices.
- What can I do with a repackaged setup? See Repackaging Legacy Installations Using the Repackaging Wizard and Converting Legacy Installations Using the Repackager Interface.
- Where does Repackager store my repackaged files and the MSI packages it builds? See Set Target Project Information and Capture Settings Panel.
- How do I identify and fix WinINSTALL conversion problems? See Troubleshooting Guidelines for WinINSTALL Conversion.
- How do I identify and fix SMS conversion problems? See Troubleshooting Guidelines for SMS Conversion.
- What do I do if I receive a `ISDEV: fatal error -5023: Error building table file` error while using Repackager? See Resolving an “Error Building Table File” Error.

Distribution Wizard

- How do I create an administrative installation? See Creating Administrative Installations for Packages.

OS Snapshot Wizard

- Why do I need an OS Snapshot? See Taking OS Snapshots.

InstallShield Editor

Answers to common questions regarding InstallShield Editor can be found under **Frequently Asked Questions** in the InstallShield Editor Help Library.

AdminStudio Interface Reference

The AdminStudio interface reference is organized into the following areas:

- AdminStudio Start Page
- Process Assistants Tab
- Process Template Editor
- AdminStudio Menus and Toolbar
- Dialog Boxes
- Wizards
- Log Files

Opening the AdminStudio Interface

To open the AdminStudio interface, perform the following steps.

Task ***To open the AdminStudio interface:***

1. Open Application Catalog.
2. On the **Home** tab, click on the **AdminStudio Tools** button and then select **Process Assistants** from the menu. The AdminStudio interface is launched, with the **Start Page** tab selected.

AdminStudio Start Page

The AdminStudio Start Page provides quick access to product information, to recently opened files, and to InstallShield resources.

AdminStudio Start Page Tabs

The AdminStudio Start Page, which is designed to help you quickly get started evaluating and using AdminStudio, provides process information on how to perform key tasks using AdminStudio tools. Information is organized into the following tabs:

- **Getting Started**—Describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. See Getting Started Tab.
- **Migrate to Application Virtualization**—Provides a flowchart that outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise. See Migrate to Application Virtualization Tab.
- **Migrate to Windows Installer**—Provides a flowchart that outlines the steps required to migrate legacy setups (such as `.exe` files) to deployable Windows Installer packages (`.msi`). See Migrate to Windows Installer Tab.
- **Set Up Infrastructure**—Lists the infrastructure setup steps that you need to perform prior to using AdminStudio for the first time: connect to an Application Catalog, configure virtual machines, and set e-mail notification settings. See Set Up Infrastructure Tab.
- **Help & Support**—Provides sources for user documentation, support, and product information.

AdminStudio Views

The AdminStudio interface is organized into the following tabs, which appear across the top of the Start Page:

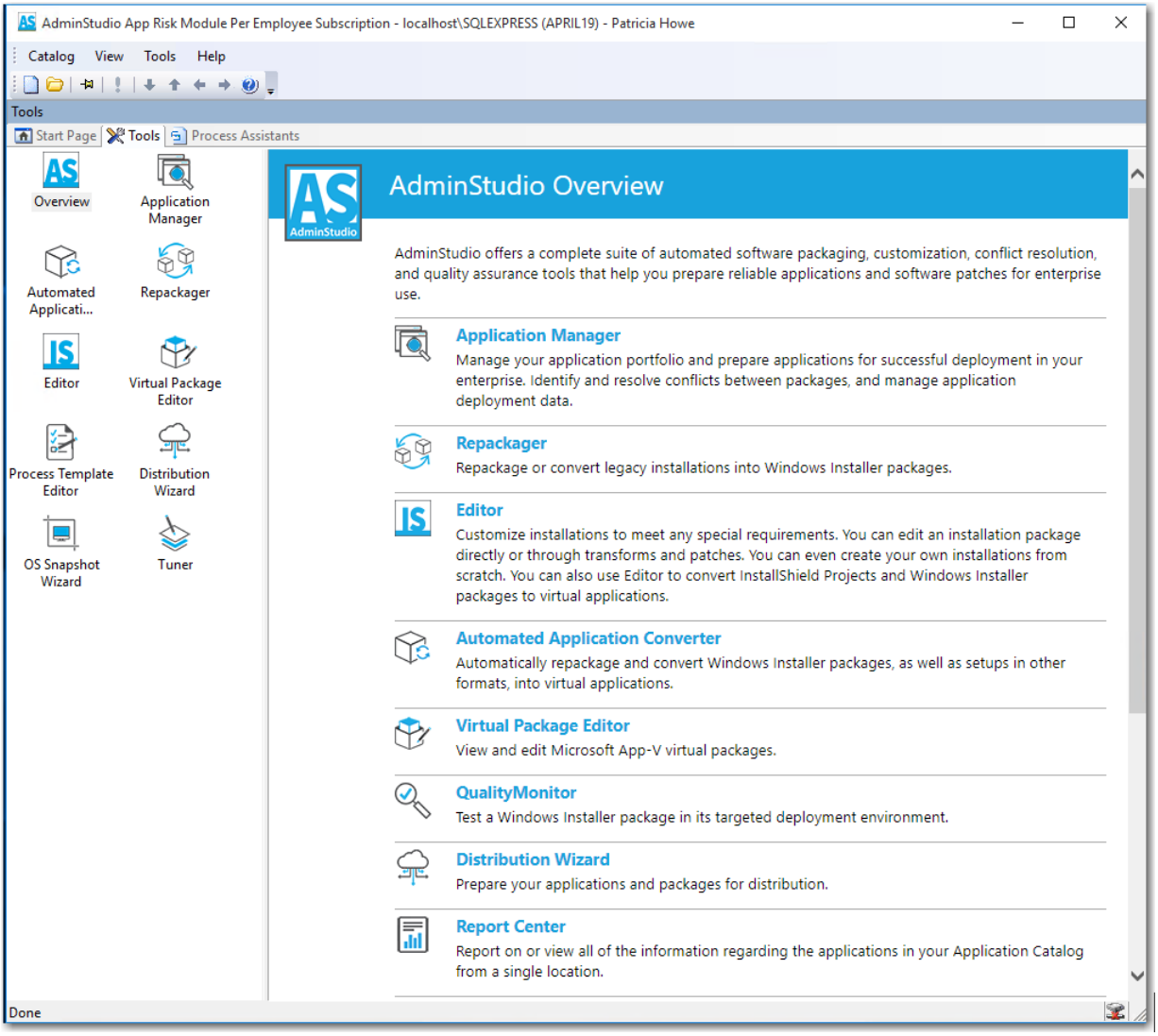


Figure 4-1: AdminStudio Interface Tabs

Click on these tabs to access the following AdminStudio views:

- **Start Page**—Initial view of AdminStudio.
- **Process Assistants**—On the Process Assistants Tab, you can create, execute, and delete projects, and access existing projects, which are the procedures followed to accomplish a set goal.

Opening the AdminStudio Interface

To open the AdminStudio interface, perform the following steps.

Process Assistants Tab

Purpose

From the Process Assistants tab, you can create, execute, and delete projects. You can also access existing projects. The drop-down filter above the Project pane allows you to view all available projects, or only a specific project.

Each project must be based on an existing workflow. In this way, projects are similar to photocopies from a master instruction sheet—all planning and design of the procedure is done to the workflow. The project is a copy of that workflow, and multiple projects can be based on the same workflow if you are performing the same procedure.

Integration with AdminStudio Workflow Manager

AdminStudio Workflow Manager is a Web-based application that manages the application lifecycle, incorporating standards (data) and methodologies (process). AdminStudio Workflows and Workflow Manager Workflows can be integrated, so that an AdminStudio Project can be a Workflow Phase in a Workflow Manager Workflow.

When an AdminStudio Project is linked to a Workflow Manager Workflow, please note the following indications on the Process Assistants tab:

- When an integrated AdminStudio project is selected on the Process Assistants tab, the name of its associated Workflow Manager Application is displayed in the Project Properties.
- When an AdminStudio Project is linked to a Workflow Manager Workflow and the workstation is not currently connected to the Workflow Manager Server, the following icon appears in the bottom right of the Process Assistants tab view:

Process Template Editor

Workflows, which can be created and modified using the Process Template Editor, are the basis for all projects in AdminStudio. These workflows consist of defined tasks, with which instructions (in the form of HTML files) and tools can be associated. Users can then create projects based on these workflows, and execute them—following the specific steps defined in the workflow. This allows you to create specific, repeatable procedures to accomplish your application migration goals.

For more information, see Workflows and Projects

You open the Process Template Editor by clicking the **Process Template Editor** icon on the **AdminStudio Tools** button or by selecting **Process Template Editor** from the Windows Start menu.

- Task Properties View

Task Properties View

Once you create a workflow, you can add tasks to it. Tasks are discrete steps in your overall process. Each task has the following configurable options:

Table 4-3 • Task Options

Option	Description
Tool	If needed, you can pick a tool to associate with the task. When a user runs the workflow, the tool can be launched from the workflow step. By default, the AdminStudio tools are included in this list. If you have added tools to the Tools Gallery, they also appear in this list. If you want to add a tool directly from the Process Template Editor, select the <New Tool> option to display the Add New Tool dialog box. This adds the tool to the Tools Gallery and makes it available for the current task. ***Note*** • *If, after adding a new tool for a task that is not included in the Tools Gallery, you assign a different tool or no tool to the task, the tool you added will no longer be available. To avoid this, when possible, add tools to the Tools Gallery*
Tool Configuration	This list contains all available command-line configurations for the selected tool. If you do not need a configuration, select <None>. Click Configure to add new configurations to the tool, which you can then select from this list.
Help File	You can associate a help file (in HTML format) with the task to provide instructions for performing the task. Enter the path and help file in this field, or use the Browse button to navigate to it. If you have yet to create an HTML page, click the Edit HTML button to the right of the Browse button to open a default page in an HTML editor.
Notes	Add any notes you want associated with this task. This field can only hold 255 characters, so additional information should be added to your help file.

AdminStudio Menus and Toolbar

The following commands and toolbar buttons are available in the AdminStudio interface.

Table 4-4 • AdminStudio Menus and Toolbar

Menu	Command	Shortcut	Button	Description
Catalog	**Connect**	Ctrl+O		Displays the Connect Application Catalog Dialog Box, where you can open an existing SQL Server Application Catalog or the AdminStudio Enterprise Server Application Catalog.
Catalog	**New**	Ctrl+N		Displays the Application Catalog Wizard, where you can create a new SQL Server Application Catalog database.

Table 4-4 • AdminStudio Menus and Toolbar (cont.)

Menu	Command	Shortcut	Button	Description
Catalog	**Disconnect**	Ctrl+D		Closes the currently open Application Catalog.
Catalog	**AdminStudio Enterprise Server - Change AES Password**			Change the password of the current user to log in to the AdminStudio Enterprise Server.
Catalog	**AdminStudio Enterprise Server - Logout**			Log out of the AdminStudio Enterprise Server.
Catalog	**Exit**	Alt+C+X		Exits AdminStudio and returns you to the Windows desktop.
Catalog	**AdminStudio Tools**			Select to open the AdminStudio Tools button, which includes the Tools Gallery and information on the selected tool.
Catalog	**Options**	Alt+T+O		Displays the **Options** dialog box, from which you can configure the location of shared resources and the frequency AdminStudio checks for updates.
Shortcut Menu	**New Workflow**			Create a new Workflow.
Shortcut Menu	**New Task**			Create a new Task.
Shortcut Menu	**Rename**			Rename selected Workflow or Task.
Shortcut Menu	**Delete**			Delete selected Workflow or Task.
Shortcut Menu	**New Project**		Ctrl-P	Create a new Project.
Shortcut Menu	**Del Project**			Delete selected Project.
Shortcut Menu	**Run Task**			Runs the tool associated with the selected task in the project.
Shortcut Menu	**Move Up**			Moves the selected task up in the task order.

Table 4-4 • AdminStudio Menus and Toolbar (cont.)

Menu	Command	Shortcut	Button	Description
Shortcut Menu	**Move Down**			Moves the selected task down in the task order.
Shortcut Menu	**Move Left**			Moves the selected task left in the task order.
Shortcut Menu	**Move Right**			Moves the selected task right in the task order.
Support	**Help Contents**	Alt+H+C		Launches the online Help Library and displays the Contents tab.
Support	**Help Index**	Alt+H+I		Launches the online Help Library and displays the Index tab.
Support	**Search Help**	Alt+H+S		Launches the online Help Library and displays the Search tab.
Support	**About Application Manager**	Alt+H+A		Displays the About dialog box with version information for AdminStudio Application Manager.
Support	**Check for Updates**	Alt+T+U		Determine if there any updates or messages available for AdminStudio.
View	**Status Bar**	Alt+V+S		Toggles the Status Bar.
View	**Output Window**			Displays testing results.

Dialog Boxes

The following dialog boxes can be opened from the AdminStudio Interface:

- About Dialog Box
- Add New Tool Dialog Box
- Command Line Properties Dialog Box
- Options Dialog Box
- Tool Properties Dialog Box

About Dialog Box

The **About** dialog box can be opened by selecting **About** from the **Help** menu of AdminStudio, Application Catalog, Automated Application Converter, Virtual Package Editor, or QualityMonitor.

This dialog box displays information about your installed version of AdminStudio, including the full version number (essential if you need technical support), and information on the edition and add-on packs that you have purchased.

The **About** dialog box includes the following information:

Table 4-5 • About Dialog Box Properties/Buttons

Properties/Buttons	Description
AdminStudio tool	The name of the specific AdminStudio tool that you have open is listed at the top of the **About** dialog box, such as Application Catalog, Automated Application Converter, etc.
Product name, edition, and add-on pack name	Below the AdminStudio tool name, the AdminStudio product name and edition is listed along with the add-on pack that was purchased (if any): • **Edition**—The edition can be Professional or Enterprise. • **Add-on packs**—Add-on packs can be Application Virtualization, Application Compatibility, Complete (which is Application Virtualization plus Application Compatibility), or Mobile.
Version	Lists the version of AdminStudio that you have installed.
Activation code	Lists the activation code that was used to activate AdminStudio.
OK	Click this button to close the dialog box.
Help	Click this button to open the AdminStudio help library page that describes this dialog box.
System Info	Click this button to open the Microsoft **Windows System Information** dialog box, which shows details about your computer's hardware configuration, computer components, and software (including drivers).
Upgrade	Click this button to open the AdminStudio **Product Activation** dialog box, where you can enter an activation code to upgrade your AdminStudio tier. For more information, see Upgrading Your Product Edition.

Add New Tool Dialog Box

The Add New Tool dialog box is displayed if you select <New Tool> from the Tool list while designing a workflow. This dialog box allows you to provide information about a tool you want accessible from AdminStudio, particularly to use in workflow tasks.

This dialog box contains the following options:

Table 4-6 • Add New Tool Dialog Box Options

Option	Description
Target	Enter the location of the tool or Browse to the application or file you want added to the tools pane for use in AdminStudio.

Table 4-6 • Add New Tool Dialog Box Options

Option	Description
Name in Tools Gallery	This description is used as the display name for the application in the Tools Gallery.
Command Line Arguments	Enter any command-line arguments for the tool. Because you may have different uses for applications, you can add the same application multiple times to the tools pane, with each instance using different command line arguments.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Enter any comments about this tool in this field.
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button ( ) below the field to open a default page in an HTML editor:
Add to Tools Gallery Check Box	When this box is checked, the tool will be added to the Tools Gallery. If unchecked, it is only available for the task where it was added.

Command Line Properties Dialog Box

The Command Line Properties dialog box is displayed when you create or edit a command-line configuration for a tool.

The Command Line Properties dialog box has two configurable options:

Table 4-7 • Command Line Properties Dialog Box Options

Option	Description
Description	Provide a description for the configuration. This assists you in differentiating similar command-line options.
Command Line	Provide the actual command-line parameters for the tool. The arrow to the left allows you to select one of the following variables to include in the command-line: • **InstallLocation**—The location where AdminStudio is installed. • **DevLocation**—The location where InstallShield Editor is installed. • **SharedPoint**—The AdminStudio shared directory. • **SourcePackage**—The name and location of the source package. • **TargetDir**—The directory where output from the selected project is stored. • **TargetFileName**—The name of the output file. • **ProjectName**—The name of the current project.

Options Dialog Box

From the **Options** dialog box, you can configure settings including application catalog settings, shared locations settings, and the frequency AdminStudio checks for updates. The dialog box consists of the following tabs:

- Locations Tab
- Updates Tab
- Quality Tab: Customer Experience Improvement Program
- Notification Settings Tab

Locations Tab

On the **Locations** tab of the AdminStudio **Options** dialog box, you specify the location of the AdminStudio Shared directory and the task help pages.

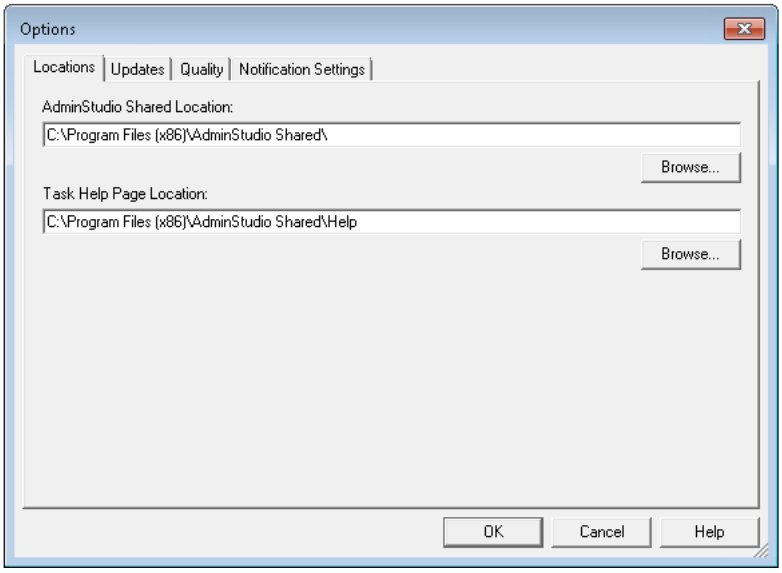


Figure 4-2: Options Dialog Box / Locations Tab

The **Locations** tab includes the following options:

Table 4-8 • Options Dialog Box/Locations Tab Options

Option	Description
AdminStudio Shared Location	Enter or browse to the shared location for AdminStudio. This location will contain shared information for repackaging and conflict identification. To maintain consistency when creating workflows, it is recommended that you set this shared location the same for each AdminStudio seat. ***Tip*** • *The **AdminStudio Shared Location** is defined during installation and normally does not need to be changed. It is usually assigned to a network folder, preferably a UNC path.*

Table 4-8 • Options Dialog Box/Locations Tab Options (cont.)

Option	Description
Task Help Page Location	Enter or browse to the directory where you want to store all HTML pages that serve as workflow task instructions.

Updates Tab

On the **Updates** tab of the AdminStudio **Options** dialog box, you can specify how frequently to check for AdminStudio software updates.

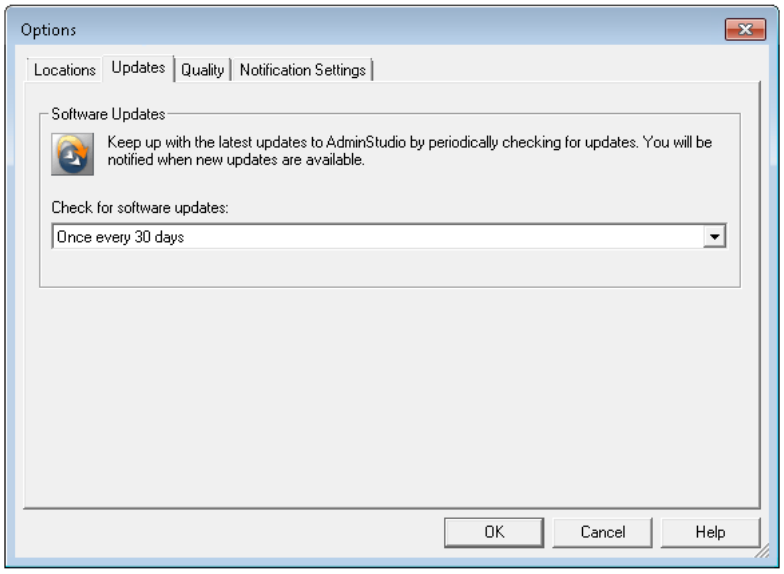


Figure 4-3: Options Dialog Box / Updates Tab

The **Updates** tab includes the following options:

Table 4-9 • Options Dialog Box/Updates Tab Options

Option	Description
Check for software updates	Specify how often you want AdminStudio to check for updates. Your options are: • Never • Once every 15 days • Once every 30 days (default) • Once every 60 days

Quality Tab: Customer Experience Improvement Program

The Customer Experience Improvement Program (CEIP) helps AdminStudio improve the quality, reliability and performance of our software and services.

If you choose to participate in the Customer Experience Improvement Program, we will collect anonymous information about how you use AdminStudio. This information helps us identify trends and usage patterns.

All information collected is anonymous, and this data collection will not affect the performance of AdminStudio tools. You will never be prompted to complete a survey, and no one from our company will contact you. You can continue working with AdminStudio without interruption.

Your membership status in the Customer Experience Improvement Program is specified on the **Quality** tab of the AdminStudio **Options** dialog box. If you initially select to participate but later you change your mind, you can opt-out of this program by changing the selection on the **Quality** tab.

Participation in the Customer Experience Improvement Program is not mandatory, but your input is appreciated.

For more information on the Customer Experience Improvement Program, visit the AdminStudio website.

Notification Settings Tab

On the **Notification Settings** tab, you can configure your SMTP notification settings. This will enable AdminStudio to send you e-mail notifications during various processes.

Currently, e-mail notifications are sent when soft time-outs are encountered while using Automated Application Converter to repackage an application on a virtual machine.

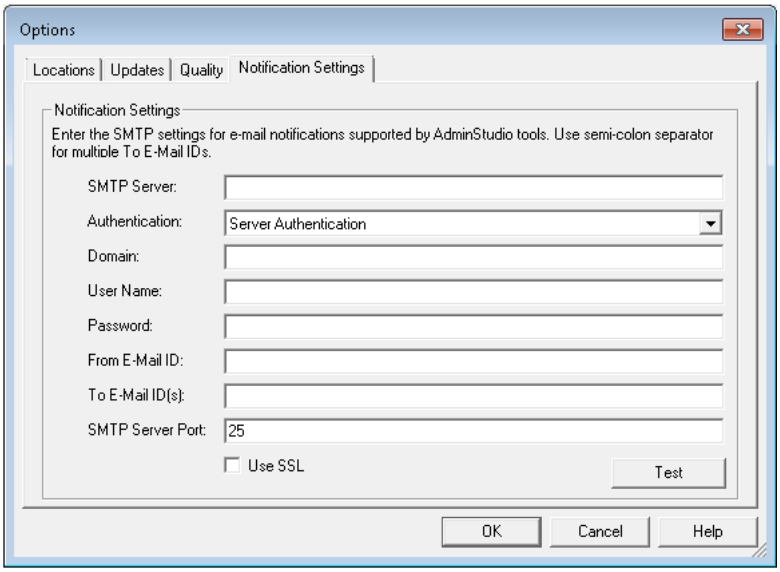


Figure 4-4: Notification Settings Tab of the Options Dialog Box

The **Notification Settings** tab of the **Options** dialog box has the following options:

Table 4-10 • Notification Settings Tab of the Options Dialog Box

Option	Description
SMTP Server	Enter the address of your e-mail server, such as: `smtp.yourcompany.com`

Table 4-10 • Notification Settings Tab of the Options Dialog Box

Option	Description
Authentication	Specify how your e-mail is authenticated by selecting one of the following options: • **Server Authentication**—Select this option if you want to perform server authentication on your AdminStudio e-mail. • **Anonymous**—Select this option if you do not want to perform authentication on your AdminStudio e-mail.
Domain	Enter the Domain of the user account listed in the **User Name** field.
User Name	Enter the name of an existing user account in the Domain specified in the **Domain** field. This user must have permission to send e-mail.
Password	Enter the password of the user account defined in the **User Name** and **Domain** fields. ***Note*** • *If your network domain requires that user passwords are changed periodically, you will have to open this dialog box again to update this account's password. To avoid this, try to obtain a user account that has a password that does not expire.*
From E-Mail ID	Enter the e-mail address to serve as the identity of AdminStudio. All e-mails sent by AdminStudio will have this e-mail address in the **From** field.
To E-Mail ID(s)	Enter the e-mail address to serve as the system account for AdminStudio e-mail. All e-mails sent to AdminStudio will be sent to this address. ***Note*** • *Use a semi-colon to separate multiple e-mail addresses in the **To E-Mail ID(s)** field.*
SMTP Server Port	Enter the port of your SMTP server.
Use SSL	Select this option if you want to use SSL security for the AdminStudio e-mail account.

Tool Properties Dialog Box

The Tool Properties dialog box is displayed when you right-click on a tool in the Tools gallery and select Properties. This dialog box contains the following tabs:

- Properties Tab
- Configuration Tab

Properties Tab

The **Properties** tab of the **Tool Properties** dialog box contains information about the tool, including the name and location of the executable, the name of the tool as it appears in the Tools gallery, and the help file associated with it (if any).

The following options can be configured:

Table 4-11 • Tool Properties Dialog Box/Properties Tab Options

Option	Description
Target	Enter the location of this tool's executable. Alternatively, click Browse and navigate to it.
Name in Tools Gallery	Provide a name for the tool as it will appear in the Tools gallery.
Command Line Arguments	Enter any default command line arguments for this tool.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Provide any comments about this tool.
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button below the field (shown below) to open a default page in an HTML editor:

Configuration Tab

From the **Configuration** tab of the Tool Properties dialog box, you can **Add**, **Modify**, or **Delete** command-line configurations associated with the tool. Each tool can have multiple configurations associated with it, for different uses.

Table 4-12 • Tool Properties Dialog Box/Configuration Tab Options

Option	Description
Command Line List	Listing of all command lines defined for this tool.

Table 4-12 • Tool Properties Dialog Box/Configuration Tab Options (cont.)

Option	Description
Add	Click to open the Command Line Properties Dialog Box, where you can add a new command line. In the Description field, you provide a description for the configuration. This assists you in differentiating similar command-line options. In the Command Line field, you provide the actual command-line parameters for the tool. The arrow to the left allows you to select one of the following variables to include in the command-line: • **InstallLocation**—The location where AdminStudio is installed. • **DevLocation**—The location where InstallShield Editor is installed. • **SharedPoint**—The AdminStudio shared directory. • **SourcePackage**—The name and location of the source package. • **TargetDir**—The directory where output from the selected project is stored. • **TargetFileName**—The name of the output file. • **ProjectName**—The name of the current project.
Modify	Click to open the Command Line Properties Dialog Box, where you can modify the selected command line.
Delete	Click to delete the selected command line.

Note • *Most AdminStudio Tools have one or more command-line configurations already defined. Although you can change or remove these configurations, there is no way to automatically reset them to their default values.*

Ways to Assign a Command Line to a Tool

There are three ways a user can assign a command line to a tool:

- **Tool Properties Panel of Add Tool Wizard**—When you add a tool, you can assign a command line on the **Tool Properties Panel** of the **Add Tool Wizard**.
- **Properties Tab of the Tool Properties Dialog Box**—When you view the tool's properties, you can assign a command line on the **Properties Tab** of the **Tool Properties** dialog box.
- **Configuration Tab of the Tool Properties Dialog Box**—Using the **Configuration Tab** of the **Tool Properties** dialog box, you can create *multiple* command lines and can use AdminStudio variables in these command lines. Then, when you go to the Workflow tab and create a new Workflow, you can associate a Tool with a task and also select which command line configuration they want to use. Once you have done that, you can go to Project tab and create a new Project. When you create a new Project, you will have to specify the Source Package and the target directory and file name. Once the Project is created, when you execute the task, AdminStudio will execute the command line configuration you previously selected by replacing the AdminStudio variables in the command line.

Note • *A command Line entered by the user in **Properties Tab** of the **Tool Properties** dialog box play no role in the Workflow and Project tab. This command line is used only when you run the tool from the AdminStudio Tools button.*

Wizards

The AdminStudio interface includes the following Wizards:

- Add Tool Wizard
- New Workflow Project Wizard

Add Tool Wizard

The Add Tool Wizard allows you to add new tools that appear in the AdminStudio Tools gallery. You can specify the tool's executable, provide command-line options for the tool, and provide a link to information about the tool.

The Add Tool Wizard includes the following panels:

- Welcome Panel
- Tool Properties Panel
- Command-Line Configurations Panel

Welcome Panel

The Add Tool Wizard allows you to add new tools that appear in the AdminStudio Tools gallery. You can specify the tool's executable, provide command-line options for the tool, and provide a link to information about the tool.

Click **Next** to proceed to the **Tool Properties Panel**.

Tool Properties Panel

From the Tool Properties panel of the Add Tool Wizard, you can enter information about the tool you are adding to the gallery.

You can configure the following options:

Table 4-13 • Add Tool Wizard/Tool Properties Panel Options

Option	Description
Target	Enter the location and name of this tool's executable. Alternately, click Browse and navigate to it.
Name in Tools Gallery	Provide a name for the tool as it will appear in the Tools Gallery.
Command Line Arguments	Enter any default command line arguments for this tool.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Provide any comments about this tool.

Table 4-13 • Add Tool Wizard/Tool Properties Panel Options (cont.)

Option	Description
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button below the field (shown below) to open a default page in an HTML editor:

Command-Line Configurations Panel

From this panel, you can add command-line configurations for the tool. Each tool can have multiple command-line configurations for different tool uses.

Click **Add** to add a new option from the **Command-Line Properties** dialog box. You can also click Modify to edit the selected command-line option, or Delete to remove the selected option.

Click **Finish** to add the tool to the Tools Gallery; click Back to return to the **Tool Properties Panel**.

New Workflow Project Wizard

The New Workflow Project Wizard assists you in creating a new project based on an existing workflow. The values you specify in this Wizard are stored in variables that can be accessed from tools, allowing for greater interoperability in AdminStudio. The New Workflow Project Wizard includes the following panels:

- Welcome Panel
- Workflow Selection Panel
- Source Package Panel
- Target Directory and File Name Panel

Welcome Panel

The New Workflow Project Wizard assists you in creating a new project based on an existing workflow. The values you specify in this Wizard are stored in variables that can be accessed from tools, allowing for greater interoperability in AdminStudio.

Workflow Selection Panel

From the Workflow Selection panel, you can specify the workflow on which you want to base this project. Available workflows appear in the workflows window.

Select the workflow you want to use, and provide the name for the new project (which is stored in the ProjectName predefined variable).

Source Package Panel

From the Source Package panel, you can specify the name and location of the source package used in this workflow. If you are creating a repackaging project, this is usually an executable, such as Setup.exe. The source package name and location is stored in the predefined variable SourcePackage.

Note • *You must specify the name and location of a source package if it is required in the workflow (by using the SourcePackage variable). For example, if the workflow specifies to launch a package with certain command-line parameters, AdminStudio needs to know what package to launch.*

Source packages can also be non-setup files. For example, if you are creating a simple workflow that involves editing a Notepad file, the source package may be a .txt file.

Target Directory and File Name Panel

From the **Target Directory and File Name** panel, specify the **Target Directory** and **Target File Name** used in this project.

All output files (such as an INC file from Repackager or an MST file from Tuner) associated with the project will be stored in the Target Directory, and the value for this directory can include a predefined variable such as `SharedPoint`. The Target File Name is the name used for all files created by project tasks in this project, with the appropriate extension appended to it depending on the file type. The Target Directory is stored in the predefined variable TargetDir and the output file name is stored in the variable `TargetFileName`.

Note • *If any workflow tasks use the TargetDir or TargetFileName variables, you must specify the target directory and package name. For example, if the workflow specifies to save a task's output, AdminStudio needs to know where to save it and what to call it.*

Click **Finish** to close the Wizard.

Log Files

AdminStudio tools generate the following log files:

Table 4-14 • AdminStudio Log Files

Log File	Location
AdminStudio.log	`C:\Program Files (x86)\AdminStudio\2020\Common`
distributer.log	`C:\Program Files (x86)\AdminStudio\2020\Common`
islc.log	`C:\Program Files (x86)\AdminStudio\2020\Repackager`

5

Managing Accounts and Directory Services

You can create an account for each person that should have access to Workflow Manager / AdminStudio Enterprise Server, or you can import accounts from a Windows Active Directory or Novell eDirectory directory service.

You can also configure various login methods to best suit your needs: account login, domain account login, and single sign-on login.

Information is presented in the following main sections:

- Managing Accounts
- Managing Directory Services Connections
- Managing Account Logins
- Accounts and Directory Services Reference

Managing Accounts

There are several ways to grant people access to Workflow Manager / AdminStudio Enterprise Server. You can:

- **Create an account**—Manually create an account for each person. See Creating a New Account.
- **Import directory service accounts**—Set up a Windows Active Directory or Novell eDirectory directory service connection, and import accounts from that directory service. See Importing Directory Services Accounts and Groups.
- **Import directory service group**—Set up a Windows Active Directory or Novell eDirectory directory service connection and import a group from that directory service. This allows you to provide dynamic access to all of the members of that group as the membership changes. For more information, see Importing Directory Services Accounts and Groups.

***Note** • For more information on the methods for logging into Workflow Manager / AdminStudio Enterprise Server, and how authentication is performed, see Managing Account Logins.*

Sample Workflow Manager Users

There are two main categories of Workflow Manager users: *consumers*, who make requests; and *administrators*, who perform the tasks to complete those requests. Consumer and administrator companies group these users.

- **In an internal environment**, the administrator company may be the IT department of a corporation, and consumer companies may be departments within that corporation.
- **In a consulting environment**, the administrator company is the organization performing the requested tasks, and the consumer companies are its clinets that make requests.

One administrator company usually does work for multiple consumer companies (or internal departments).

A user's assigned roles determine how much functionality is available to that user. Permissions for all of Workflow Manager's functions are assigned to roles, as described in Managing Roles and Permissions, and roles are assigned to users. The tasks that administrators and consumers can perform depend upon the permissions of their assigned roles.

To help you get started using Workflow Manager, sample users are automatically created during installation. The names of the users along with their assigned roles demonstrate typical Workflow Manager users.

Table 5-1 • Sample Workflow Manager Users With Their Assigned Roles

User Type	User Name	Role	How They User Workflow Manager
Consumer Company Users	user@company.com user@requester.com	User Application User	• Submit workflow requests • Monitor the progress of their workflow requests
	lm@company.com	License Manager	• Manage a company's license compliance
	cm@company.com	Configuration Manager	• Manages a company's software distribution.
	pm@requester.com	Project Manager	• Monitor the progress of all of the workflow requests submitted by his company
	tester@requester.com	UA Tester	• User acceptance tester.

Table 5-1 • Sample Workflow Manager Users With Their Assigned Roles

User Type	User Name	Role	How They User Workflow Manager
Administrator Company Users	repackager@servicer.com	Repackager	• Perform workflow request tasks • Create and view reports • Create and view issues
	techlead@servicer.com	Tech Lead	• Same permissions as the **Repackager** role but also has the permission to assign work
	admin@servicer.com	SCAdmin	• Assign work • Monitor the progress of all of the workflow requests that his company is working on • Create new consumer companies, and create new consumer and administrator user accounts • Create roles, projects, and templates • Assign permissions to roles • Assign roles to consumers and administrators for each workflow request • Communicate with customers via Workflow Manager

Filtering by Account Status

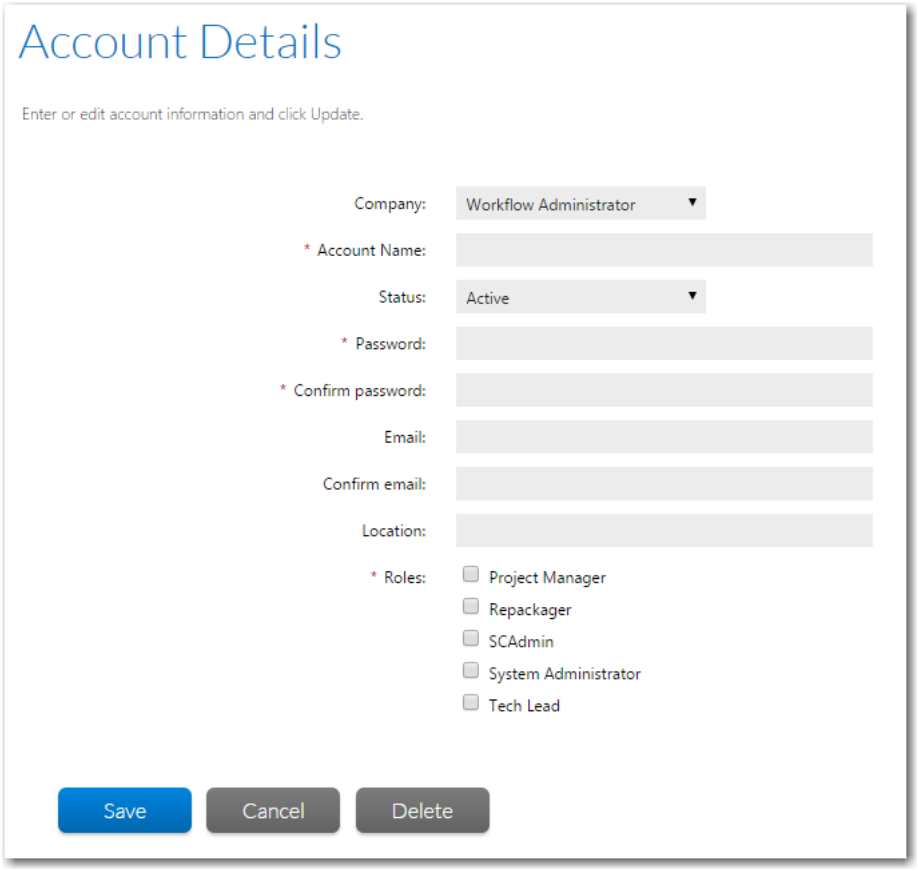
Some accounts may be disabled or inactive, meaning they can no longer access the system (see Disabling an Account). You may filter the **Account Administration** grid by this account status, by selecting either **Active**, **Inactive** or **All** (show all accounts, regardless of the status) from the **Status** list. The grid will automatically refresh once you make your choice.

Creating a New Account

If the people accessing Workflow Manager / AdminStudio Enterprise Server are not represented in a directory service, you will need to create login accounts for them manually.

Task ***To create a new account:***

1. On the **Settings** menu, click **Accounts and Groups**. The **Account Administration** page opens.
2. Click **Add**. The **Account Details** page opens.



3. Enter information to identify your new account as described on in Account Details Page.
4. Click **Save**.

Importing Directory Services Accounts and Groups

If you have defined a directory service connection, as described in Creating a New Directory Service Connection, you can choose to import accounts or groups from that directory service into Workflow Manager / AdminStudio Enterprise Server.

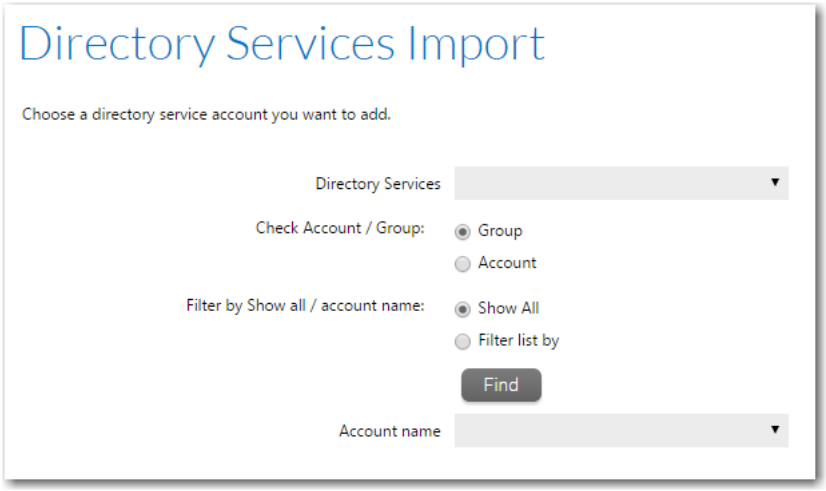
When an account is added to Workflow Manager / AdminStudio Enterprise Server from a directory service, only information that uniquely identifies the user in the directory service is stored. Information such as telephone number and email address will always be queried at run time, so that the most current details are obtained.

Workflow Manager / AdminStudio Enterprise Server supports Windows Active Directory and Novell eDirectory directory services.

Note • *For more information on support for directory services, see Managing Directory Services Connections and Managing Account Logins.*

Task ***To import accounts or groups from a directory service:***

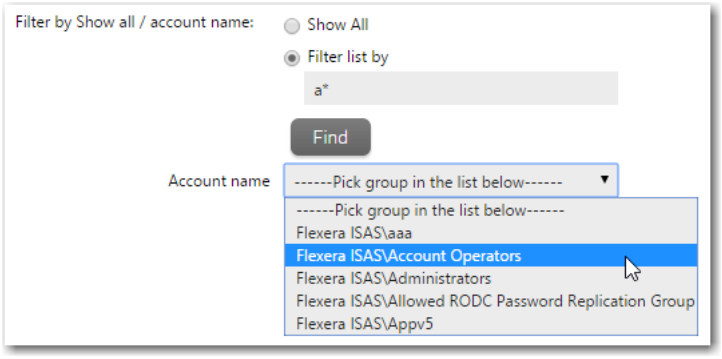
1. On the **Settings** menu, click **Accounts and Groups**. The **Account Administration** page opens.
2. Click the **Directory Service Import** button. The **Directory Services Import** page opens.



3. Choose the directory service containing the account or group you want to import from the **Directory Service** list.
4. Next to **Check Account / Group**, select the **Account** (to import a single account) or **Group** option.
5. Next to **Filter by Show all / account name** list, select one of the following:
   - **Show All**—Select this option to select your account from a list of all accounts and groups in the directory service.
   - **Filter list by**—Select this option to only return accounts and groups which match the criteria you specify, and enter your criteria in the associated text box.

     Click **Find** to filter the list of accounts and groups. For example, to search for all of the users that start with the letter P, use the asterisk wildcard character (*) and enter **P*** in the **Filter list by** box.

   In either case, click **Find** to return your chosen list of accounts and groups.



6. From the **Pick group/user in the list below** list, select the group or account to import into Workflow Manager / AdminStudio Enterprise Server. The **Account Details** page opens, providing a read-only view of the imported account or group's account name and password.

If you import an account that is a member of a previously-imported group, that account inherits its group's roles. You can then assign additional roles to that account.

See Account Details Page for more information about the **Account Details** page.

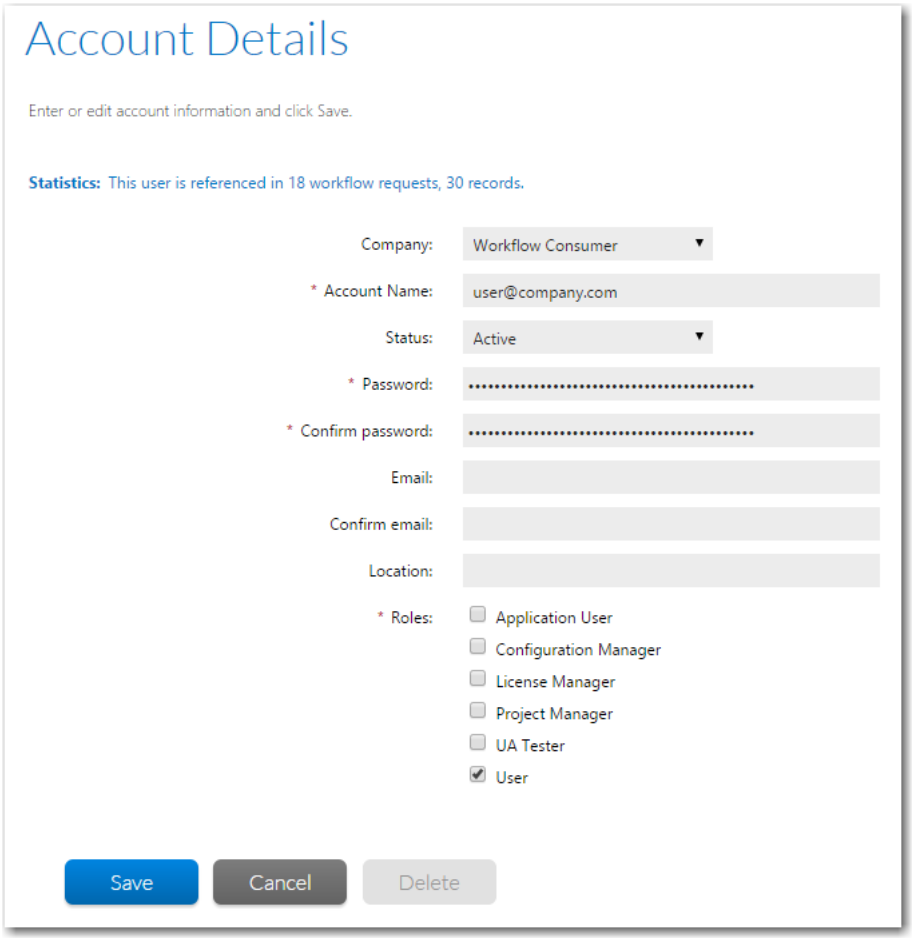
7. Click Save. The **Account Details** page closes and your new account or group appears in the list on the **Account Administration** page.

Viewing or Changing an Existing Account

You may view or update the details of any of the accounts visible on the **Account Administration** page.

Task ***To view or update an existing account:***

1. On the **Settings** menu, click **Accounts and Groups**. The **Account Administration** page opens.
2. Locate the account that you want to work with.
3. Click the user name to open the **Account Details** page.



4. View or update the account as required. See Account Details Page for more details.
5. Click **Save**.

Disabling an Account

Accounts may have a status of **Active** or **Inactive**. Inactive accounts:

- Cannot log in.
- Cannot be assigned any work.
- Are not listed on the **Account Administration** page.
- Do not get any email notifications.
- Cannot be selected as a **Consumer Contact** or **Administrator Contact** for a project.

Deleting vs. Disabling a User Account

If an account is not associated with a workflow request, you can delete that account. This means that if you create an account by accident, you can delete the account before it has a chance to interact with Workflow Manager / AdminStudio Enterprise Server. See Disabling an Account for details.

If a person is associated in any way with an open or completed workflow request, or with a project or workflow template, you cannot delete that person's account from the system. This is because Workflow Manager / AdminStudio Enterprise Server stores references to such accounts for historical, tracking and reporting purposes. If you no longer wanted this account to interact with the product, you would instead disable it by setting its **Status** to **Inactive**.

Task ***To disable a user account:***

1. Open the chosen account for editing, as described in Viewing or Changing an Existing Account.
2. On the **Account Details** page for that account, set the **Status** to **Inactive**, then click **Save**.

Deleting an Account

If an account is not referenced by any workflow requests or templates in the product, you can remove it from the system.

Task ***To delete an account:***

1. On the Settings menu, click **Accounts and Groups**. The **Account Administration** page opens.
2. Locate the account that you want to work with.
3. Click the user name to open the **Account Details** page.
4. Click **Delete**. You are prompted to confirm the deletion.

   Tip • *The **Delete** button will not be visible if the account is referenced by a workflow request or template.*

5. Click **SAve**. The **Account Details** page closes and the account you deleted is no longer listed on the **Account Administration** page.

Managing Directory Services Configuration

Workflow Manager / AdminStudio Enterprise Server can be integrated with Windows Active Directory and Novell eDirectory. This enables you to set up automatic login with the product based upon directory service authentication.

All directory service-related tasks can be managed starting from the **Directory Services** page, which you can access clicking **Directory Services** on the **Settings** menu.

Information about managing directory services is presented in the following sections:

- **Directory service connections**—If you import a directory services group, all members of that group can login to Workflow Manager / AdminStudio Enterprise Server without requiring you to import them individually. Workflow Manager / AdminStudio Enterprise Server can then retrieve attributes, such as email address or telephone number, from the directory service dynamically. For information on integrating with directory service users and groups, see Managing Directory Services Connections.
- **Directory service attributes**—[Workflow Manager only] You may add data elements to your workflow templates which, for accounts imported from directory services, will be directly populated from directory service attributes (such as account name, email address or location). For information on enabling the use of Directory Service Attributes, see Managing Directory Services Attributes.

Note • *For more information about Directory Services and Lightweight Directory Access Protocol (LDAP), see Lightweight Directory Access Protocol (LDAP) Overview on the Microsoft TechNet website.*

Managing Directory Services Connections

Rather than manually creating an account for each person who will use Workflow Manager / AdminStudio Enterprise Server, you can import accounts from Windows Active Directory or Novell eDirectory directory services. To integrate Workflow Manager / AdminStudio Enterprise Server with a directory service individual account or group, you need to set up a directory service connection.

This section includes the following topics:

- Creating a New Directory Service Connection
- Viewing or Changing an Existing Directory Service Connection
- Deleting an Existing Directory Service Connection

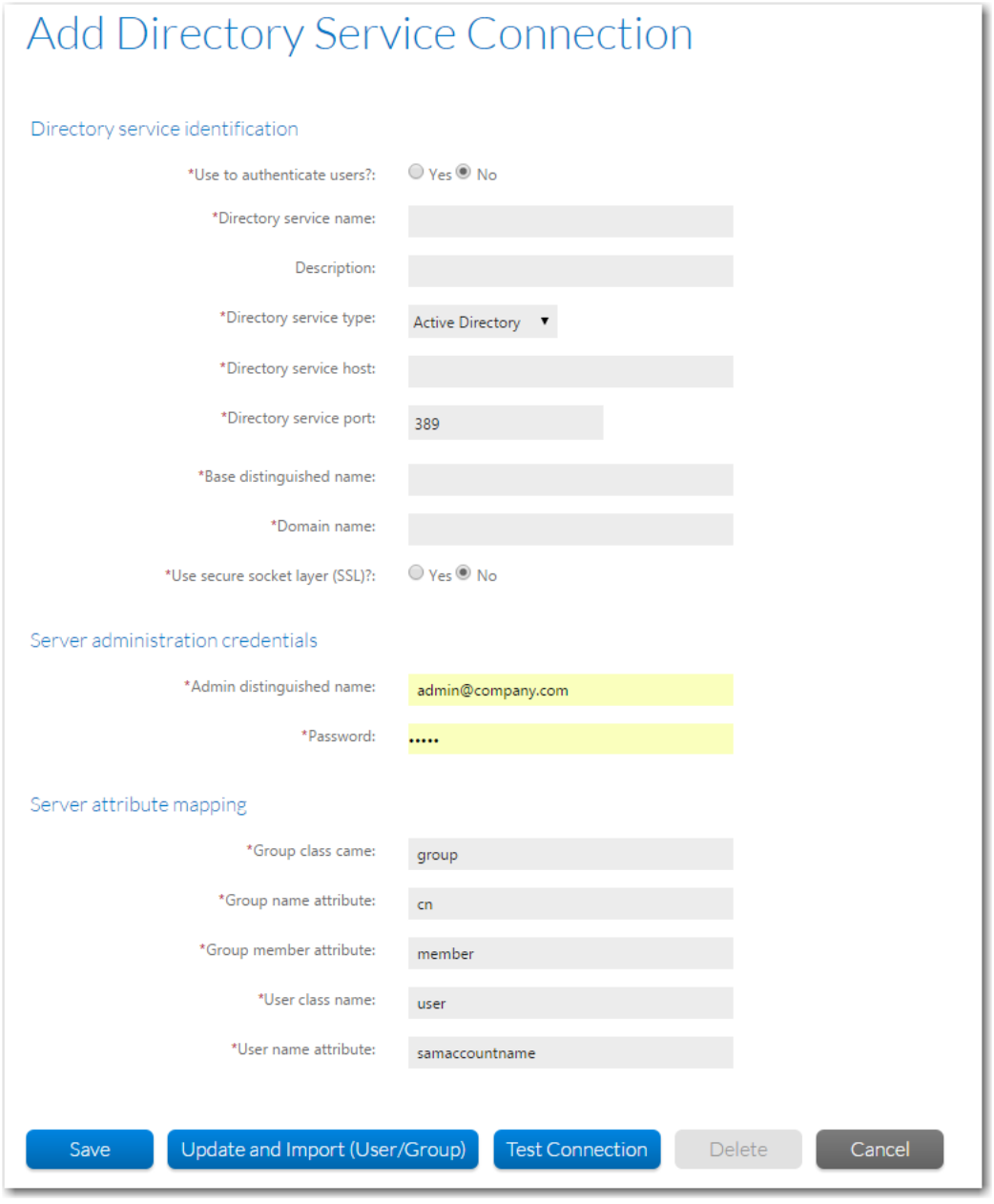
Creating a New Directory Service Connection

Directory services connections are used to import accounts into Workflow Manager / AdminStudio Enterprise Server, so as to authenticate Active Directory or eDirectory account-holders logging into Workflow Manager / AdminStudio Enterprise Server.

You can choose to have a directory service listed in the **Domain** list on the Workflow Manager / AdminStudio Enterprise Server login page. This enables people with accounts in the directory service to login to Workflow Manager / AdminStudio Enterprise Server using their enterprise network credentials

Task ***To add a Directory Service connection:***

1. On the **Settings** menu, click **Directory Services**. The **Directory Services Administration** page opens.
2. Click **Add**. The **Add Directory Service Connection** page opens.



3. Enter the relevant connection details. See Add Directory Service Connection Page for more information.
4. Click the **Test Connection** button to ensure the settings you entered can be used to successfully connect to this directory service.
5. Once your connection is successful, do one of the following:

- Click **Save** to save your new connection and return to the **Directory Services Administration** page
- Click **Update and Import (User/Group)** to save your new connection and open the **Directory Services Import** page, where you can immediately import an account for your connection. See Importing Directory Services Accounts and Groups for more information.

Viewing or Changing an Existing Directory Service Connection

You may view or update the details of any existing directory service connection.

Task ***To view or update an existing directory service connection:***

1. On the **Settings** menu, click **Directory Services**. The **Directory Services Administration** page opens.
2. Locate the directory service connection that you want to work with.
3. Click the directory service name to open the **Add Directory Service Connection** page opens.
4. View or update the connection as required. See Add Directory Service Connection Page for more information.
5. If you update the connection details, click the **Test Connection** button to ensure the settings you entered can still connect successfully to your directory service.
6. Do one of the following:
   - Click **Save** to save your new connection and return to the **Directory Services Administration** page.
   - Click **Update and Import (User/Group)** to save your new connection and open the **Directory Services Import** page, where you can immediately import an account for your connection. See Importing Directory Services Accounts and Groups for more information.
   - Click **Cancel** to close the **Add Directory Service Connection** page without saving your changes.

Deleting an Existing Directory Service Connection

In order to delete a directory service connection, all references to the directory service must be removed from Workflow Manager / AdminStudio Enterprise Server. If there are any accounts imported from the directory service or attributes associated with it, you will be unable to delete the connection.

Task ***To delete an existing directory service connection:***

1. Open the **Directory Service Administration** page by clicking **Directory Services** on the **Settings** menu.
2. Locate the directory service connection that you want to remove. The **Add/Edit Directory Service Connection** page opens.
3. Click **Delete**. You are prompted to confirm the deletion.

Tip • *The **Delete** button will be disabled if your connection is referenced by an account or attribute.*

4. Click **OK**. The **Add/Edit Directory Service Connection** page closes, and the connection you deleted is no longer listed on the **Directory Services Administration** page.

Managing Directory Services Attributes

When creating a Workflow Manager template, you can assign a **Data Element** the **Data Type** of **Directory Service**. This means that when a directory service-authenticated account completes the workflow step requesting that particular data element, information is pulled from the directory service to automatically populate the field, such as that account's name, email address, or location.

Each piece of information that can be returned from the directory service is referred to as an attribute. Workflow Manager allows you to select which of the many directory service attributes you want available for use as data elements in workflow templates.

Figure 5-1: Example of Fields Populated With Directory Services Attributes

Note • *If a person is using Workflow Manager through an account not authenticated from a directory service, such fields will be enabled and left blank, ready for manual entry.*

Information about managing directory service attributes is organized in the following topics:

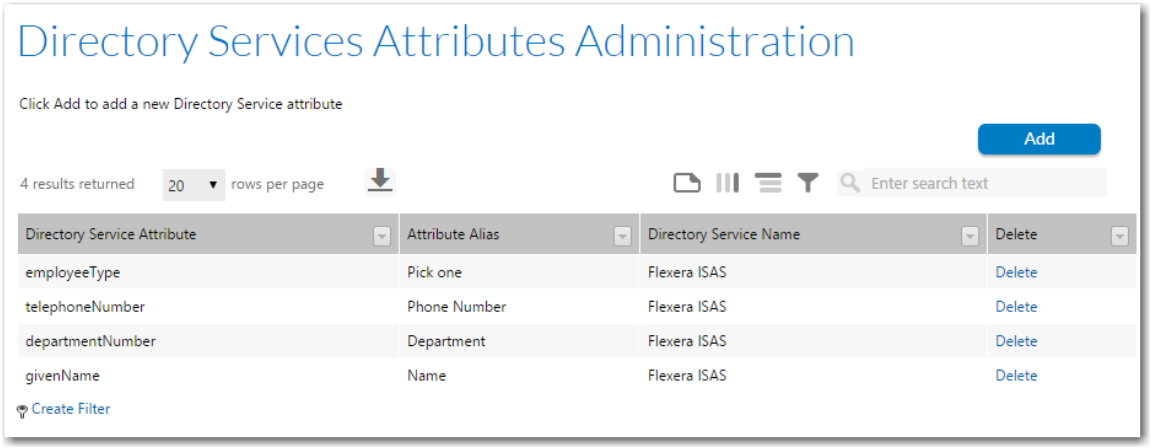
- Setting Up a New Directory Service Attribute
- Deleting an Existing Directory Service Attribute

Setting Up a New Directory Service Attribute

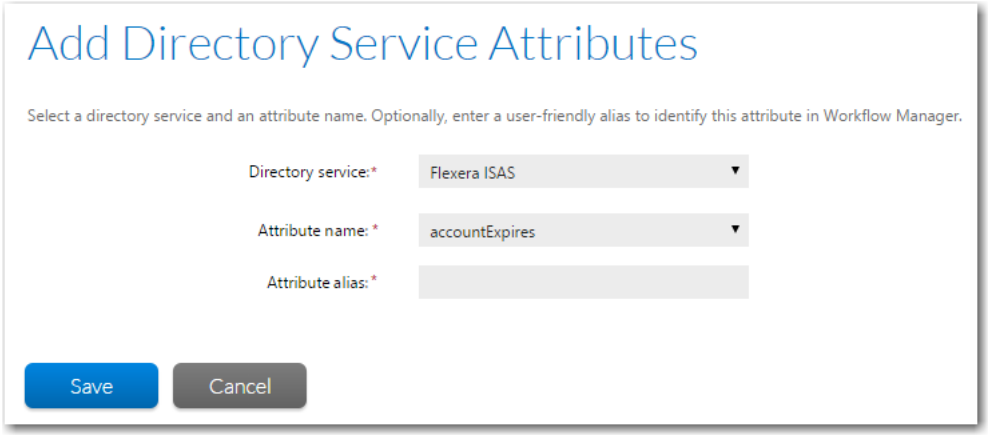
To make a new directory service attribute available for use in workflow templates, do the following.

Task ***To set up a new directory service attribute:***

1. Open the **Directory Service Attributes Administration** page by clicking **Directory Service Attributes** on the **Settings** menu. The **Directory Services Attributes Administration** page opens.



2. Click **Add**. The **Add Directory Service Attributes Page** page opens.



3. Select the directory service whose attribute you want to make available to workflow templates from the **Directory service** list. Refer to Creating a New Directory Service Connection for more information about setting up connections to your directory services.

4. The **Attribute name** field lists all of the directory service attributes from the chosen directory service. Select the one you want to add to the Workflow Manager database. An example of a directory service attribute might be `employeeNumber` or `documentAuthor`.

5. Enter a more user-friendly identifier for the attribute in the **Attribute alias** field. For example, you may want to identify the `documentAuthor` attribute as **`Author`** in Workflow Manager.

6. Click **Save** to register the attribute with Workflow Manager.

Tip • *If Workflow Manager is unable to connect to the server (so no attributes can be retrieved), the* ***Save*** *button is disabled.*

Deleting an Existing Directory Service Attribute

In order to delete a directory service attribute, all references to that directory service attribute must be removed from Workflow Manager. If there are any references to the directory services attribute, you will be unable to delete it.

Task ***To delete an existing directory service attribute:***

1. On the **Directory Services** page, click **Manage Directory Services Attributes**. The **Directory Services Attributes Administration** page opens
2. Locate the attribute that you want to remove.
3. Click the **Delete** hyperlink in the **Delete** column to the right of your chosen attribute. The attribute will be removed without prompting, so be sure you select the correct one.

Managing Account Logins

Workflow Manager / AdminStudio Enterprise Server allows you to login either with your domain account or with a different named account.

Each method is discussed in this section, as is how to set the session timeout value.

- Login Methods
- Using Account Login
- Using Domain Account Login
- Using Single Sign-On Login

Login Methods

There are multiple ways you can log in to the application:

Table 5-2 • Login Methods

Method	Description	Related Topics
Using Workflow Manager / AdminStudio Enterprise Server account	Login using an account created specifically for Workflow Manager / AdminStudio Enterprise Server.	• Using Account Login • Creating a New Account

Table 5-2 • Login Methods (cont.)

Method	Description	Related Topics
Using domain account credentials	Login using your domain account credentials. To set this up, you need to import accounts from a directory service (Active Directory or Novell eDirectory) into Workflow Manager/ AdminStudio Enterprise Server. ***Note*** • *To login using your domain account, ensure the **Anonymous Access** option on the IIS Manager **Authentication** view is set to **Enabled**. For more information, see Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On.*	• Using Domain Account Login • Creating a New Directory Service Connection • Importing Directory Services Accounts and Groups
Using single sign-on login	Be automatically logged in to Workflow Manager / AdminStudio Enterprise Server, based on your domain account credentials. This is referred to as *single sign-on* login. If you set up this option, IIS performs account authentication; in all other cases, authentication is the role of Workflow Manager / AdminStudio Enterprise Server itself. ***Note*** • *To login using the single sign-on method, ensure the **Enable anonymous access** option on the **IIS Authentication Methods** dialog box is not selected. For more information, see Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On.* ***Note*** • *Single sign-on is not supported for Novell eDirectory accounts.*	• Using Single Sign-On Login • Creating a New Directory Service Connection • Importing Directory Services Accounts and Groups
Using guest account login	Login using an anonymous guest account, set up to view a restricted set of Workflow Manager features – such as viewing reports or searching for a request.	• Using Guest Account Login • Setting Up a Guest Account • Logging in as a Guest

Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On

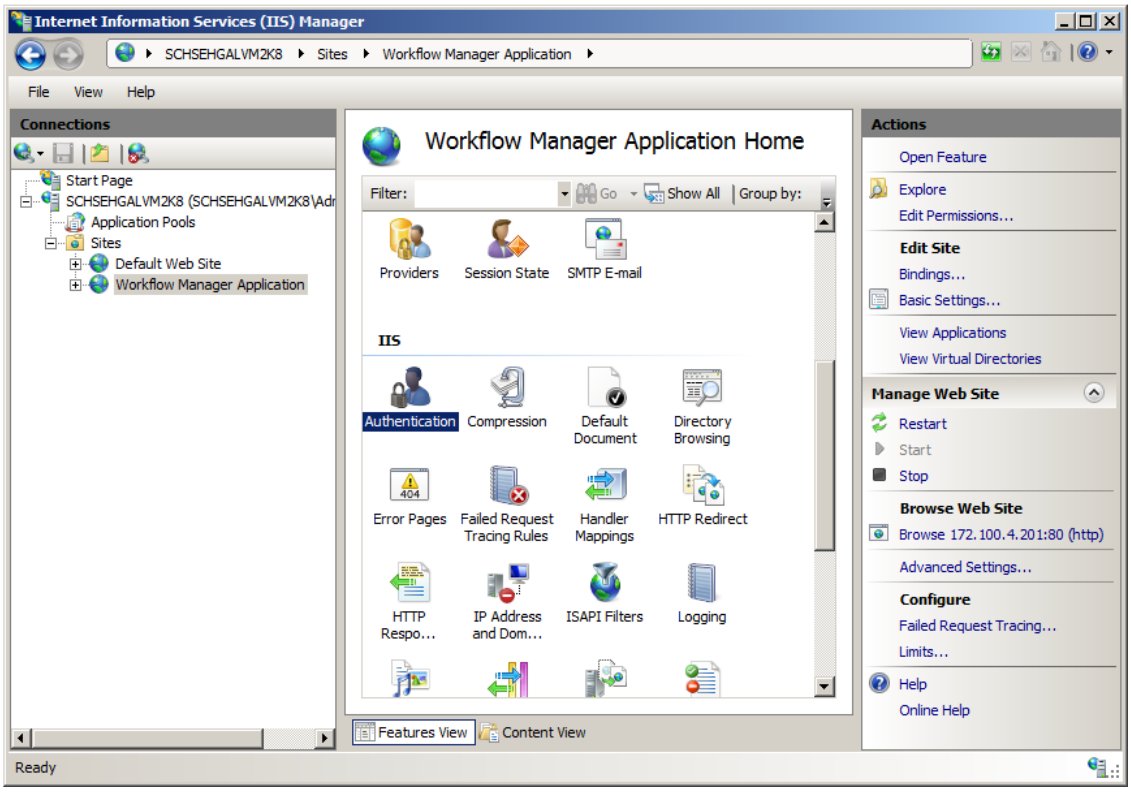
If you wish to login to Workflow Manager / AdminStudio Enterprise Server using your domain account or using the single sign-on method (as described in Login Methods), you may need to update the **Anonymous Authentication** option in IIS Manager.

Note • *The instructions in this topic explain how to set the **Anonymous Authentication** option in IIS 7. The instructions for setting this option in IIS 6 are slightly different. Refer to the Internet Information Services Manager 6 help for more information.*

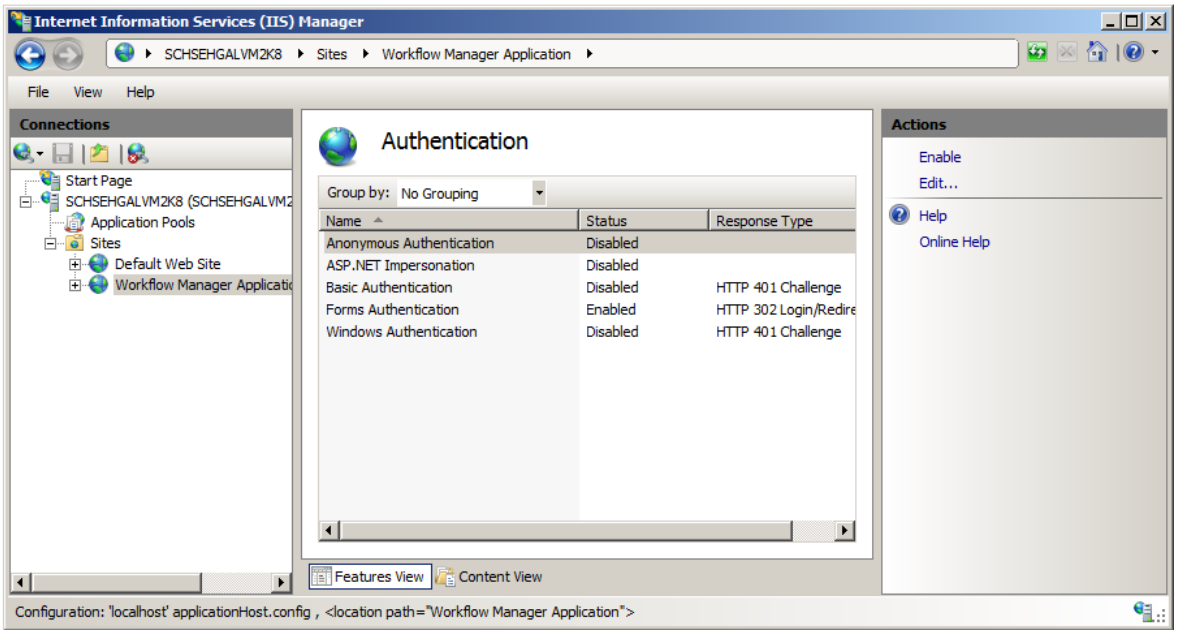
Important • *For single-sign on to work, In addition to performing these steps you must also select the Use to authenticate users? option on the **Add/Edit Directory Services Connection** page for that directory service.*

Task ***To set the Anonymous Authentication option in IIS Manager:***

1. Open the Internet Information Services (IIS) Manager. For instructions, refer to the following MSDN help topic, *Opening IIS Manager,* at http://msdn.microsoft.com/en-us/library/ms525920(v=vs.90).aspx
2. In the IIS tree view, select **Sites > Workflow Manager Application**. The **Workflow Manager Home** view opens.



3. Under **IIS**, double-click **Authentication**. The **Authentication** view opens.



4. Do one of the following:

   - **To login using your domain account**, set the **Anonymous Authentication** option to **Enabled** by selecting it and then clicking **Enable** in the **Actions** menu on the right.
   - **To login using the single sign-on method**, set the **Anonymous Authentication** option to **Disabled** by selecting it and then clicking **Disable** in the **Actions** menu on the right.

   Important • *If you disable the **Anonymous Authentication** option, make sure that the **Windows Authentication** option is set to **Enabled**.*

5. Make sure that the **Forms Authentication** option remains set to **Enabled**.

Using Account Login

You may wish to manually specify account names and passwords for all people logging in to Workflow Manager / AdminStudio Enterprise Server.

- **To set up account logins**—Follow the steps in Creating a New Account to manually create an account (**Account Name** and **Password**) for each person that you want to have access to Workflow Manager / AdminStudio Enterprise Server.
- **To login using an account login**—Once accounts have been created, people may login by entering their assigned **Account Name** and **Password** on the Workflow Manager / AdminStudio Enterprise Server login page. Workflow Manager / AdminStudio Enterprise Server will be responsible for authenticating the supplied details.

Using Domain Account Login

You may want to use your usual domain credentials to login to Workflow Manager / AdminStudio Enterprise Server.

Setting Up Workflow Manager / AdminStudio Enterprise Server to Use Domain Credentials

To set up Workflow Manager / AdminStudio Enterprise Server to use domain credentials, perform the following steps:

Task ***To set up Workflow Manager / AdminStudio Enterprise Server to use domain credentials:***

1. Set up a Windows Active Directory or Novell eDirectory directory service connection. See Creating a New Directory Service Connection for more information.
2. On your Workflow Manager / AdminStudio Enterprise Server server, open IIS Manager and enable the **Anonymous Authentication** option. This specifies that Workflow Manager / AdminStudio Enterprise Server will be responsible for authenticating login attempts. Refer to Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On for further details.
3. Import relevant accounts into Workflow Manager / AdminStudio Enterprise Server from your directory service. See Importing Directory Services Accounts and Groups for more information.

Logging in Using Your Domain Account Login

If your account has been imported from a directory service, or belongs to an imported group, you may then enter your usual domain account name and password on the Workflow Manager / AdminStudio Enterprise Server login page. Workflow Manager / AdminStudio Enterprise Server will connect to the relevant directory service, and pass through the supplied account name and password so that it can authenticate you.

Note • *When entering your account name, it is not necessary to specify the directory service domain name.*

Using Single Sign-On Login

With single sign-on, you will be automatically logged in to Workflow Manager / AdminStudio Enterprise Server using your domain credentials, as long as your domain account has been imported into the system.

Note • *Single sign-on is not supported for Novell eDirectory accounts.*

Setting Up Single Sign-On

To set up single sign-on for Workflow Manager / AdminStudio Enterprise Server, perform the following steps:

Task ***To set up single sign-on for Workflow Manager / AdminStudio Enterprise Server:***

1. Set up a Windows Active Directory directory service connection. See Creating a New Directory Service Connection for more information.
2. On your Workflow Manager / AdminStudio Enterprise Server server, open IIS Manager and disable the **Anonymous Authentication** option. This specifies that IIS will be responsible for authenticating login attempts. Refer to Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On for further details.

3. Import relevant accounts into Workflow Manager / AdminStudio Enterprise Server from your directory service. See Importing Directory Services Accounts and Groups for more information.

Modifying the Registry to Enable Using Single Sign-On

To connect to an enterprise Application Catalog using Single Sign-On, you must first modify registry data.

Task ***To modify the Registry:***

1. On the AdminStudio machine, open the Registry Editor (`Regedit`).
2. Open HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\InstallShield\AdminStudio\ASES.
3. Add a new string named Enable Automatic Authentication.
4. Set the string value to 1.

Logging in Using Single Sign-On

Once single sign-on has been set up and your account has been imported, when you next open Workflow Manager / AdminStudio Enterprise Server, the IIS web server checks to see if your domain credentials are valid in the Active Directory domain server. If they are, you will automatically be logged in to Workflow Manager / AdminStudio Enterprise Server. Workflow Manager / AdminStudio Enterprise Server does not need to directly connect to the Active Directory server.

Using Guest Account Login

Rather than creating an account for each person using Workflow Manager, you may instead choose to create a generic guest account, with restricted access to some lower-risk features.

This section describes how to setup a guest account and how to login using a guest account:

- Setting Up a Guest Account
- Logging in as a Guest

Setting Up a Guest Account

The Workflow Manager administrator can set up a guest account to permit people without login credentials to access features such as viewing a report or searching for a workflow request. By using a guest account, administrators do not have to create separate accounts for people who only need very limited functionality.

Task ***To configure a guest account:***

1. Manually create a new account in Workflow Manager to use as the guest account. See Creating a New Account for further information.
2. Assign your new account to roles with limited permissions. See Managing Roles and Permissions for more information.

Tip • *Be very careful about assigning your account to roles with access to advanced features, since these features will then be available to every person who logs in as a guest.*

3. Update the `web.config` file, located in the Workflow Manager web application `wwwroot` directory. Enter the name of the your new account in the following location of the `web.config` file:

```
<!-- Guest System Access -->
<add key="GuestAccount" value="username@companyname.com" />
```

After a guest account key is added to the `web.config` file, the **Guest Access** option will appear on the Login page.

Note • *If the account name specified in the* `GuestAccount` *key does not exist in the Workflow Manager database, Workflow Manager will display an error when an operator tries to log in as a guest. A* `GuestAccount` *key with a blank value (that is, with* `value = ""`*) will be ignored.*

Note • *Every Workflow Manager portal has its own* `web.config` *file. You may update a different* `GuestAccount` *key for each portal by updating its local* `web.config` *file. Workflow Manager portals will use the* `GuestAccount` *key configured at the portal site to log in guests. If a* `GuestAccount` *value is not configured (the key is missing, or has blank value), Workflow Manager will instead use the* `GuestAccount` *key from the parent site.*

Logging in as a Guest

Before anyone can log on to Workflow Manager anonymously, a guest account needs to be configured, as described in Setting Up a Guest Account. If a guest account is set up, and if single-sign on authentication is not configured, users can log on to your Workflow Manager site as a guest. When the Workflow Manager **Login** page opens, a user would select the **Guest Access** option to log in anonymously.

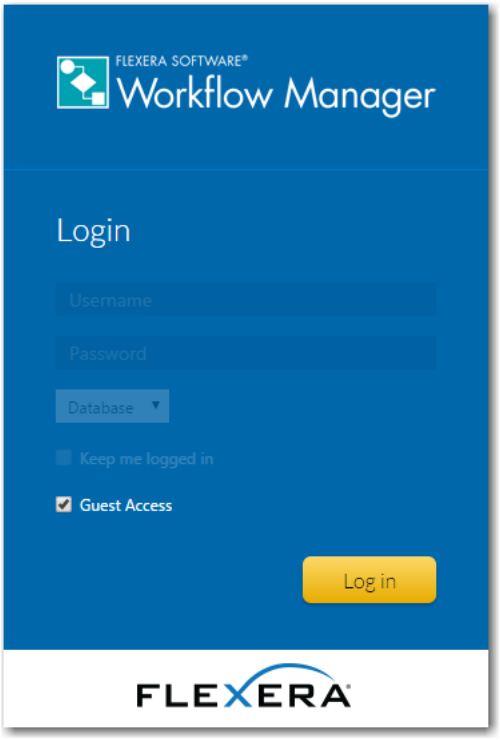


Figure 5-2: Logging in Using Guest Access Option

Important • *If single-sign on authentication is configured, the* ***Guest Access*** *option is not available.*

Tip • *The* ***Guest Access*** *option will not be available on the login page unless the* `GuestAccount` *key is present in the* `web.config` *file, with a non-empty* `value` *(that is, the key does not have* `value = ""`*). See Setting Up a Guest Account for more information.*

Accounts and Directory Services Reference

This section details the contents of the Workflow Manager / AdminStudio Enterprise Server pages that are used to manage users and directory services:

Table 5-3 • Accounts and Directory Services Reference

Page	Subpages
Account Administration Page	• Account Details Page • Directory Services Import Page
Directory Services Administration Page	• Add Directory Service Connection Page

Table 5-3 • Accounts and Directory Services Reference

Page	Subpages
Directory Services Attributes Administration Page	• Add Directory Service Attributes Page

Account Administration Page

The **Account Administration** page lists the accounts defined in Workflow Manager / AdminStudio Enterprise Server. You can view this page by clicking **Accounts and Groups** in the **Settings** menu.

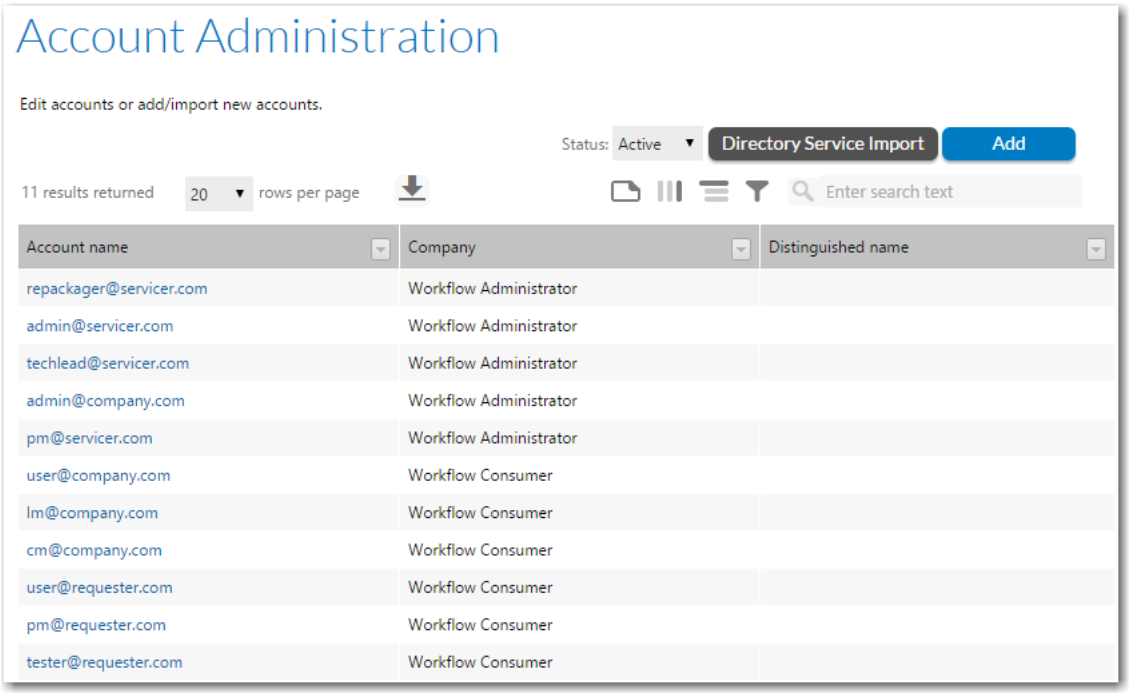


Figure 5-3: Account Administration Page

Tip • *This page will only list accounts associated with the company that your own login account belongs to. To view accounts belonging to all companies (and also to view the* `suams` *super user account), you will need to log in with a super user account, assigned the* ***System Administrator*** *role. For more information, see*

Use the **Account Administration** page to:

- Drill through to a page showing the details of a single existing account, where you may either update (see Viewing or Changing an Existing Account), disable (see Disabling an Account) or delete (see Deleting an Account) that account.
- Create a new account (see Creating a New Account).
- Import accounts or groups from a directory service (see Importing Directory Services Accounts and Groups).
- Filter the lists of visible accounts by status (see Filtering by Account Status).

Note • *For more information on the methods for logging into Workflow Manager / AdminStudio Enterprise Server and how authentication is performed, see Managing Account Logins.*

The **Account Administration** page lists the following account details, some of which are hidden by default:.

Table 5-4 • Account Administration Page Options

Option	Description
Add	Click to access the Account Details Page, where you can add a new account.
Directory Service Account/Group Import	Click to access the Directory Services Import Page, where you can import an account or group from a directory service.
Account Name	The login name that will be used to access Workflow Manager / AdminStudio Enterprise Server. Typically, account names are in the format of `accountname@companyname.com`. If the account represents a directory services group, the group's name will instead be listed in this column.
Company	The company within your organization which this account belongs to.

Account Details Page

The **Account Details** page allows you to view and update the details of an individual account, either manually created or imported from a directory service. The **Account Details** page is opened by either clicking a user name or the **Add** button on the **Account Administration** page.

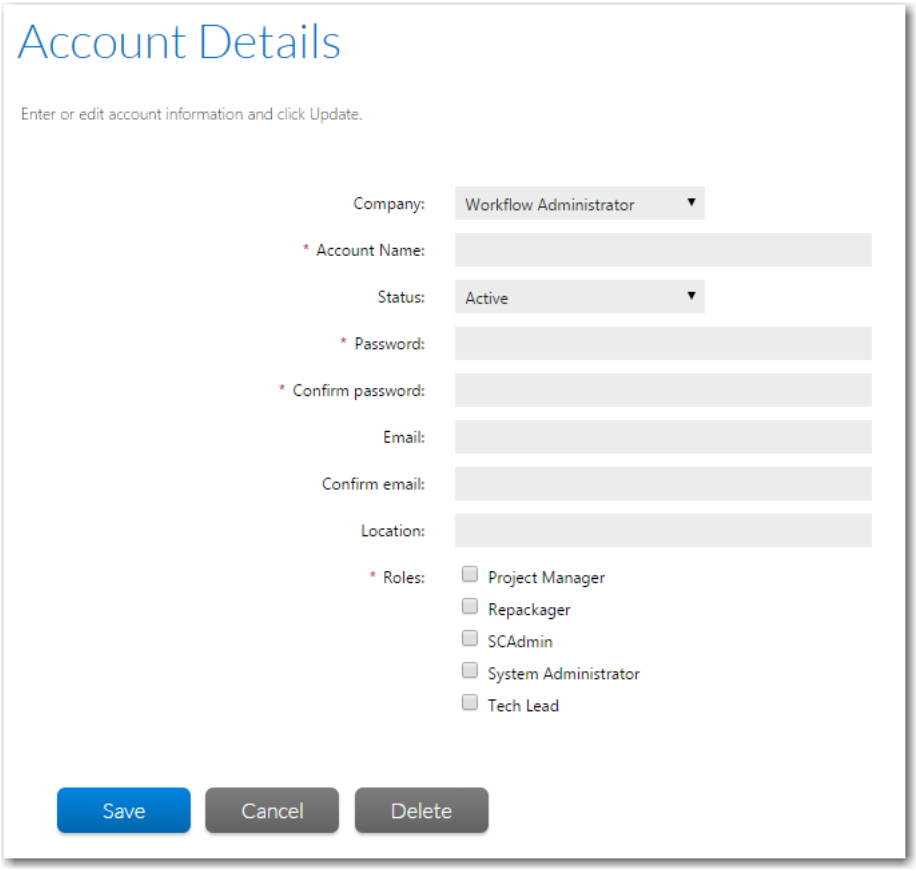


Figure 5-4: Account Details Page

Tip • *You may not delete an account which is referenced by any workflow requests or templates. Instead, you may disable such an account if it is no longer required. See Filtering by Account Status for details.*

The following fields are available on the **Account Details** page:

Table 5-5 • Account Details Page Fields

Field	Description
Company	[Workflow Manager only] Select the company that this account belongs to. If you update this field, the list of roles at the bottom of this page will dynamically update to show only those belonging to the selected company.
Account Name	Enter a unique login name to identify this account. You will use this account name to login to Workflow Manager / AdminStudio Enterprise Server. To ensure that you can easily identify which company your account belongs to, it is good practice to create an account name of the form: `username@companyname.com`. ***Note*** • *This field will be disabled for any account or group imported from a directory service.*

Table 5-5 • Account Details Page Fields

Field	Description
Status	If you wish this account to interact with Workflow Manager / AdminStudio Enterprise Server, select **Active**. The **Inactive** option disables this account. See Disabling an Account for further information.
Password	Enter a password for this account.
Confirm password	To ensure you did not misspell the password you entered in the **Password** field, re-enter it in the **Confirm password** field. You will be unable to save your changes unless you enter matching passwords in the **Password** and **Confirm Password** fields. ***Note*** *• This field will be disabled for any account or group imported from a directory service.*
Email **Confirm email**	If the person using this account is to receive notifications when his input is required to complete a workflow, enter a valid email address in this field, and reenter it in the **Confirm email** field. ***Note*** *• This field only does not appear for an account or group imported from a directory service, because this information is retrieved dynamically when needed.*
Location	Optionally, enter a geographic location that you can use to group accounts together, such as **`New York Office`** or **`Midwest Region`**.
Roles	Select the roles you wish this account to belong to. These roles will determine how the account may interact with Workflow Manager / AdminStudio Enterprise Server. Only those roles belonging to the selected company are displayed. See Role Permission Lists for more information.

Directory Services Import Page

If you have defined a directory service connection, as described in Creating a New Directory Service Connection, you can choose to import accounts or groups from that directory service into Workflow Manager / AdminStudio Enterprise Server. You import those accounts or groups using the **Directory Services Import** page, which is opened by clicking **Directory Service Account/Group Import** on the **Account Administration** page.

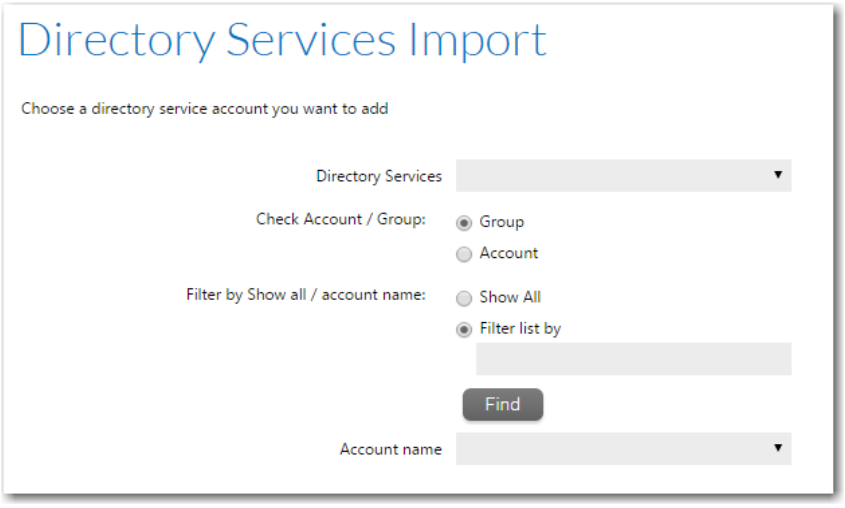


Figure 5-5: Directory Services Import Page

Note • *Workflow Manager / AdminStudio Enterprise Server supports Windows Active Directory and Novell eDirectory directory services.*

The following options are included:

Table 5-6 • Directory Services Account/Group Add View

Option	Description
Select a Directory Service	Choose the directory service containing the account or group you want to import. ***Note*** • *For information on defining a Directory Service Connection, see Creating a New Directory Service Connection.*
Select Group or User	Specify whether you are importing a **User** (a single account) or a **Group**.
Filter Directory Service List	Select one of the following: • **Show All**—Select this option to select your account from a list of all accounts and groups in the directory service. • **Filter list by**—Select this option to only return accounts and groups which match the criteria you specify, and enter your criteria in the associated text box. For example, to search for all of the users that start with the letter P, use the asterisk wildcard character (*) and enter **P*** in the **Filter list by** box. In either case, click **Find** to return your chosen list of accounts and groups.

Table 5-6 • Directory Services Account/Group Add View (cont.)

Option	Description
Pick group/user in the list below	Select the group or account to import into Workflow Manager / AdminStudio Enterprise Server. The **Account Details** page opens, providing a read-only view of the imported account or group's account name and password. ***Note*** *• If you import an account that is a member of a previously-imported group, that account inherits its group's roles. You can then assign additional roles to that account.*

The selected user or group is then opened in the Account Details Page.

Directory Services Administration Page

Workflow Manager / AdminStudio Enterprise Server can be integrated with Windows Active Directory and Novell eDirectory. This enables you to set up automatic login with Workflow Manager / AdminStudio Enterprise Server based upon directory service authentication.

Directory services connections are used to import users and groups into Workflow Manager / AdminStudio Enterprise Server, and to authenticate Active Directory or Novell eDirectory users logging into Workflow Manager / AdminStudio Enterprise Server. If you import a group, all members of that group can then login to Workflow Manager / AdminStudio Enterprise Server without requiring you to import them individually. Workflow Manager / AdminStudio Enterprise Server can then retrieve attributes, such as email address or telephone number, from the directory service dynamically.

You can choose to have a directory service listed in the **Domain** list on the Workflow Manager / AdminStudio Enterprise Server login page. This enables users in this directory service to login using their enterprise network credentials.

You may also add data elements to your workflow templates which, for accounts imported from directory services, will be directly populated from directory service attributes (such as account name, email address or location).

Directory Services tasks can be managed starting from the **Directory Services Administration** page, which you can open by clicking **Directory Services** on the **Settings** menu.

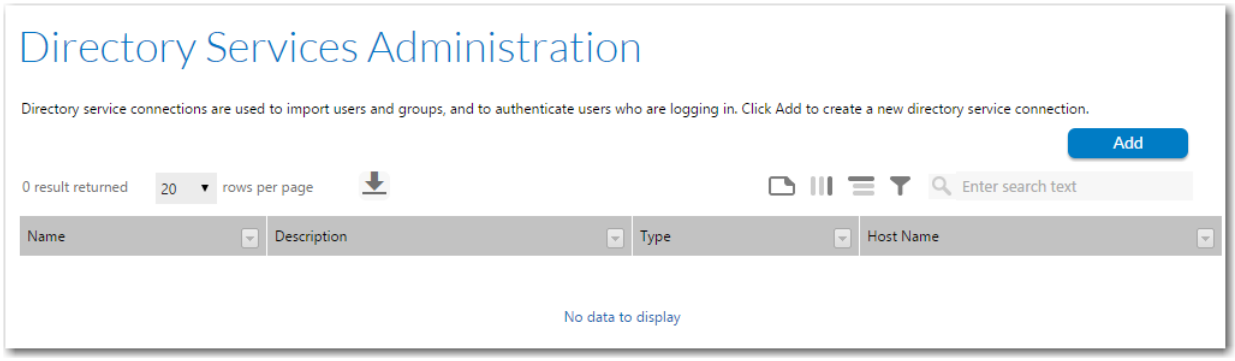


Figure 5-6: Directory Services Administration Page

The **Directory Services Administration** page lists the following information about each connection:

Table 5-7 • Directory Services Administration Page Options

Option	Description
Name	A unique identifier for the directory service connection. You might like to enter the domain name as the description, since that should be unique for your organization.
Description	A more detailed description of the directory service.
Type	Identifies this directory service as either Windows Active Directory or Novell eDirectory.
Host Name	The name or IP address of the server on which this directory service is running.

Note • *For more information about Directory Services and Lightweight Directory Access Protocol (LDAP), see Lightweight Directory Access Protocol (LDAP) Overview on the Microsoft TechNet website.*

Add Directory Service Connection Page

The **Add Directory Service Connection** page allows you to view and update the settings required to connect to a directory service. You can also use this page to remove a connection which is no longer required.

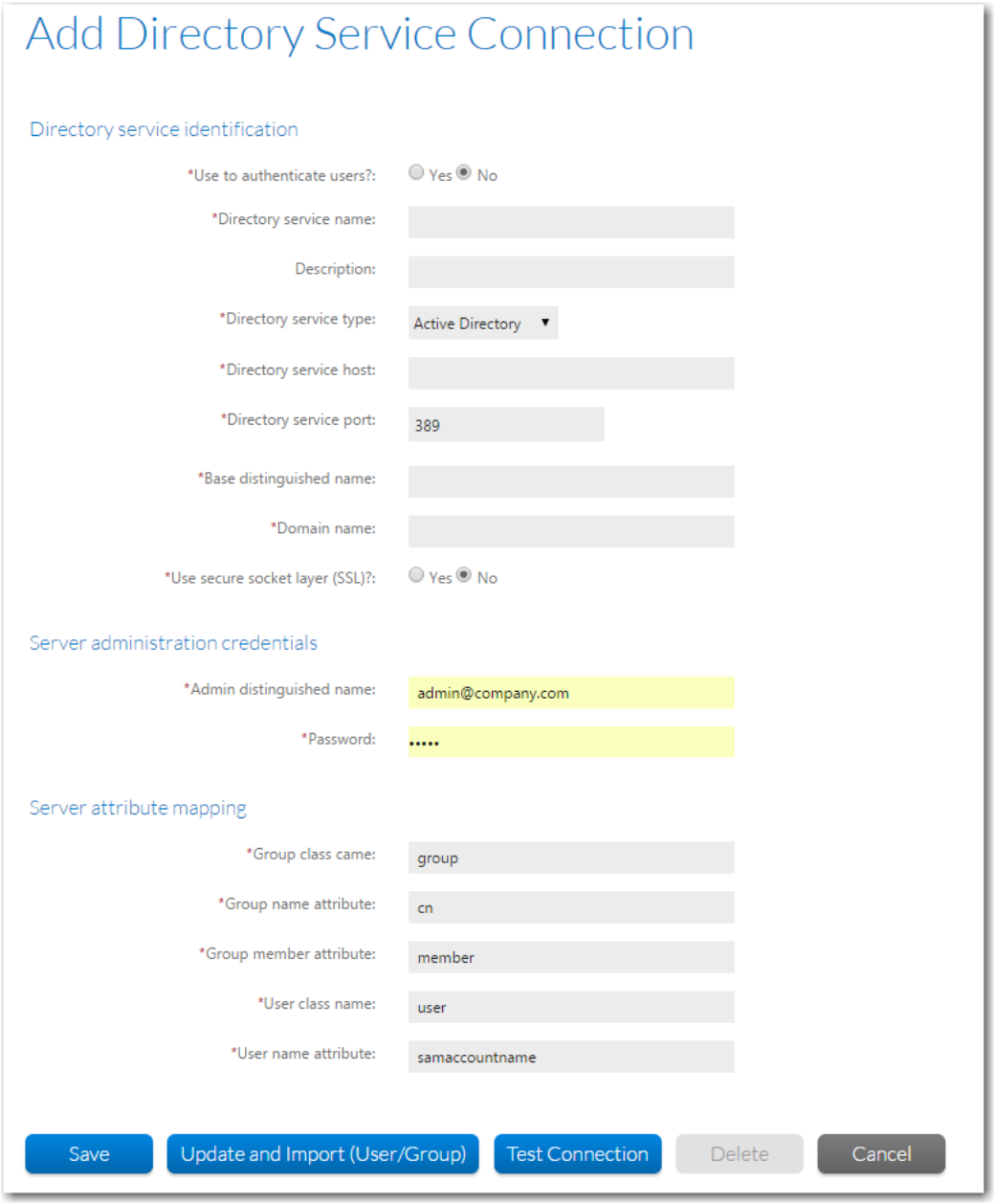


Figure 5-7: Edit Directory Service Connection Page

The following fields are available on the **Add/Edit Directory Service Connection** page:

Table 5-8 • Add/Edit Directory Service Connection Page Fields

Option	Description
Use to authenticate users?	Select **Yes** to include this directory service in the **Domain** list on the Workflow Manager / AdminStudio Enterprise Server login page. This enables people to login to Workflow Manager / AdminStudio Enterprise Server using their enterprise network credentials. You can create multiple directory service connections, but only those connections that have this field set to **Yes** will be listed in the **Domain** list. ***Note*** • *This option must be selected in order to enable single sign-on, which means that users are automatically logged on to Workflow Manager using their enterprise network credentials, bypassing the Login screen. However, to enable single sign-on, you must also perform the steps listed in Setting the Anonymous Authentication Option in IIS Manager to Enable Single Sign-On.* ***Note*** • *In an enterprise, there is usually only one directory service that is responsible for authenticating accounts. However, you can create additional directory service connections to import accounts and groups.*
Directory service name	Enter a name to identify this directory service in the domain list on the Workflow Manager / AdminStudio Enterprise Server login page. You may wish to use the domain name associated with the directory service.
Description	Some further information about this directory service. You may wish to identify which parts of the organization use this directory service for authentication, for example.
Directory service type	Select the type of directory service you are integrating with. The following options are available. • Active Directory • Novell eDirectory
Directory service host	The name or IP address of the server on which this directory service is running.
Directory service port	Enter the port number of the server on which the directory service is running, to which Workflow Manager / AdminStudio Enterprise Server should connect in order to send LDAP queries. The default port number is **389**.
Base distinguished name	Enter the base distinguished name (DN) to identify the root node of this directory service. For example, for MyCompany, the base DN could be: `dc="MyCompany", dc="com"`
Domain name	Enter the domain name of this directory service.

Table 5-8 • Add/Edit Directory Service Connection Page Fields (cont.)

Option	Description
Use secure socket layer (SSL)?	Select **Yes** if this directory service is configured to use Secure Socket Layer (SSL).
Connect anonymously?	Select **No** if you do not want to permit anonymous connections. If you select **Yes** to permit anonymous connections, Workflow Manager / AdminStudio Enterprise Server may not be able to authenticate directory service users and may not be able to add directory service users/groups into Workflow Manager / AdminStudio Enterprise Server.
Admin distinguished name	Enter the distinguished name of an operator who has permission to retrieve account/group information and authenticate an account against this directory service, in the **`domainName\userName`** format.
Password	The password associated with the credentials specified in **Admin Distinguished Name**.
Group class name	Enter the object class name used to identify groups in this directory service. Default values are: • For Active Directory: **`group`** • For Novell eDirectory: **`groupofnames`**
Group name attribute	Enter an attribute used by this directory service to name groups. The default value for both Active Directory and Novell eDirectory is **`cn`**.
Group member attribute	Enter an attribute used by this directory service to define member groups. Default values are: • For Active Directory: **`member`** • For Novell eDirectory: **`uniquemember`**
User class name	Enter the object class name used by this directory service for user accounts. Default values are: • For Active Directory: **`user`** • For Novell eDirectory: **`inetorgperson`**
User name attribute	Enter the attribute used by this directory service to identify user accounts. Default values are: • For Active Directory: **`samaccountname`** • For Novell eDirectory: **`uid`**
Save	Click to save your entries and return to the Directory Services Administration Page.
Update and import (User/Group)	Click to save your entries and open the Directory Services Import Page.

Table 5-8 • Add/Edit Directory Service Connection Page Fields (cont.)

Option	Description
Test Connection	Click to test to see if the settings that you entered can be used to successfully connect to this directory service.

Directory Services Attributes Administration Page

Edition • *This feature applies to Workflow Manager only.*

The **Directory Services Attributes Administration** page lists all attributes which you have chosen to make available in workflow templates. You can view this page by clicking **Manage Directory Services Attributes** on the **Directory Services** page.

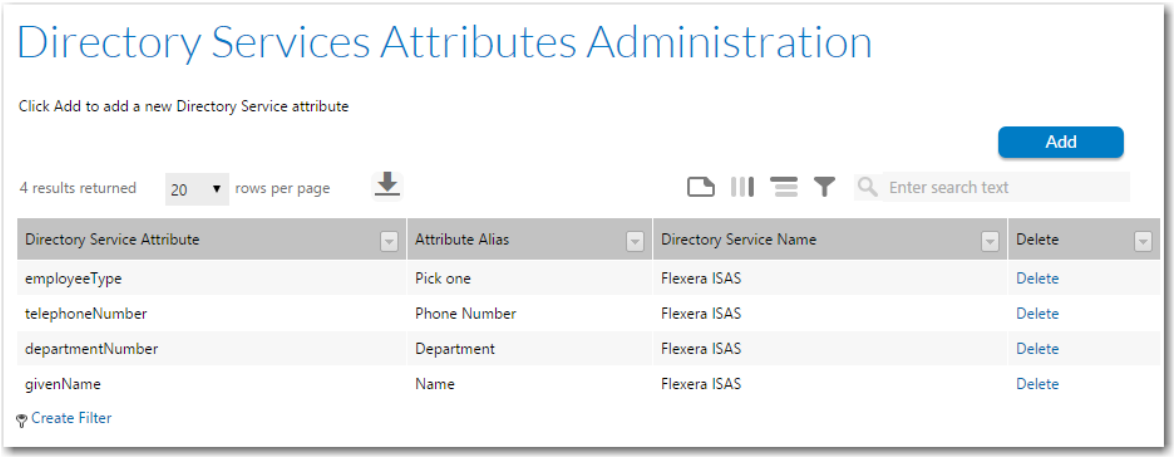


Figure 5-8: Directory Services Attributes Administration Page

Directory service attributes can be used when defining data elements. When a data element is defined as a directory service attribute, when this data element appears during a data entry step in a workflow, information about the logged in user will be pulled from the directory service to populate those fields, such as: department, location, employee number, etc.

The **Directory Services Attributes Administration** page lists the following information about each attribute:

Table 5-9 • Directory Services Attributes Administration Page Options

Option	Description
Directory Service Attribute	The name of an attribute available from one of your defined directory services. Attributes are used in directory services protocol to access information directories, such as `employeeNumber` or `documentAuthor`.
Attribute Alias	A more user-friendly name for a directory service attribute. For example, you may choose to give a directory service called homePhone the friendly name **`Home telephone number`**.

Table 5-9 • Directory Services Attributes Administration Page Options (cont.)

Option	Description
Directory Service Name	The directory service this attribute belongs to.

Add Directory Service Attributes Page

Edition *• This feature applies to Workflow Manager only.*

When you click **Add** on the **Directory Services Attributes Administration** page to add a new directory service attribute into the database, the **Add Directory Service Attributes** page opens.

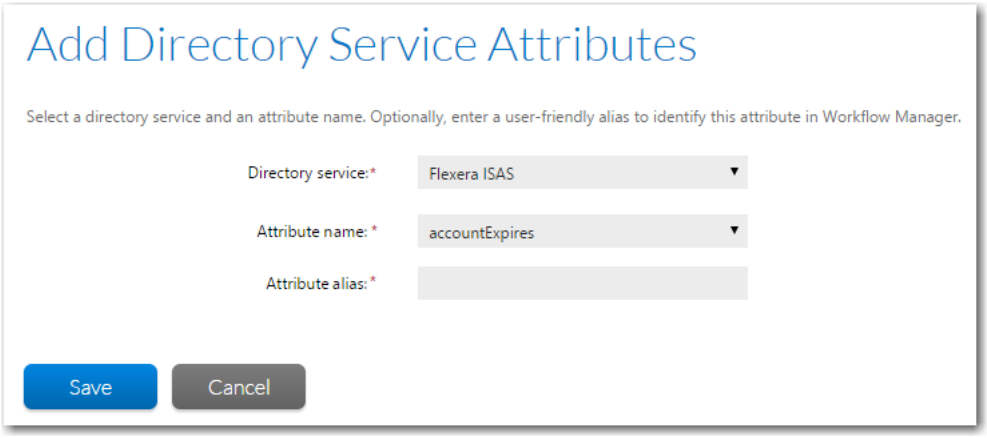


Figure 5-9: Add Directory Service Attributes Page

The **Add Directory Service Attributes** page includes the following options:

Table 5-10 • Add Directory Service Attributes Page Options

Option	Description
Directory service	Select the directory service whose attribute you want to make available to workflow templates from the **Directory Servic**e list. Refer to Creating a New Directory Service Connection for more information about setting up connections to your directory services.
Attribute name	Lists all of the directory service attributes from the chosen directory service. Select the one you want to add to the Workflow Manager database. An example of a directory service attribute might be `employeeNumber` or `documentAuthor`.
Attribute alias	Enter a more user-friendly identifier for the attribute in the **Attribute Name** field. For example, you may want to identify the `documentAuthor` attribute as **Author** in Workflow Manager.

Table 5-10 • Add Directory Service Attributes Page Options (cont.)

Option	Description
Save	Click to register the defined directory service attribute. ***Tip*** *• If Workflow Manager is unable to connect to the server (and no attributes are retrieved), the* ***Save*** *button is disabled.*

When a data element is defined as a directory service attribute during Workflow Manager template creation, when this data element appears in the workflow, information will be pulled from the directory service to populate those fields, such as:

Figure 5-10: Example of Fields Populated With Directory Services Attributes

However, if the user is not connected using Directory Service authentication, then the fields will be left blank and will be enabled.

6

Managing Roles and Permissions

All of the Workflow Manager / AdminStudio Enterprise Server permissions are related to roles. The people using and administering Workflow Manager / AdminStudio Enterprise Server are granted access based on the roles they belong to.

Information on using roles and permissions is presented in the following sections:

- AdminStudio and Workflow Manager Roles and Permissions
- Role Management
- Roles Reference

AdminStudio and Workflow Manager Roles and Permissions

Each role consists of a set of permissions to allow access to different features or areas of AdminStudio and Workflow Manager. Every person who needs to work with these applications or administer the system is then assigned to one or more roles, and the set of features he can access is a combination of the permissions supplied by all of his roles.

This section includes the following topics:

- Role Permission Lists
- System Roles

Role Permission Lists

Permissions to perform all AdminStudio and Workflow Manager functions are assigned using roles. A user has permission to perform only those tasks that are explicitly selected in the role(s) that the user is assigned to.

This section describes all of the AdminStudio and Workflow Manager permissions:

- Administration and Reports Permissions
- Workflow Manager Permissions

- AdminStudio Client Tools Permissions

Administration and Reports Permissions

AdminStudio Enterprise Server and Workflow Manager have the following permissions:

Table 6-1 • AdminStudio Enterprise Server and Workflow Manager Permissions

Feature	Right	This right grants permission to...
Approval	**View**	See the **Approval Administration** page.
	Edit	Modify an existing approval template, including adding or removing users.
	Add	Add/delete an approval template. ***Important*** • *After you create a template and add a new role, the check boxes for the new role are enabled by default in the Workflow Phase Permissions.*
Directory Services	**View**	See the **Administration/Directory Services** tab and view the **Directory Services** page, the **Directory Services List** page, and the **Directory Services Attributes Administration** page.
	Edit	Modify an existing directory service.
	Add	Create a new directory service.
	Delete	Remove an existing directory service.
Email Templates	**Edit**	Modify email templates.
External Data Sources	**Edit**	View and modify the settings on the **External Data Sources** tab.
Global Email Administration	**View**	See the **External Email Address Administration** page.
	Edit	Modify the settings on the **External Email Address Administration** page.
People	**View**	See the **Account Administration** and **Account Details** pages.
	Edit	Modify an existing account.
	Add	Create a new account and import accounts from a directory service.
	Delete	Remove an existing account.

Table 6-1 • AdminStudio Enterprise Server and Workflow Manager Permissions

Feature	Right	This right grants permission to...
Roles	**View**	See the **Role Administration** and **Role Details** pages.
	Edit	Modify an existing role.
	Copy	Duplicate an existing role.
	Add	Create an existing role. ***Note*** • *You must have this permission in order to be able to upgrade an existing Application Catalog.*
	Email Notify Enabled	Enable email notification.
System Settings	**View**	View system settings.
	Edit	Edit system settings.
Reports / All Reports	**Edit**	Modify an existing custom report.
	Add	Create a new custom report by making a selection under **Custom Reports** on the **Reports** menu.
	Delete	Remove an existing custom report.
Reports / Package Reports	**View**	See the **Search Packages** page and view Package Reports.

Workflow Manager Permissions

The Workflow Manager category covers general access to Workflow Manager, allowing you to specify precisely which areas of the product people may use.

Table 6-2 • Workflow Manager Permissions

Category	Right	This right grants permission to...
Workflow Request	**View**	See the **Properties** and **Issues** tabs on the **Workflow Request** page for an existing workflow request.
	Edit	Modify an existing workflow request.
	Copy	Duplicate an existing workflow request.
	Add	Create a new workflow request.
	Delete	Remove an existing workflow request.
	Monitor Workflow Progress	See the **Progress** tab of the **Workflow Request** page, and complete workflow steps and workflow phases.
	View Related Workflows	See the **Related Workflows** tab of the **Workflow Request** page.
	Add Related Workflows	Link one workflow request to another on the **Related Workflows** tab
	Deleted Related Workflow Links	Unlink related workflows.
	Edit Workflow Due Period	Modify the **Workflow due period** field on the **Properties** tab of the **Workflow Request** page.
	View Properties	See the **Properties** tab of the **Workflow Request** page.
	View Uploaded Files	See the **Uploaded Files** tab of the **Workflow Request** page.
	View Downloadable Files	See the **Downloadable Files** tab of the **Workflow Request** page.
Workflow Request (Continued)	**View Documents**	See the **Documents** tab of the **Workflow Request** page.
	Upload Documents	Upload documents on the **Documents** tab of the **Workflow Request** page.
	Delete Documents	Delete documents on the **Documents** tab of the **Workflow Request** page.
	Download Documents	Download documents on the **Documents** tab of the **Workflow Request** page.

Table 6-2 • Workflow Manager Permissions

Category	Right	This right grants permission to...
Calendar Settings	**View**	See the **Calendar Settings Administration** page.
	Edit	Modify calendar settings.
Consumer Company	**View**	See the list of existing consumer companies, and view their details.
	Edit	Update details of an existing consumer company.
	Add	Create a new consumer company.
	Delete	Remove an existing consumer company.
Issues	**View**	See the **Issues** tab of the **Workflow Requests** page, and drill through to see details of an individual issue.
	Respond	Respond to an existing issue.
	Add	Create a new **E-mail** or **Knowledge Base** issue.
	Close	Close an existing **E-mail** or **Knowledge Base** issue.
	Add/Close Critical Issue	Create and close **Critical** issues.
My Notifications	**View**	See the **My Notifications** pages: **My Default Project Notifications** and **My Workflow Notifications**.
	Edit	Modify settings on the **My Default Project Notifications** and **My Workflow Notifications** pages.
Workflow and Template Permissions	**View**	See the permission and email settings on the **Template Details** and **Project Details** pages.
	Edit	Modify the permission and email settings on the **Template Details** and **Project Details** pages.
Projects	**View**	See the **Project Administration** page, and see (but not update) project details on the **Project Details** page.
	Edit	Update details of an existing project.
	Add	Create a new project.
	Delete	Remove an existing project.

Table 6-2 • Workflow Manager Permissions

Category	Right	This right grants permission to...
Search	**Simple Search**	Enter keywords in search box on the **Home** page to search for a workflow request by name.
	Advanced Search	Search for workflow requests by specifying multiple criteria on the **Filter Your Search** menu.
Task Approval	**Approve on Behalf**	Enables user to approve a workflow step with a **Step Type** of **Approval Task** on behalf of any and all users.
Templates	**View**	See the **Template Administration** page, view template details on the **Template Details** page, and add an external data source by clicking **External Data Sources** on the **Settings** menu.
	Copy	Duplicate an existing template.
	Add	Modify an existing template and create a new one.
Terminology	**View**	See and modify system terminology.
Work Assignment	**View**	See a list of existing work assignments on the **View Assignments by Account** page.
	Assign	Assign work on the **Assign Work** page.
Workflow Administrator Company	**View**	See the list of existing administrator companies, and view and update administrator company details.
	Add	Create a new administrator company.
Workflow Status Management	**View**	See the **Workflow Status Administration** page and see (but not modify) workflow status details on the **Edit Workflow Status** page.
	Edit	Update workflow status details on the **Edit Workflow Status** page.
	Add	Create a new workflow status.
	Delete	Remove an existing workflow status.
Time Off Admin	**Admin**	Gives user permission to see time off records for all users who are members of the administrator company or one of its associated consumer companies. Without this role permission, a user can only see their own time off records.

AdminStudio Client Tools Permissions

The AdminStudio client tools have the following permissions:

Table 6-3 • AdminStudio Client Tools Permissions

Category	Right	This right grants permission to...
AdminStudio Client Interface Process Assistants Tab	**Edit**	View and edit the projects on the **Process Assistants** tab that are assigned to him.
	Create	Create new projects and assign them to users. Users with the **Create** permission see a list of all users and their associated Projects on the **Process Assistants** tab. Users with only the **Edit** permission cannot create new projects and can only view and edit projects that are assigned to him.
	Delete	Delete a project from the **Process Assistants** tab.
AdminStudio Client Interface Process Template Editor	**View**	View existing workflows using the **Process Template Editor**.
	Edit	Modify existing workflows using the **Process Template Editor**.
	Create	Create a new workflow using the **Process Template Editor**.
	Delete	Delete a workflow from the **Process Template Editor**.
AdminStudio Client Interface AdminStudio Tools button	**Add**	Add a new tool to the **AdminStudio Tools** button.
	Edit	Modify the properties of an existing tool on the **AdminStudio Tools** button.
	Delete	Delete an existing tool on the **AdminStudio Tools** button.
AdminStudio Client Interface General	**Modify Tools Options Dialog**	Set options on the **Locations**, **Updates** and **Quality** tabs of the AdminStudio **Options** dialog box. Users without this permission can view the **Options** dialog box but cannot make any changes.
	Change Default Application Catalog	Permits user to edit the **Make this the default shared Application Catalog** option on the **Connect Application Catalog** dialog box
Application Catalog / Conflict Solver	**Select Tests to Execute**	Permits user to edit the selections on the **Select Tests to Execute** dialog box.

Table 6-3 • AdminStudio Client Tools Permissions

Category	Right	This right grants permission to...
Application Catalog / Conflict Solver Conflicts	**Run Analysis**	Perform conflict analysis on a package.
	Resolve	Resolve any automatically resolvable conflicts found during conflict analysis on a package.
	Modify Rules	Open the **Rules Viewer** and create and edit new rules.
	Modify Data	Create new groups, rename existing groups, and modify group properties. ● Permits user to cut and paste a group to a new location. ● Permits user to copy/cut and paste a package into a new group. ● Permits user to modify options on the **Resolution Options** dialog box. ● Permits user to edit a package **Description** on the **Products View.**
Application Catalog / Conflict Solver Package	**Delete**	Delete a package from the Application Catalog.
	Import	Import a package into the Application Catalog.
	Modify Extended Attributes	Modify a package's metadata on the Extended Attribute view.
	Delete History	Delete a package's history log (which contains a record of any operation that materially changes a software package or the data associated with it).

Table 6-3 • AdminStudio Client Tools Permissions

Category	Right	This right grants permission to...
Application Catalog	**Run Merge Wizard**	No longer used.
Other	**Run Validation**	Validate a Windows Installer package against custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards.
	Scan for Dependencies	Generate a list of all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog.
	Run Best Practices	Use Analyze to evaluate source packages to see if they meet Windows Installer best practices rules.
	Modify Tools Options Dialog	Open the Application Catalog **Options** dialog box and set options.
	Change Application Catalog	Connect to a different Application Catalogs by selecting **Connect** on the **Application Catalog** menu. Without this permission, this selection is disabled.
OS Security Patch Wizard	**Import Patch**	Use **Import Wizard** to import Windows operating system patches into the Application Catalog.
	Run Analysis	Use the **Patch Impact Analysis Wizard** to analyze the impact of installing an OS Security patch on user machines.
Software Repository	**Overwrite**	Import a duplicate package into the Software Repository, overwriting the existing version.

System Roles

AdminStudio is installed with default **System Roles** which cannot be modified. These roles were created based upon the typical needs of people accessing the product, and have only the permissions that these people would require to perform their day-to-day tasks. You can assign these system roles to people within your enterprise, or can copy and then modify these roles to customize them for your organization.

Any new roles that you create, either manually or by copying and modifying system roles, are considered *user roles*. These can be freely modified.

Copied system roles or new roles that you create have a **Role Type** of **Account** (user roles), while default roles created during installation have a **Role Type** of **System** (system roles). A role's **Role Type** is listed on the **Role Administration** page and cannot be changed.

Information about system roles is organized in the following sections:

- Super User Role: AMSSuper

- Default System Roles
- Default System Accounts

Super User Role: AMSSuper

The default **AMSSuper** role has full rights to administer and use AdminStudio Enterprise Server and AdminStudio. During installation, the following super user account is created and assigned the **AMSSuper** role:

- **User Name**: suams
- **Password**: suams

Important • *Upon first login using the* suams *account, it is important that you change the password.*

This role is unique in that it allows the user to manage roles and accounts from all companies. All other roles are associated with a specific company within your organization, and so grant access only to roles, accounts and other entities belonging to that company.

An operator assigned the **AMSSuper** role can create administrator companies. All other tasks should be performed by a person belonging to a workflow administrator role.

Note • *The **AMSSuper** role is not listed on the **Role Administration** page unless you are logged on using the* suams *account.*

Default System Roles

When Workflow Manager / AdminStudio Enterprise Server is installed, the following system roles are created:

Table 6-4 • Default System Roles

Role Name	Company Name	Description
Configuration Manager	**Workflow Consumer**	Use for people managing the software configuration of computers in the enterprise, whose responsibilities may include deployment of software.
License Manager	**Workflow Consumer**	Use for people managing the license compliance of software throughout the enterprise.
Consumer Project Manager	**Workflow Consumer**	Use for people managing the requests submitted by workflow consumers.
Workflow Project Manager	**Workflow Administrator**	Use for people managing the completion of the submitted requests.
Repackager	**Workflow Administrator**	Use for people performing software application repackaging in your organization.

Table 6-4 • Default System Roles

Role Name	Company Name	Description
System Administrator	**Workflow Administrator**	Use for people who configure Workflow Manager or take action on workflow requests submitted by workflow consumers.
Tech Lead	**Workflow Administrator**	Use for people performing technical infrastructure tasks in your organization.
UA Tester	**Workflow Consumer**	Use for people performing user acceptance testing in your organization.
User	**Workflow Consumer**	Use for general workflow consumers; employees in your organization who will submit workflow requests related, for example, to installation of new software.

Note • *If you did not purchase Workflow Manager, the roles associated with the workflow consumer company are not listed.*

You can view the permissions of each of these roles by selecting the role on the **Role Administration** page, and then expanding the **Role Permissions** list.

Default System Accounts

When Workflow Manager / AdminStudio Enterprise Server is installed, an account is created for each of the system roles.

To see what functionality one of these default system accounts has, select the associated role on the **Role Administration** page, and then expand the **Role Permissions** list.

Note • *By default, the password for each of these default system accounts is the same as the text prior to the @ sign (such as* `lm` *for* `lm@company.com`*).*

Role Management

This section describes how to review and manage the roles created for your organization.

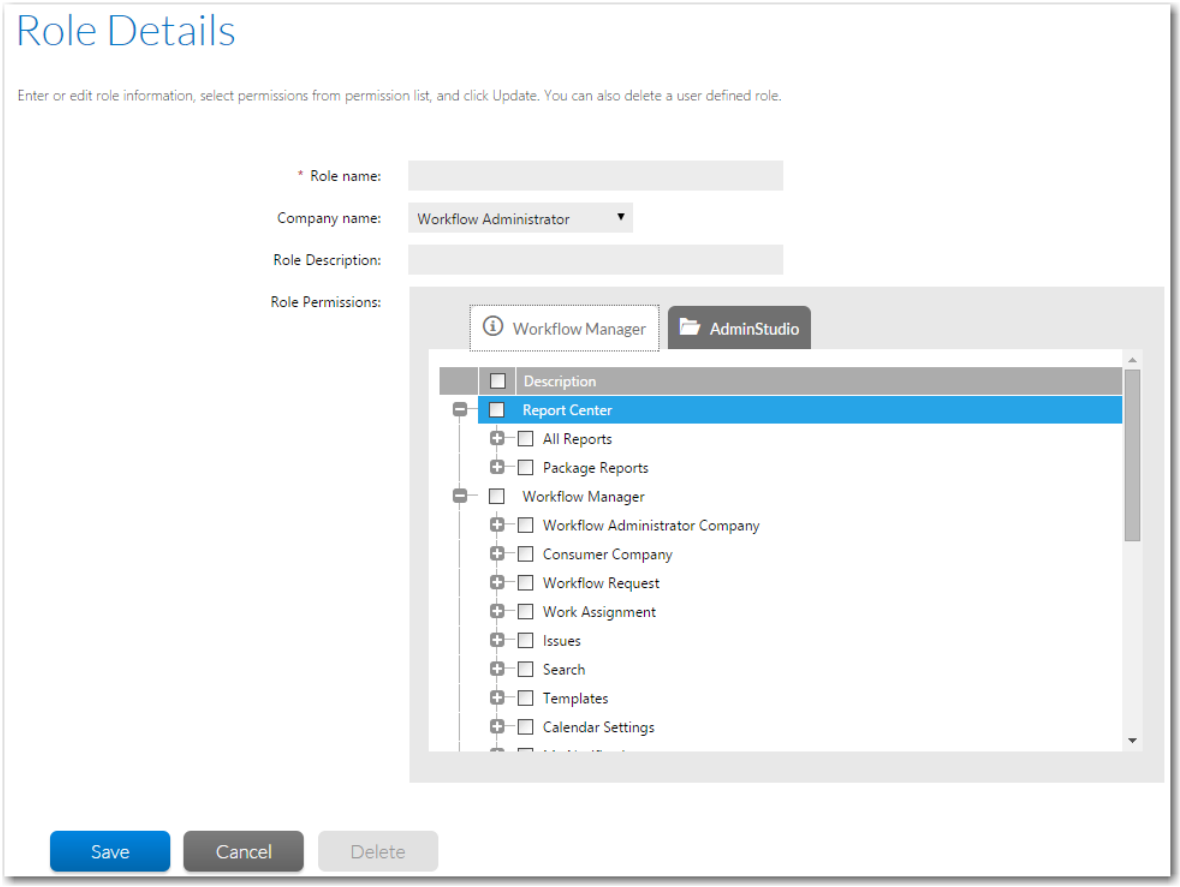
- Creating a New Role
- Viewing or Changing an Existing Role
- Copying an Existing Role
- Deleting a Role

Creating a New Role

If the default system roles aren't flexible enough to cover all security requirements in your enterprise, you may need to create new roles.

Task ***To create a new role:***

1. Click **Roles** on the **Settings** menu. The **Role Administration** page opens.
2. Click the **Add** button. The **Role Details** page opens.



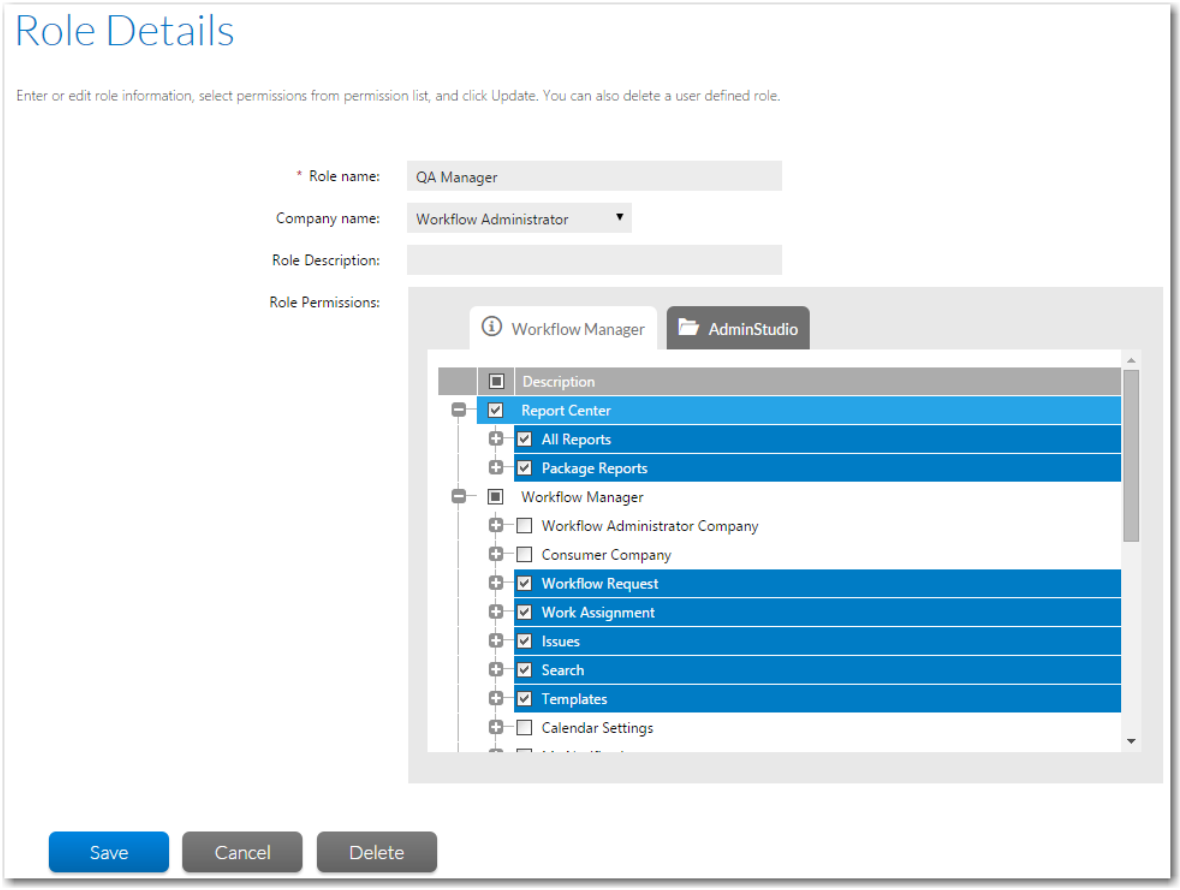
3. Enter details to identify the role, and assign appropriate permissions. For more information, see Role Details Page and Role Permission Lists.
4. Click the **Save** button. The **Role Details** page closes, and the new role now appears in the list on the **Role Administration** page.

Viewing or Changing an Existing Role

You may view the details of any role listed on the **Role Administration** page, but cannot update any of the default system roles.

Task ***To view or update an existing role:***

1. Click **Roles** on the **Settings** menu. The **Role Administration** page opens.
2. Locate the role that you want to work with.
3. Click role you want to edit. The **Role Details** page for that role opens.



4. View or update the role as required. For more information, see Role Details Page and Role Permission Lists.
5. Click **Save** to save your changes and return to the **Role Administration** page. To exit without making any changes, click **Cancel**.

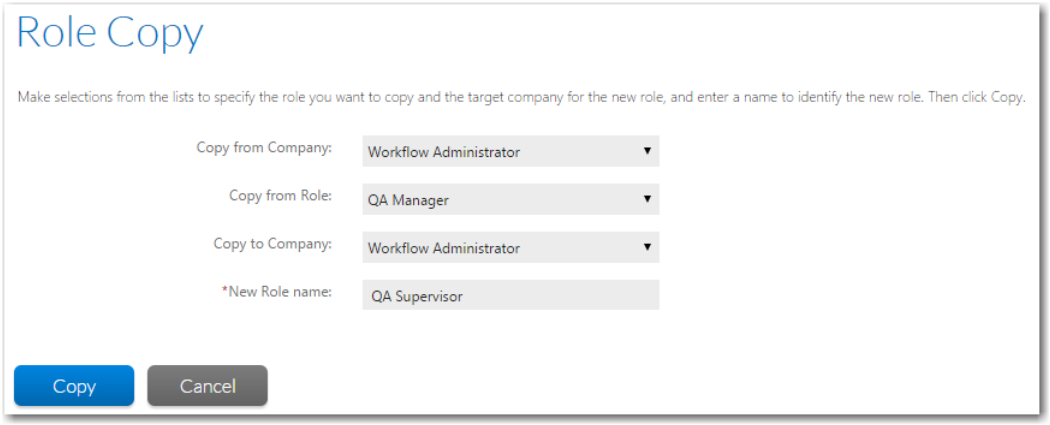
Tip • *The **Save** button will be disabled if you are viewing a system role.*

Copying an Existing Role

You can make a copy of any existing role, and then customize it for your organization. This is particularly useful for tweaking system roles, since you cannot modify them directly. Because people may only be assigned to roles created for their specific company, you may also want to copy roles if they are common to more than one of the companies defined in your organization.

Task ***To copy an existing role:***

1. Click **Roles** on the **Settings** menu. The **Role Administration** page opens.
2. Click the **Copy** button. The **Role Copy** page opens.



3. Select the name of the company whose roles you want to copy from the **Copy from Company** list.
4. Select the role you want to copy from the **Copy from Role** list.
5. Select the name of the company you are creating the new role for from the **Copy to Company** list. You may select the same company that your original role belongs to.
6. Enter a name in the **New Role Name** field to uniquely identify this role.

Note • *You are not permitted to have two roles in the same company with the same name. You can, however, use the same role name in more than one company.*

7. Click the **Copy** button. The **Role Copy** page closes, and the new role appears in the list on the **Role Administration** page.
8. If you want to edit the new role's **Role Description** or modify its permissions, perform the steps listed in Viewing or Changing an Existing Role.

Deleting a Role

If a role is no longer relevant to your organization, you can choose to delete it. You cannot remove any of the default system roles.

Task ***To delete an existing role:***

1. Click **Roles** on the **Settings** menu. The **Role Administration** page opens.
2. Click the role that you want to delete. The **Role Details** page for that role opens.
3. Click **Delete**. You are prompted to confirm the deletion.

Tip • *The **Delete** button will be disabled if you are viewing a system role.*

4. Click **OK**. The **Role Details** page closes and the role you deleted is no longer listed on the **Role Administration** page.

Roles Reference

Reference information for roles is presented in the following sections:

- Role Administration Page
- Role Copy Page
- Role Details Page

Role Administration Page

The **Role Administration** page lists roles defined in the system. You can view this page by clicking **Roles** in on the **Settings** menu.

Use this page to:

- Drill through to a page showing the details of and permissions associated with a single existing role, where you may either update (see Viewing or Changing an Existing Role) or delete (see Deleting a Role) that role.
- Create a new role (see Creating a New Role).
- Copy an existing role (see Copying an Existing Role).

Tip • *This page will only list roles associated with your account's company. To view all roles, you will need to log in with the super user account, assigned the **AMSSuper** role.*

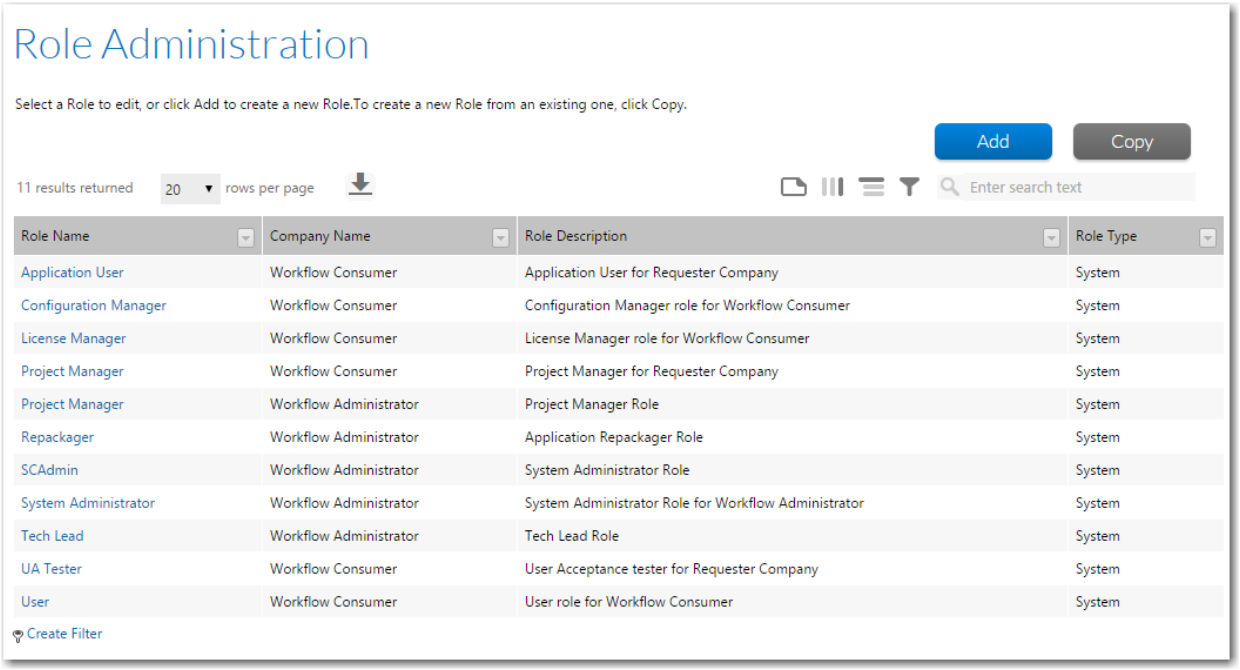


Figure 6-1: Role Administration Page

The **Role Administration** page lists the following role details, some of which are hidden by default:

Table 6-5 • Role Administration Page

Option	Description
Role Name	A brief identifier for the role.
Company Name	Only people belonging to this company can be assigned to this role.
Role Description	A more detailed description of the role's purpose.
Role Type	Identifies this role as one of the following: • **System**—Role was created during installation and cannot be deleted or modified. However, it can be copied and modified to create an **Account** role. • **Account**—Role created by an administrator either by copying an existing system role or by creating a new role. For more information, see System Roles.

Role Copy Page

Use the **Role Copy** page to copy an existing **System** or **Account** role.

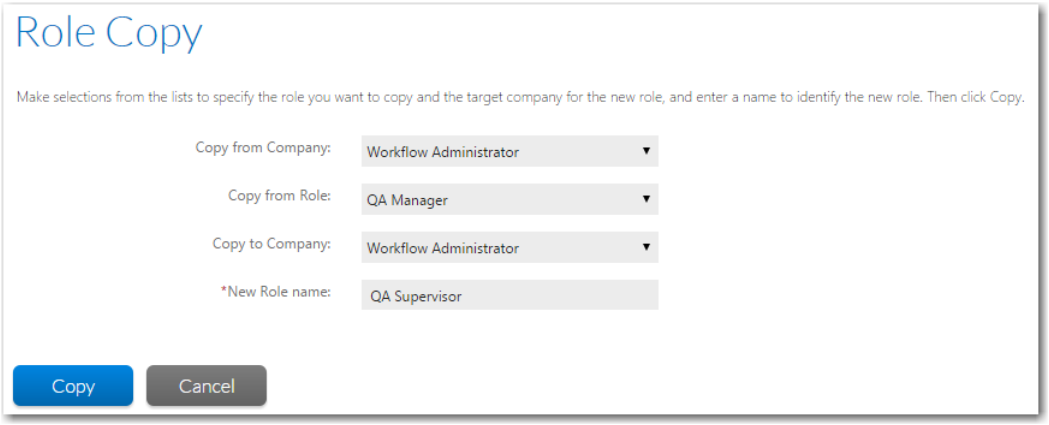


Figure 6-2: Role Copy Page

The following fields are available on the **Role Copy** page:

Table 6-6 • Fields on the Role Copy Page

Field	Description
Copy from Company	Select the name of the company that has a role that you want to copy.
Copy from Role	Select the name of the role that you want to copy. You can copy either System or Account roles.
Copy to Company	Select the name of the company that this new role is being created for.
New Role Name	Enter a name to identify this new role.

Role Details Page

The **Role Details** page allows you to view and update the details of an individual role, and to set the specific features in Workflow Manager / AdminStudio Enterprise Server that are accessible by people assigned the role. Also use this page to remove an existing role which is no longer required.

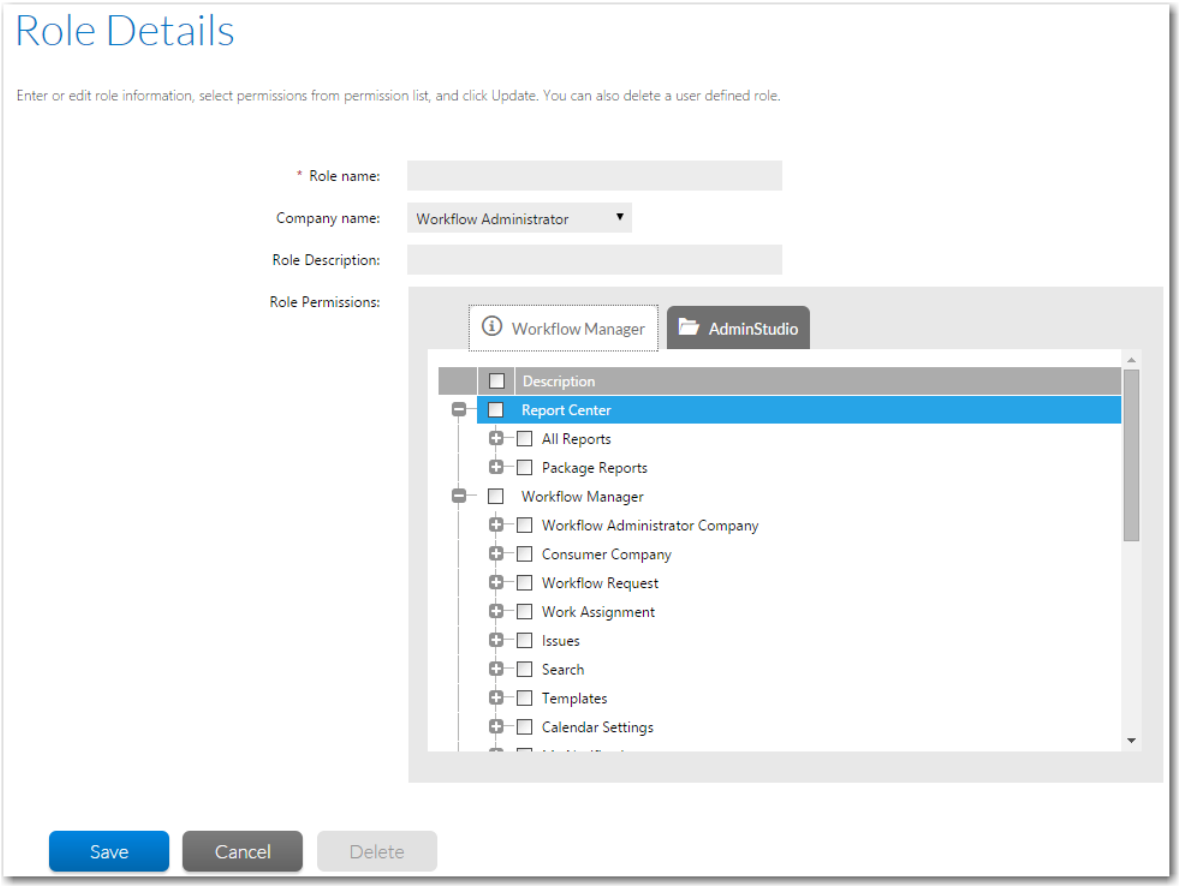


Figure 6-3: Role Details Page

The following fields are available on the **Role Details** page:

Table 6-7 • Fields on the Role Details Page

Field	Description
Role name	A brief description of the role. Roles belonging to a given company must all have distinct names.
Role company	The company that the role belongs to. This field is read-only except when you are creating a new role.
Role Description	Identify the purpose of the role.
Role Permissions list	Select the specific areas of the product that this role is to access. Expand out the tree for detailed information about the permissions available. • Selecting a check-box automatically selects all child check-boxes. • Similarly, deselecting any check-box automatically deselects all children. • You can enable access to all features in the product by clicking the **Select All** button, or remove all access by clicking **Clear All**. For more information about available permissions, see Role Permission Lists.

Managing Applications and Application Catalog Databases

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

The Application Catalog serves as the central repository for applications in all formats. You use Application Manager to manage your applications and their deployment types in the Application Catalog. Tasks you perform to manage your Application Catalog include importing applications and packages, setting up automatic package import, organizing packages into groups, viewing and editing application and package data, and viewing reports on Application Catalog data.

Information about using Application Manager is organized into the following sections:

Table 7-1 • Topics Regarding Using Application Manager

Section	Description
About the AdminStudio Host Process	Describes the AdminStudio Host Process, which is where most of Application Manager's core functionality resides.
Managing Application Catalogs	Explains how to create, connect to, search for, organize, view information about, and delete packages from an Application Catalog.

Table 7-1 • Topics Regarding Using Application Manager

Section	Description
Importing	Explains how to import the following deployment types into the Application Catalog: • Windows Installer packages (`.msi`) with any associated transforms (`.mst`) or patches (`.msp`) • Applications and packages from ConfigMgr (Formerly called as System Center Configuration Manager) • Microsoft UWP app packages (`.appx`), which is the packaging format used to distribute and install apps on Windows 8.x and 10, and is the only format allowed for Universal Windows Platform (UWP) apps. • Virtual applications in Microsoft App-V, Citrix XenApp, VMware ThinApp • Mobile apps in Apple iOS, Google Android, and Windows Store formats • macOS desktop applications in `.dmg`, `.pkg`, and Apple Mac App Store app formats • Installation packages (`.exe`), both legacy installers and complex installer executables that may contain bundled packages • PowerShell wrapped package (`.ps1`) files, which contain bundled packages • Package Feed Module • Microsoft MSIX package (`.msix`) • Merge modules (`.msm`) • OS snapshots (`.osc`)
Monitored Directory for Package Automation	Describes how to configure the Monitored Directory for Package Automation.
Identifying Vendor Command Line Arguments	Describes the command line switches for a software package which imports into the catalog and also wrap the package along with the command line switches in one single wizard.
Viewing Application Testing and Analysis Reports on the Reports Tab	Describes the available Application Catalog reports that are displayed on the Reports tab, and explains how to create your own custom reports.
Managing ConfigMgr (Formerly called as System Center Configuration Manager) Application Model Data	Describes how to view metadata for applications in Application Manager.
Managing macOS Desktop Application Metadata	Describes how to view extracted metadata for macOS desktop applications.

Table 7-1 • Topics Regarding Using Application Manager

Section	Description
Managing Mobile App Metadata	Explains how to view extracted mobile app data, how to manage iOS Enterprise Policy Configuration files, and how to view mobile app reports.
Managing App Portal Application Information	Describes how to view and edit the information used when an application is added to the App Portal catalog: title, descriptions, categories, template, and keywords.
Enabling Application Extended Attributes	Explains how to record custom data for applications by defining custom extended attributes, and displaying those attributes on a new **Extended Attributes** tab of the **Application View**.
Managing ConfigMgr (Formerly called as System Center Configuration Manager) Package Deployment Data	Describes how to view and edit data related to the deployment of packages to Microsoft System Center 2012 Configuration Manager and later.
Managing App-V Package Deployment Data	Describes how to view and edit data related to the deployment of App-V 5.0 packages.
Managing Casper Package Deployment Data	Describes how to view and edit data related to the deployment of macOS packages to JAMF Casper Server.
Managing Citrix XenApp Package Deployment Data	Describes how to view and edit data related to the deployment of packages to Citrix XenApp Server.
Managing Altiris Package Deployment Data	Describes how to view and edit data related to the deployment of packages to Symantec Altiris Server.
Managing Workspace ONE Package Deployment Data	Describes how to view and edit data related to the deployment of packages to Workspace ONE Server.
Managing App-V Virtual Environments	Explains how to create App-V virtual environments for App-V 5.0 packages for both Microsoft App-V Servers and Microsoft System Center 2012 Configuration Manager Servers.
Viewing Additional Package Data	Describes how to view and edit package metadata for Windows Installer and App-V packages in Application Manager.
Creating, Importing, and Managing PowerShell-Wrapped Packages	Details AdminStudio's support for PowerShell-wrapped packages.
Using the Conversion Wizard to Perform Virtualization or Repackaging	Explains how to upgrade an App-V 4.6 package in your Application Catalog to App-V 5.0 format. Also explains how to use the Conversion Wizard to perform conversion of Windows Installer or legacy packages to specified virtual formats or to perform repackaging.

Table 7-1 • Topics Regarding Using Application Manager

Section	Description
Using Test on Virtual Machine Wizard	Explains how to use this wizard to quickly launch a specified virtual machine and install a selected Windows Installer (`.msi`) package, App-V package (`.appv`), PowerShell-wrapped (`.ps1`) package, or installation executable (`.exe`) package (both legacy installers and complex installation executables) for testing.
Software Repository Integration into Other AdminStudio Tools	Explains how to manage a software package's associated installation files using the Software Repository.
Taking OS Snapshots	Explains how to create OS Snapshots using the OS Snapshot Wizard.
Reference	Describes the Application Manager interface, and the views, wizards and dialog boxes related to managing Application Catalog databases.

About the AdminStudio Host Process

Most of Application Manager's core functionality resides in the AdminStudio Host Process, separate from its user interface.

Using a host process gives AdminStudio better scalability and enables the development of clients that use Application Manager's core functionality. For example the Application Manager user interface and the AdminStudio PowerShell Cmdlet are now clients to this AdminStudio Host process.

Important • *AdminStudio Host must be running in order to use Application Manager or the AdminStudio PowerShell Cmdlets.*

When Application Manager is launched, the AdminStudio Host process is automatically started, and its icon is added to the System Tray.

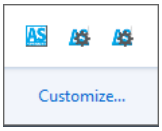


Figure 7-1: AdminStudio Host Icon in System Tray

Managing Application Catalogs

Package information is stored in Application Catalogs. When you use Application Manager, you will either automatically be logged into an Application Catalog, or you will have to connect to an existing catalog. Once you have connected to a catalog, you can manipulate data in Application Manager, including creating and organizing groups and entering extended attribute data.

The following topics relate to Application Catalog management:

- About AdminStudio Application Catalogs

- About the Application Manager Ribbon Interface
- Connecting to an Application Catalog for the First Time
- Connecting to an Existing Application Catalog
- Creating New Application Catalogs
- Upgrading an Existing Application Catalog
- Specifying a Default AdminStudio Application Catalog
- Creating Multiple Named Connections to Distribution Systems
- Setting Up a Distribution Connection for Microsoft Intune
- Integrating with Other Flexera Applications via the Flexera Service Gateway
- Managing an Application's Flexera Identifier
- Entering Microsoft ACT Database Connection Settings
- Searching an Application Catalog
- Package Customization
- Disconnecting from an Application Catalog
- Organizing Your Application Catalog Using Groups
- Deleting Packages and Applications
- Browsing to Package Location from Application Manager Tree

About AdminStudio Application Catalogs

This section provides the following information:

- Application Catalog Organization and Structure
- Overview of Application Catalogs
- Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog
- Required Permissions on Application Catalog Databases

Application Catalog Organization and Structure

AdminStudio's Application Manager uses an application-centric organization model. The tree is structured to display multiple deployment formats under a parent **Application** node. One application can have multiple packages (or deployment types) such as Windows Installer, Microsoft App-V, Citrix XenApp, VMware ThinApp, Apple iOS, Google Android, PowerShell-wrapped packages, etc.

The following image illustrates Application Manager’s tree structure.

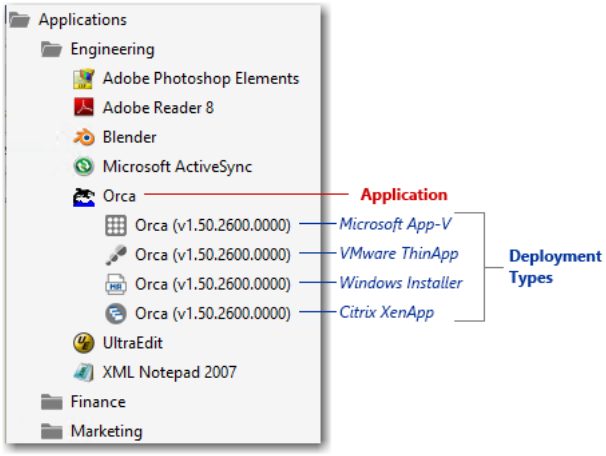


Figure 7-2: AdminStudio's Application Model

Application Manager's application-centric structure uses the following icons to represent applications and their deployment types:

Table 7-2 • Application Manager Tree Icons

Icon	Description
	Group
	Application (Default) ***Note*** • *For most applications, an icon included in the application files is used to represent it in the tree. If an application does not include an icon, this default icon is used.*
	Windows Installer package (`.msi`)
	Microsoft App-V 4.x (`.sft`) and Microsoft App-V 5 (`.appv`) virtual packages
	Citrix XenApp virtual package (`.profile`)
	VMware ThinApp virtual package (`.exe`)
iOS	Apple iOS mobile app (`.ipa`)
	Apple iOS mobile app (link to public store)
	Google Android mobile app (`.apk`)

Table 7-2 • Application Manager Tree Icons

Icon	Description
	Google Android mobile app (link to public store)
	Microsoft UWP app package (`.appx`) or Windows 8 app (`.appx`)
	Windows Store mobile app (link to public store)
	Legacy application (`.exe`)
	PowerShell-wrapped package (`.ps1`)
	Microsoft MSIX package (`.msix`)

Deployment types (packages) are grouped under Application nodes, and Applications are grouped into Groups.

Overview of Application Catalogs

To perform enterprise level testing, it is very efficient if the data can be consolidated into a single database. The AdminStudio Application Catalog database consolidates data from many installation packages in a single location.

Application Catalogs can contain the contents of: Windows Installer (`.msi`) packages; Merge Module (`.msm`) packages; OS snapshots (`.osc`); PowerShell-wrapped packages (`.ps1`); virtual packages in Microsoft App-V, Citrix XenApp, VMware ThinApp; and mobile apps in Apple iOS, Google Android, and Windows Store formats. Depending upon the import options that are set, an Application Catalog can also include binary records. Application Catalogs can also store application, workflow, permissions, Microsoft Patch, and Workflow Manager data.

Sharing Application Catalogs

In multi-user environments, the AdminStudio Application Catalog is typically shared. The AdminStudio Administrator can make a shared catalog the default catalog, which results in all AdminStudio users who use the same shared location using the same shared Application Catalog. If the shared Application Catalog is changed, users will automatically open up the new shared Application Catalog. For more information, see Specifying a Default AdminStudio Application Catalog.

Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog

***Edition** • Application Manager is included with AdminStudio Professional and Enterprise Editions.*

***Edition** • AdminStudio Enterprise Server Tools are included with AdminStudio Enterprise Edition.*

You can connect to any standalone SQL Server Application Catalog, or you can connect to the AdminStudio Enterprise Server Application Catalog:

- **Standalone**—A Standalone Application Catalog is not associated with the Enterprise Server tools. The AdminStudio client tools connect directly to the database server hosting the standalone Application Catalog.
- **AdminStudio Enterprise Server**—All of the AdminStudio Enterprise Server tools—Security Console, Reports, and Workflow Manager—are connected to the AdminStudio Enterprise Server Application Catalog. AdminStudio client tools can also connect to this Application Catalog, allowing you to store all data generated on a package in the same location, and to link packages in this Application Catalog to Workflow Manager workflow requests.

Note • *AdminStudio Enterprise Tools are always connected to the AdminStudio Enterprise Server database.*

For more information on using Application Catalogs, see the following:

- Connecting AdminStudio Client Tools to a Standalone Application Catalog
- Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

Required Permissions on Application Catalog Databases

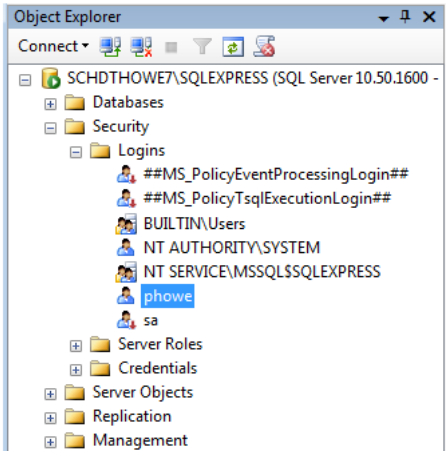
In order to connect to an AdminStudio Application Catalog database, users require the following permissions on the database:

- `db_datareader`
- `db_datawriter`
- `alter`
- `execute`

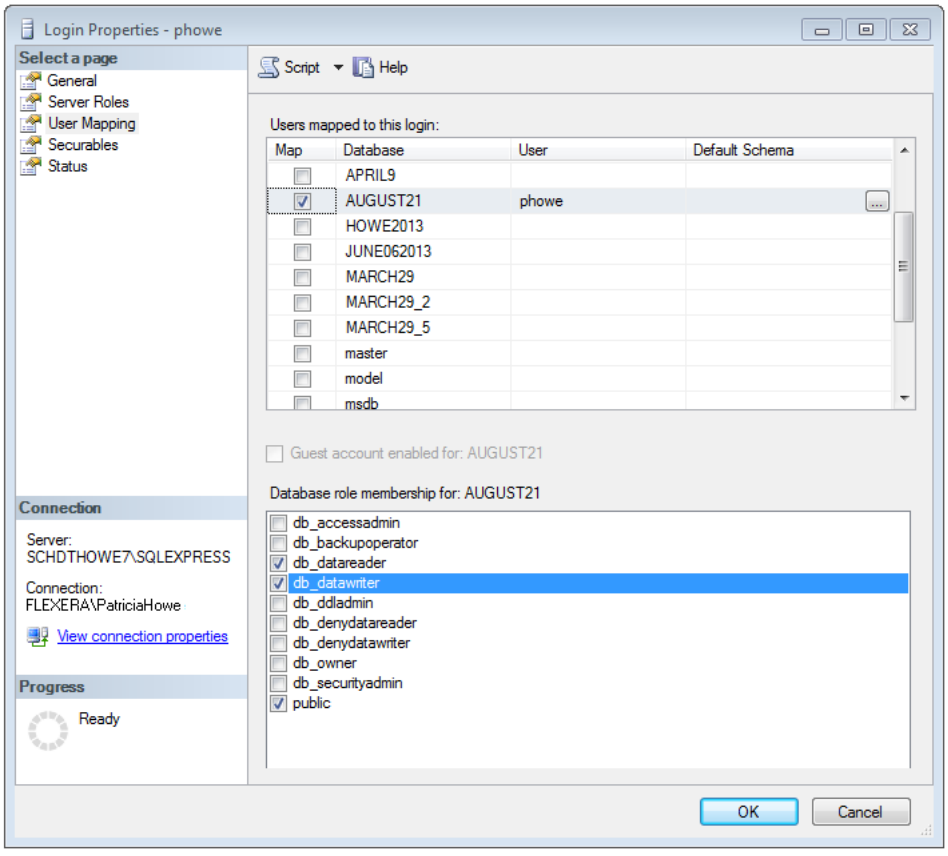
To assign these required permissions, perform the following steps:

Task ***To assign required permissions to an AdminStudio Application Catalog:***

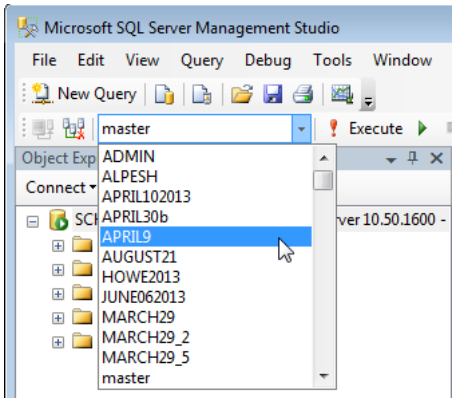
1. Open Microsoft SQL Server Management Studio.
2. In the **Object Explorer**, open the **Security > Logins** node and right-click on the user account or user group that you want to assign permissions to.



3. Select **Properties** from the shortcut menu. The **Login Properties** dialog box opens.
4. Select **User Mapping** in the tree. The **User Mapping** view of the **Login Properties** dialog box opens.



5. In the **Users mapped to this login** list, select the database that you want to assign permissions to.
6. In the **Database role membership for: [DatabaseName]** list, select `db_datareader` and `db_datawriter`.
7. Click **OK** to close the **Login Properties** dialog box.
8. In the toolbar, open the drop-down list and select the name of the AdminStudio database that you want to assign permissions to.

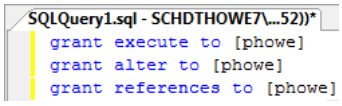


9. Next, click the **New Query** button in the toolbar to open the **Query Editor**.

10. Enter the following query:

```
grant execute to [username]
grant alter to [username]
grant references to [username]
```

For example:



11. Click the **Execute** button in the toolbar. The following message will be displayed:

```
Command(s) completed successfully.
```

About the Application Manager Ribbon Interface

Application Manager includes a ribbon interface to provide quick and easy access to Application Manager tasks.

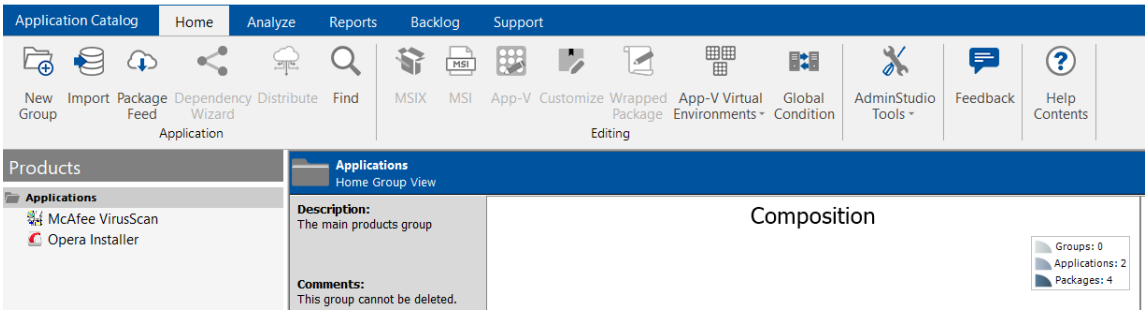


Figure 7-3: Application Manager's Ribbon Interface

The ribbon interface includes the **Application Catalog** tab menu, along with buttons that are grouped in four additional tabs: **Home**, **Analyze**, **Reports**, and **Support**.

- Application Catalog Tab Menu
- Home Tab

- Analyze Tab
- Reports Tab
- Backlog Tab
- Support Tab

Note • *The Application Manager ribbon interface incorporates all of the functionality that, in previous releases, was available in the menus.*

Application Catalog Tab Menu

The Application Catalog tab menu is opened by clicking the **Application Catalog** tab:

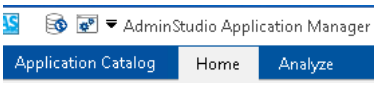


Figure 7-4: Application Catalog Tab and Other Controls

The Application Catalog tab menu includes database connection related tasks, as well as commands for setting Application Manager options, and for exiting from the application.

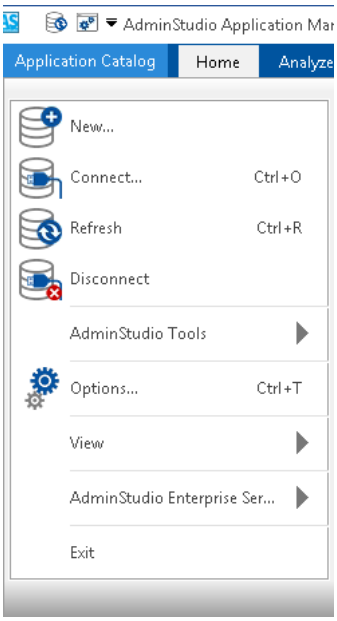


Figure 7-5: Application Catalog Tab Menu

For detailed information on each of the items on this menu, see Application Catalog Tab Menu.

Home Tab

The **Home** tab includes buttons to import packages into the Application Catalog, edit packages, use Software Repository commands, and distribute packages.

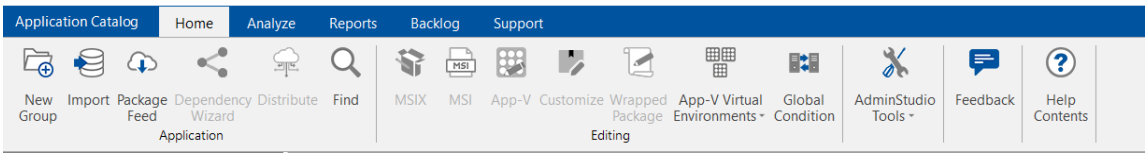


Figure 7-6: Home Tab of Application Manager Ribbon

For detailed information on each of the buttons on this tab, see Home Tab of Application Manager Ribbon.

Analyze Tab

The **Analyze** tab includes buttons to analyze a package's readiness for deployment and to detect and resolve package conflicts.

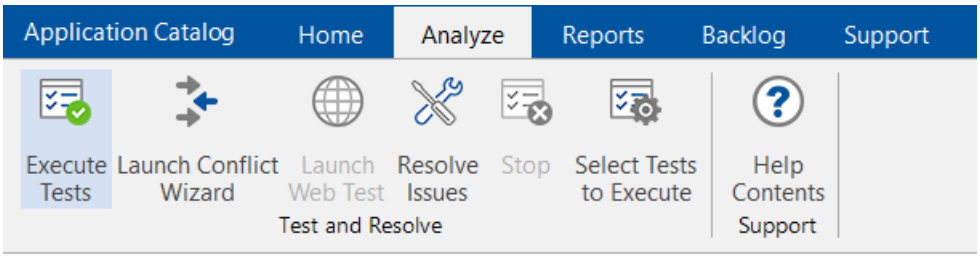
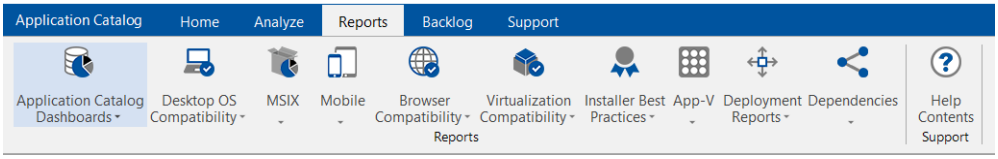


Figure 7-7: Analyze Tab of Application Manager Ribbon

For detailed information on each of the buttons on this tab, see Analyze Tab of Application Manager Ribbon.

Reports Tab

When you select the **Reports** tab of the Application Manager Ribbon, you can view detailed reports on the test results and overall status of the applications and packages in the Application Catalog. For more information, see Reports Tab of Application Manager Ribbon.



Backlog Tab

When you select the Backlog tab, you can add, import, match, customize, and execute the package requests. For more information, see Backlog Tab of Application Manager Ribbon.

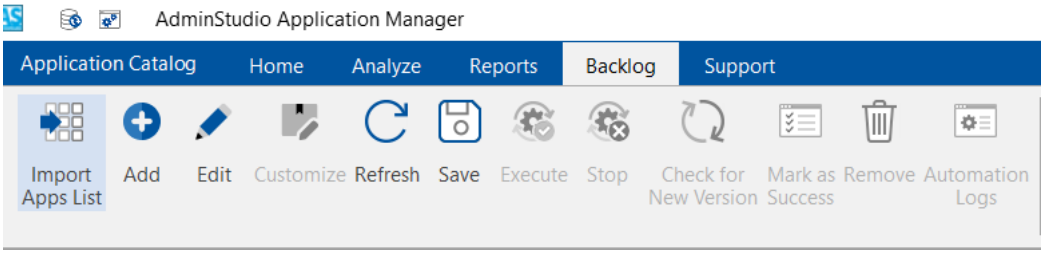


Figure 7-8: Backlog Tab of Application Manager Ribbon

Support Tab

The **Support** tab includes buttons to give you quick access to the AdminStudio help library and information specific to the current release of AdminStudio.

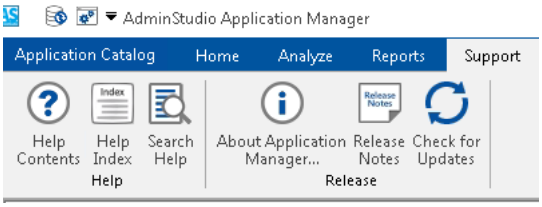


Figure 7-9: Support Tab of Application Manager Ribbon

For detailed information on each of the buttons on this tab, see Support Tab of Application Manager Ribbon.

Connecting to an Application Catalog for the First Time

When you initially open AdminStudio, because a default Application Catalog has not yet been set, the **Default Application Catalog** dialog box opens, prompting you to either create a new Application Catalog to connect to an existing Application Catalog.

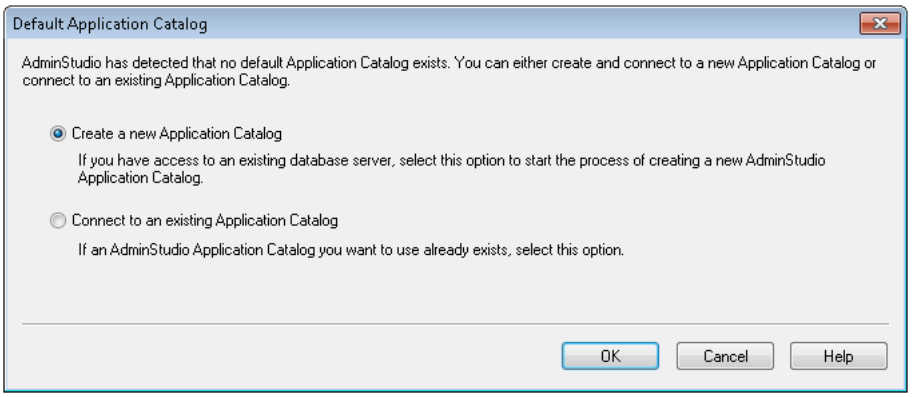


Figure 7-10: Default Application Catalog Dialog Box

To get started using AdminStudio, select one of the following options:

- **Create a new Application Catalog**—Choose this option to create a new Application Catalog database on the SQL Server database server that you specify. The **Application Catalog Wizard** opens.

- **Connect to an existing Application Catalog**—Choose this option to connect to an existing AdminStudio Application Catalog database. The **Connect Application Catalog** dialog box opens.

Note • *In previous releases of AdminStudio, you could choose the **Quick Start** option on this dialog box to instruct AdminStudio to automatically install Microsoft SQL Server 2005 Express on your machine and create a new Application Catalog. Starting with AdminStudio 11.5, the **Quick Start** option is no longer included on this dialog box. However, if you do not have access to a Microsoft SQL Server database, you can download and install the current version of Microsoft SQL Server Express from the Microsoft website. For more information, see the SQL Server Express Edition page of the Microsoft website.*

Connecting to an Existing Application Catalog

Edition • *AdminStudio Enterprise Server Tools are included with AdminStudio Enterprise Edition.*

From Application Manager, you usually connect to an Application Catalog by selecting **Connect** on the Application Catalog tab menu. From AdminStudio, you select **Connect** on the **Home** menu. AdminStudio supports SQL Server databases.

You can choose to connect to a standalone Application Catalog database or the AdminStudio Enterprise Server Application Catalog database.

- Connecting AdminStudio Client Tools to a Standalone Application Catalog
- Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

Note • *See Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog for more information.*

Note • *AdminStudio Enterprise Tools are always connected to the AdminStudio Enterprise Server database.*

Connecting AdminStudio Client Tools to a Standalone Application Catalog

To connect to an existing Standalone Application Catalog from Application Manager, perform the following steps.

Task ***To connect to an existing Standalone Application Catalog from the AdminStudio client tools:***

1. Perform one of the following:
   - **AdminStudio**—On the **Home** menu, click **Connect**.
   - **Application Manager**—On the Application Catalog tab menu, click **Connect**.

   The **Connect Application Catalog** dialog box opens, displaying three tabs: **Enterprise Server**, **Standalone**, and **Recent**.
2. Click the **Standalone** tab. The Standalone tab opens, prompting you to enter database connection information.

3. If you want this Application Catalog to be the default shared Application Catalog used in your organization, select the corresponding option at the bottom of the dialog box.
4. Select the **Server** where the Application Catalog is stored.
5. Specify how the database server should verify the authenticity of the login—either using **Windows Authentication** or **Server Authentication**. If you selected **Server Authentication**, enter the appropriate **Login ID** and **Password**.
6. In the **Catalog** box, enter the name of the Application Catalog you want to open.
7. Click **Test** to test the connection to the database.
8. Click **OK**.

Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

The AdminStudio Enterprise tools—Reports, Security Console, and (optionally) Workflow Manager—are configured during installation to connect to an Application Catalog, which is referred to as the Enterprise Server Application Catalog.

You can also connect the AdminStudio *client* tools to the Enterprise Server Application Catalog. This allows you to have all of the client and enterprise tools reference the same database.

To connect to the AdminStudio Enterprise Server Application Catalog from an AdminStudio client tool, perform the following steps.

Task ***To connect to the AdminStudio Enterprise Server Application Catalog:***

1. Perform one of the following:
   - **AdminStudio**—On the **Home** menu, click **Connect**.
   - **Application Manager**—On the Application Catalog tab menu, click **Connect**.

   The **Connect Application Catalog** dialog box opens, displaying three tabs: **Enterprise Server**, **Standalone**, and **Recent**.
2. Open the **Enterprise Server** tab.
3. The URL to the AdminStudio Enterprise Server is listed above the **Authentication** field. If the AdminStudio Enterprise Server has not yet been configured with the AdminStudio client tools (such as when it is set to its default value of `http://localhost`), click the URL link to open the **Select AdminStudio Enterprise Server URL** dialog box, and enter the URL for location of the AdminStudio Enterprise Server associated with this installation of AdminStudio.
4. From the **Authentication** list, select either **AdminStudio Enterprise Server User** or **Windows Authentication**.

   Important • *When using **AdminStudio Enterprise Server User** authentication, if Anonymous authentication is turned off in IIS, both the user's machine and the AdminStudio Enterprise Server need to be on the same domain in order for login to succeed.*
5. If you selected **AdminStudio Enterprise Server User**, enter your AdminStudio Enterprise Server **User Name** and **Password** (provided by your System Administrator).

6. Click **Login**. After a successful login, the **Provider**, **Server**, and **Catalog** name of the Enterprise Server database is listed.
7. Click **OK**.

Login Troubleshooting: Error 0x800A1518

If you are using a Workflow Manager Web Portal website with custom security zone settings and your AdminStudio Enterprise Server URL is using an IP address, you may receive Error 0x800A1518 when you attempt to connect to the AdminStudio Enterprise Server database from Application Manager or the AdminStudio interface.

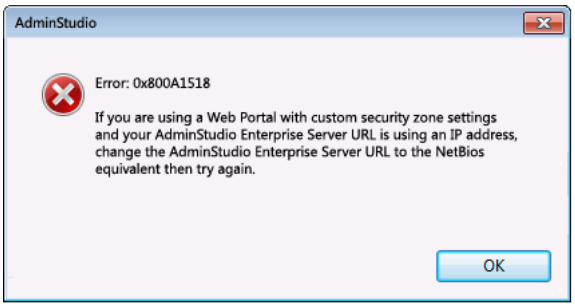


Figure 7-11: Error 0x800A1518

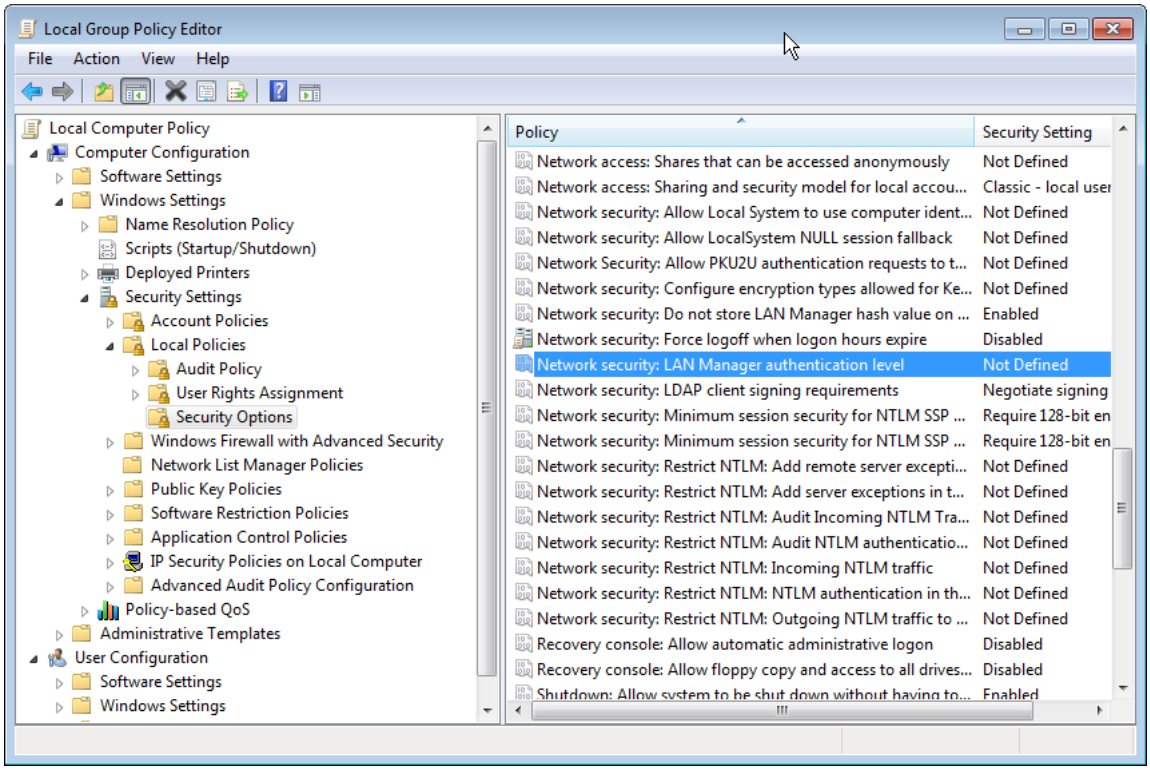
Windows has a policy setting that is not set by default: **Network security: LAN Manager authentication level**. If both the client workstation and the web server do not have this policy configured, they will sometimes not communicate properly, and this prevents AdminStudio from being able to connect to the catalog database. This can be an intermittent problem.

If you receive this error, first change the AdminStudio Enterprise Server URL to the NetBios equivalent and then try again. For example, if you are connecting to `http://120.12.1.15`, the NetBios equivalent would be `http://wfmportal`.

Also, if you are using Kerberos and different levels of authentication on your network, be sure that the workstations and servers all use the same settings. In particular, the **Network security: LAN Manager authentication level** must be set to the same value throughout the network, otherwise it may not be possible to log into a Workflow Manager Portal website that uses Windows Authentication. To set the LAN Manager authentication level, perform the following steps:

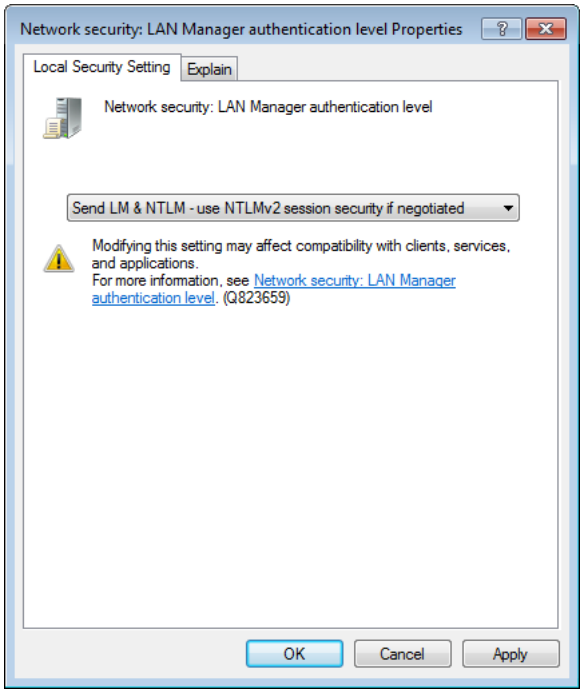
Task ***To set the Network security: LAN Manager authentication level:***

1. On the client machine, run gpe`dit.msc` to open the Windows **Local Group Policy Editor**.
2. In the **Local Computer Policy** tree, select **Computer Configuration > Windows Settings > Security Settings > Local Policies > Security Options**. The **Security Options** view opens.
3. Locate the **Network security: LAN Manager authentication level** policy.



Note • *If the **Network Security: LAN Manager authentication level** is set to **Not Defined** on both a Windows 7 client and a Windows Server 2012 server, AdminStudio cannot connect to the Windows Server 2012 and the 0x800A1518 error could be generated.*

4. Double-click on **Network Security: LAN Manager authentication level** to open its **Properties** dialog box.



5. Select **Send LM & NTLM - use NTLMv2 session security if negotiated** from the list and click **OK**.
6. Repeat the above procedure to modify the **Network Security: LAN Manager authentication level** setting on the server machine.

Creating New Application Catalogs

You can create new Application Catalogs using the AdminStudio interface, or using scripts. This section includes the following topics:

- Creating New Application Catalogs Using the AdminStudio Interface
- Creating New Application Catalogs Using Scripts

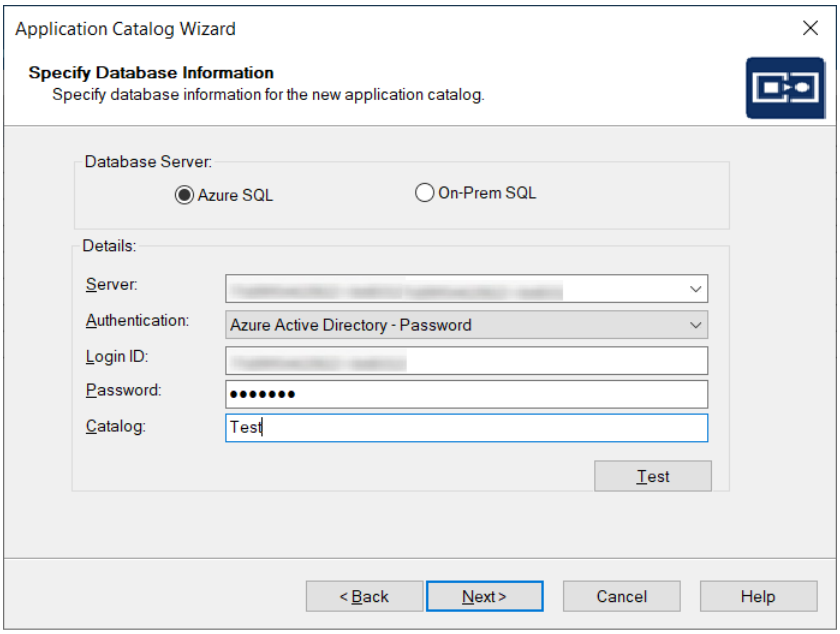
Creating New Application Catalogs Using the AdminStudio Interface

To create a new SQL / Azure Server Application Catalog database, perform the following steps.

Task ***To create a new standalone Application Catalog:***

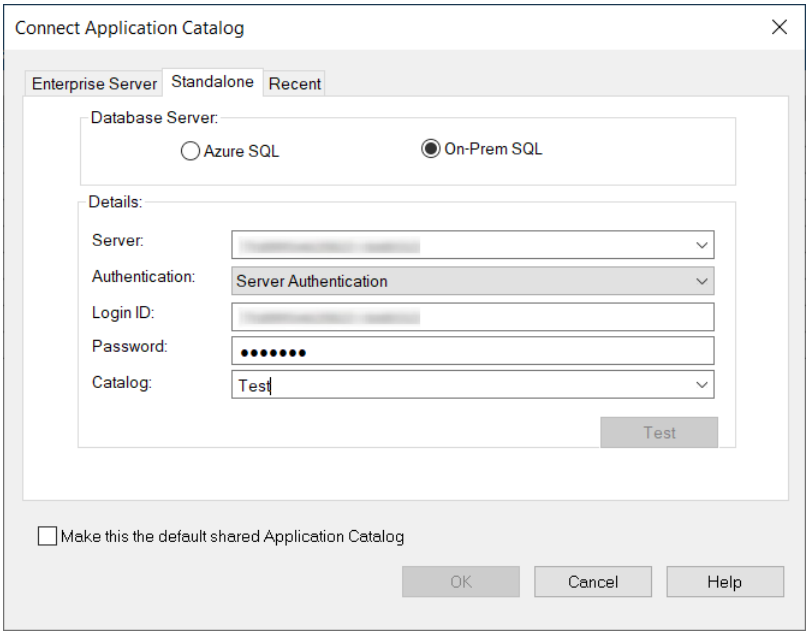
1. Click **New** on the Application Catalog tab menu (or click **Create** on the AdminStudio **Home** menu). The **Welcome** panel of the **Application Catalog Wizard** opens.
2. Click **Next**. The **Specify Database Information** panel opens.
3. On the **Specify Database Information** panel, you have the following two options:
   a. **Azure SQL—**If you select this option, perform the following steps:
      - Enter or select the name of the **Server** where this Application Catalog will be stored.

- Specify how the database server should verify the authenticity of the login—either using **Server Authentication** or **Azure Active Directory - Password**. If you selected **Server Authentication**, enter the appropriate **Login ID** and **Password**. If you selected **Azure Active Directory - Password**, enter the appropriate Azure Login ID and Password. For more information on Azure Database Creation, see here.
- In the **Catalog** field, enter the name of the Application Catalog you are creating.
- Click **Test** to test the connection to the database.



b. On-Prem SQL—If you select this option, perform the following steps:

- Enter or select the name of the **Server** where this Application Catalog will be stored.
- Specify how the database server should verify the authenticity of the login—either using **Windows Authentication** or **Server Authentication**. If you selected **Server Authentication**, enter the appropriate **Login ID** and **Password**.
- In the **Catalog** field, enter the name of the Application Catalog you are creating.
- Click **Test** to test the connection to the database.



4. Click **Next**. The **Select Software Repository Location** panel opens, prompting you to select the location where the Software Repository will store imported packages and their associated files. For more information, see Software Repository Integration into Other AdminStudio Tools.

5. If you want to store data associated with this Application Catalog's packages in the Software Repository, choose the **Enable the Software Repository** option, select a **Software Repository Location**, and enter a **Login ID** and **Password** of the **Proxy Account** that must be used to access this Repository.

Note • *The Proxy Account needs full control on the **Software Repository Location** folder at the directory level as well as at the sharing level. Only such accounts can be used as a Proxy Account to access the Software Repository.*

6. Click **Next**. The **Creating Application Catalog** panel opens and reports on the creation progress. When the Application Catalog has been created, a message appears stating that the creation was successful.

7. Click **Finish**. The new Application Catalog now opens in AdminStudio.

Tip • *The first time AdminStudio is run, anyone can create an SQL Server database (providing they have access to an SQL Server and database creation rights). However, once an AdminStudio Application Catalog has been created, the database creator becomes the Application Catalog administrator, and security rights are in place in AdminStudio.*

Creating New Application Catalogs Using Scripts

Typically, users with administrative privileges in AdminStudio use **New** on the Application Catalog Tab Menu to create a new Application Catalog.

However, because of security concerns, some database administrators may be hesitant to grant the database creation rights that are necessary to create an Application Catalog database using SQL Server to AdminStudio users. Consequently, the database administrator must manually create the database using scripts and provide the necessary read and write access for users to that database. AdminStudio is shipped with database creation SQL scripts to make it easy for database administrators to manually create new Application Catalogs.

- Scripts to Run
- Creating an Application Catalog Using Scripts

Scripts to Run

When creating an Application Catalog database using scripts, you need to execute both standard scripts and scripts for AdminStudio plug-ins:

- Standard Scripts
- Plug-In Scripts

Standard Scripts

AdminStudio is shipped with the following database creation SQL scripts:

```
AMS_System_Schema.sql
WFM_System_Data.sql
AMSCreateIndex.sql
WFM_SampleTemplates.sql
AS_System_Schema.sql
AS_ApplicationModel.sql
OsSecurityPatch.sql
MergeModule.sql
PredeploymentTest.sql
SystemManagementServer.sql
WFM_JobManager.sql
AS_TestCenter_Schema.sql
Seed_Data.sql
AS_ApplicationModelSeedData.sql
AS_TestCenter_SeedData.sql
CustomReportWizard.sql
AS_StoredProcedures.sql
VirtualizationReadiness.sql
AS_UI_Support.sql
GroupPackageTree.sql
AS_TestCenter_StoredProcedures.sql
Reporting.StoredProcedures.sql
ApplicationExtendedAttributes.sql
MobileTables.sql
MobileProgrammability.sql
MobileSeedData.sql
DAR_Schema.sql
AS_ShimDB_Schema.sql
AS_ShimDB_SeedData.sql
```

These SQL scripts are located in the following directory:

```
[AdminStudioInstallDirectory]\Support\SQL_Scripts
```

Note • *You can also find the list of the standard scripts that you are required to run to create a new database in the following nodes of the upgrade.xml file (in the AdminStudio Support folder):*

//AdminStudioUpgrade/WorkflowManager/Create/SQLServer
//AdminStudioUpgrade/AdminStudio/Create/SQLServer

Plug-In Scripts

In addition to the scripts located in the Support\SQL_Scripts directory, you also need to run any SQL script files that are found in the Common\Plugins directory. Because AdminStudio provides extensible plug-in functionality, the list of SQL scripts in this directory is not fixed. However, the following table lists the plug-in scripts that are shipped with the product:

```
Airwatch.sql
Altiris.sql
ApkDeepLink.sql
AppV.sql
AppV5Conversion.sql
AppvServer.sql
AutomatedApplicationConverter.sql
Casper.sql
IpaDeepLink.sql
Msi.sql
WebDeploy.sql
XenApp.sql
Xpf.sql
```

Important • *The order in which these scripts are run is not important; however, they must be run after the set of scripts listed in Standard Scripts.*

Creating an Application Catalog Using Scripts

To create an Application Catalog database on SQL Server, perform the following steps:

Task ***To use scripts to create an AdminStudio Application Catalog on SQL Server:***

1. Log on to your SQL Server.
2. Launch the Enterprise Manager and Query Analyzer.
3. In Query Analyzer, execute a CREATE DATABASE command to create and identify the new Application Catalog database.
4. Select the newly created database in Query Analyzer.
5. Execute the following scripts **in this order**:

   ```
   AMS_System_Schema.sql
   WFM_System_Data.sql
   AMSCreateIndex.sql
   WFM_SampleTemplates.sql
   AS_System_Schema.sql
   AS_ApplicationModel.sql
   OsSecurityPatch.sql
   ```

```
MergeModule.sql
PredeploymentTest.sql
SystemManagementServer.sql
WFM_JobManager.sql
AS_TestCenter_Schema.sql
Seed_Data.sql
AS_ApplicationModelSeedData.sql
AS_TestCenter_SeedData.sql
CustomReportWizard.sql
AS_StoredProcedures.sql
VirtualizationReadiness.sql
AS_UI_Support.sql
GroupPackageTree.sql
AS_TestCenter_StoredProcedures.sql
Reporting.StoredProcedures.sql
ApplicationExtendedAttributes.sql
MobileTables.sql
MobileProgrammability.sql
MobileSeedData.sql
DAR_Schema.sql
AS_ShimDB_Schema.sql
AS_ShimDB_SeedData.sql
```

6. Execute all of the SQL scripts found in the `Common\Plugins` directory. By default, the following scripts are found in the `Plugins` directory:

```
Airwatch.sql
Altiris.sql
ApkDeepLink.sql
AppV.sql
AppV5Conversion.sql
AppvServer.sql
AutomatedApplicationConverter.sql
Casper.sql
IpaDeepLink.sql
Msi.sql
WebDeploy.sql
XenApp.sql
Xpf.sql
```

Upgrading an Existing Application Catalog

When you attempt to open an AdminStudio 5.x to 2013 R2 Application Catalog in AdminStudio 2022 R2, you are prompted to upgrade it to use the AdminStudio 2022 R2 schema.

Log files for the upgrade are created in the following directory:

AdminStudio Shared Directory`\ConflictSolver\Logs`

Note • *Note the following regarding upgrading an existing Application Catalog:*

- *The upgrade of AdminStudio 3.0, 3.01, and 3.5 databases is not supported by AdminStudio 7.0 or later.*
- *Starting with AdminStudio 8.0, Microsoft Access databases are not supported.*
- *Starting with AdminStudio 9.01, Oracle databases are not supported.*

- *When an SQL Server Application Catalog database is upgraded, the old tables are not dropped from the Application Catalog.*

Important • *Specific permissions are required to upgrade an Application Catalog. See Required Permissions on Application Catalog Databases.*

Upgrading an Application Catalog Using Scripts

To see a list of the scripts that you are required to run to upgrade an existing Application Catalog database, view the following nodes of the `upgrade.xml` file (in the AdminStudio `Support` folder):

```
//AdminStudioUpgrade/WorkflowManager/Upgrades/Upgrade/SQLServer
//AdminStudioUpgrade/AdminStudio/Upgrades/Upgrade/SQLServer
```

For detailed information on upgrading an Application Catalog using scripts, see the *AdminStudio Installation Guide*.

Specifying a Default AdminStudio Application Catalog

You can specify a default Application Catalog so that each time you open AdminStudio, you will be prompted to login to the same Application Catalog database.

You can also configure your enterprise so that *all of the users at your enterprise* will be prompted to login to the same Application Catalog each time they open AdminStudio.

- Setting a Default Application Catalog for Yourself
- Setting the Default Application Catalog for Your Enterprise

Setting a Default Application Catalog for Yourself

Whenever you connect to an Application Catalog, you can designate it as the default Application Catalog by selecting the **Make this the shared default Application Catalog** option on the **Connect Application Catalog** dialog box.

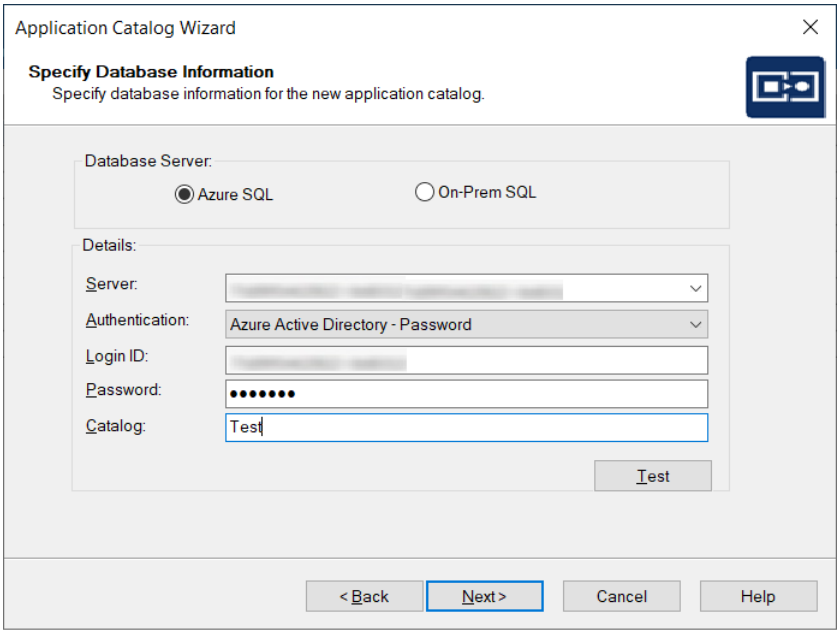


Figure 7-12: Default Shared Application Catalog Option on the Connect Application Catalog Dialog Box

Setting the Default Application Catalog for Your Enterprise

To configure your enterprise so that *all of the users at your enterprise* will be prompted to login to the same Application Catalog each time they open AdminStudio, all users need to be connected to the same `AdminStudio Shared.ini` file that is stored in a shared network location that is available to all users. To set a default Application Catalog for all AdminStudio users at your enterprise, you need to edit the `Shared AdminStudio.ini` file.

Task ***To set the Default Application Catalog for your enterprise:***

1. First, the AdminStudio System Administrator needs to perform the following steps to set the default Application Catalog for the enterprise:

   a. Locate and copy the following file on the machine where you installed AdminStudio:

   `C:\Program Files (x86)\AdminStudio Shared\Shared AdminStudio.ini`

   b. Copy this file to a shared network location that is accessible to all of the users in your enterprise.

   c. Open the `Shared AdminStudio.ini` file that you just copied to a shared network location.

2. Insert one of the following in the `[Database Settings]` section of the `Shared AdminStudio.ini` file, depending upon the authentication type you are using:

Authentication Type	Entry

Windows Authentication	`[Database Settings]`
	`DefaultDatabase=PROVIDER=MSOLEDBSQL19; User ID=userid;`
	`Initial Catalog=nameofdatabase; Data Source=nameofsqlserver;`
	`Integrated Security=SSPI;Use Encryption for Data=False;`

Authentication Type	Entry
Server Authentication	`[Database Settings]` `DefaultDatabase=PROVIDER=MSOLEDBSQL19; User ID=userid; PWD=password; Initial Catalog=nameofdatabase; Data Source=nameofsqlserver;Use Encryption for Data=False;`

Azure SQL Server Authentication	When using this authentication method, you need to include the User ID and Password elements to provide the Azure Server login credentials:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`User ID=UserName;`
	`Password=passwd;`
	`Initial Catalog=CatalogName;`
	`Use Encryption for Data=False;`

Azure Active Directory Authentication	When using this authentication method, you need to include the User ID and Password elements to provide the Azure Active Directory login credentials:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`User ID=UserName;`
	`Password=passwd;`
	`Authentication=ActiveDirectoryPassword;`
	`Initial Catalog=CatalogName;`
	`Use Encryption for Data=False;`

Note • *AdminStudio stores an encrypted version of the database connection string in an INI key called `SecuredDatabaseKey`. If this is present, then it takes precedence over the `DefaultDatabase` string. If the password is omitted in the `DefaultDatabase` string, then the user will be prompted for it when starting AdminStudio. And when the user provides the password, it will be saved as part of the encrypted `SecureDatabaseKey` string (and not in the clear text `DefaultDatabase` string).*

3. Next, each AdminStudio user in the enterprise needs to perform the following steps to set the location of *their* **AdminStudio Shared Location** directory to the same shared network directory that the System Administrator configured.

   a. Launch AdminStudio.

   b. On the **Tools** menu, click **Options**. The AdminStudio **Options** dialog box opens.

   c. Open the **Locations** tab.

   d. Set the **AdminStudio Shared Location** to the shared network location provided by your System Administrator.

Caution • *If a user is not assigned to a Role that has the **Modify AdminStudio Tools Options Dialog** permission, they cannot change the **AdminStudio Shared Location** setting on the **Options** dialog. In this situation, the location of the AdminStudio Shared Location would be set during installation.*

Note • *The Roles assigned to a user determine that user's permissions:*

- *The **Create** and **Connect** options on the **Catalog** menu on the AdminStudio interface (and the **New** and **Connect** options on the Application Catalog tab menu) are disabled for users that are not assigned to a Role that has permission to perform those actions.*
- *Users that are assigned to Roles that have the **Modify AdminStudio Tools Options Dialog** permission can change the location of the **AdminStudio Shared Location** setting on the **Options** dialog (accessed by selecting **Options** on the **Tools** menu from the AdminStudio Interface). For those users who do not have that permission, **Options** on the **Tools** menu is disabled, so they are unable to change the location of the AdminStudio Shared location.*

Creating Multiple Named Connections to Distribution Systems

You can define multiple named connections to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, Symantec Altiris Server, Microsoft App-V Server, Casper Suite Server, Microsoft Intune, and Workspace ONE Server distribution systems. This enables you to both have multiple connections easily available during import and distribution, and to refer to those connection settings by name in AdminStudio PowerShell Cmdlets commands.

Important • *In order to distribute packages to an App-V Server, the WinRM service must be running, and the App-V Server must be in the list of trusted hosts. For more information, see Microsoft App-V Server Distribution Requirements.*

You need to specify named connections to distribution systems in order to enable Application Manager to perform the following tasks:

Table 7-3 • Application Manager Integration with Distribution Systems

Category	ConfigMgr	System Center Configuration Manager		Citrix XenApp Server	Symantec Altiris Server	Microsoft App-V Server	Workspace ONE Server	Casper Server	Microsoft Intune
	2111	2012	2007						
Import packages from	✓	✓	✓						
Distribute applications to	✓	✓		✓	✓	✓	✓	✓	✓

Table 7-3 • Application Manager Integration with Distribution Systems

Category	ConfigMgr 2111	System Center Configuration Manager 2012	System Center Configuration Manager 2007	Citrix XenApp Server	Symantec Altiris Server	Microsoft App-V Server	Workspace ONE Server	Casper Server	Microsoft Intune
Supported deployment types:	Windows Installer App-V 4.x, 5.0 Apple iOS (local file and public store link) Google Android (local file and public store link) Windows Store (local file and public store link) Legacy installer PowerShell-wrapped package MSI Package MSIX Package Microsoft UWP app packages	Windows Installer App-V 4.x, 5.0 Apple iOS (local file and public store link) Google Android (local file and public store link) Windows Store (local file and public store link) Legacy installer PowerShell-wrapped package MSI Package	Windows Installer App-V 4.x Legacy installer	Citrix XenApp App-V 4.x	Windows Installer VMware ThinApp Legacy installer PowerShell-wrapped package	App-V 5.0	Apple iOS (local file and public store link) Google Android (local file and public store link) MSI Package	macOS (local file and public store link)	MSI Package MSIX Package Intunewin
Distribute packages to	✓	✓	✓						
View application deployment status	✓	✓			✓	✓	✓	✓	✓

Table 7-3 • Application Manager Integration with Distribution Systems

Category	ConfigMgr 2111	System Center Configuration Manager 2012	System Center Configuration Manager 2007	Citrix XenApp Server	Symantec Altiris Server	Microsoft App-V Server	Workspace ONE Server	Casper Server	Microsoft Intune
View and edit package deployment data	✓	✓	✓	✓	✓	✓	✓	✓	✓

You specify distribution system connection settings on the **Distribution System** tab of the **Options** dialog box.

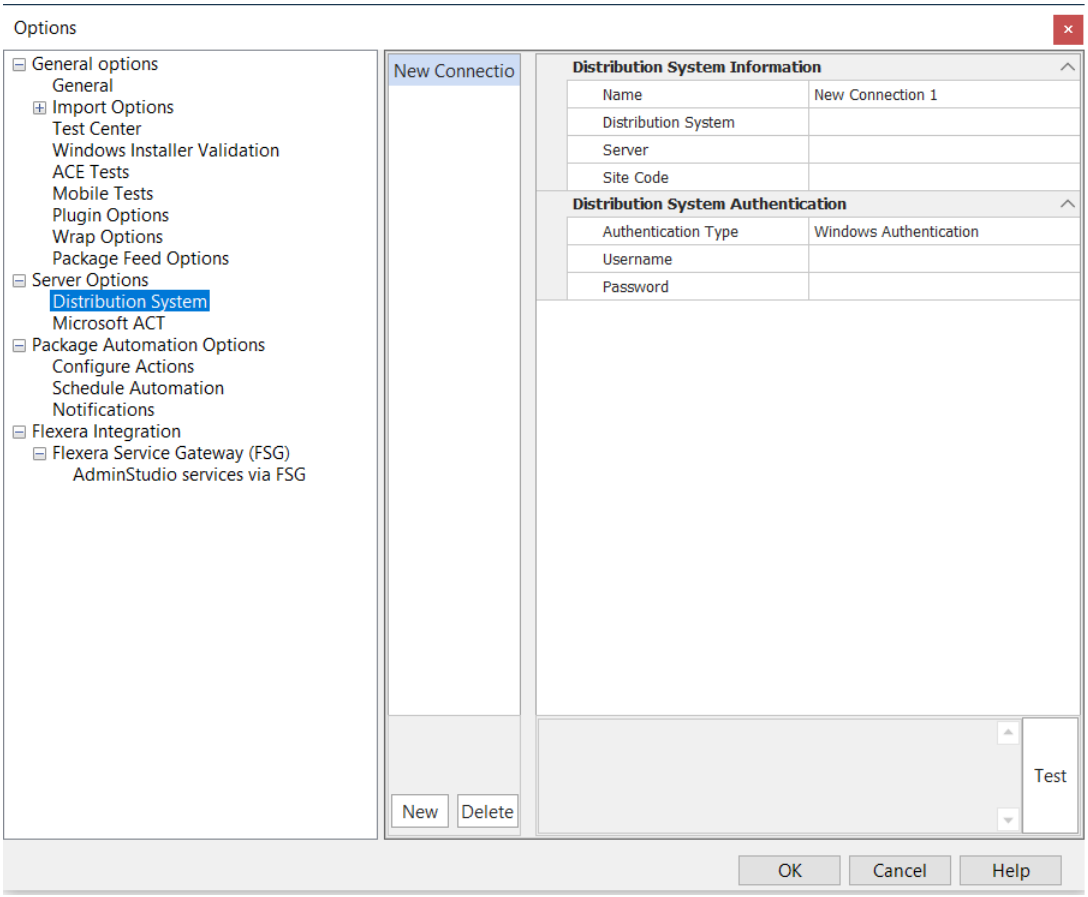


Figure 7-13: Distribution System Information Tab of Options Dialog Box

After you enter these connection settings, those distribution systems will be available to you when importing and distributing applications and packages. Also, information from those distribution systems will be displayed:

- **ConfigMgr (Formerly called as System Center Configuration Manager)**—The **Deployment Data** tab of the **Home Deployment Type View** lists deployment data that will be used by ConfigMgr (Formerly called as System Center Configuration Manager) when deploying the selected package.
- **Microsoft Intune**—The **Deployment Data** tab of the **Home Deployment Type View** lists deployment data that will be used by Intune when deploying the selected package.

- **Citrix XenApp**—For App-V 4.x and Citrix XenApp packages, the **XenApp Deployment Data** tab of the **Home Deployment Type View** will be displayed, listing deployment data that will be used by Citrix XenApp Server when deploying the selected package.
- **Altiris**—For Windows Installer, VMware ThinApp packages, the **Altiris Deployment Data** tab of the **Home Deployment Type View** will be displayed, listing deployment data that will be used by Symantec Altiris Server when deploying the selected package.
- **Casper**—For macOS packages, the **Casper Deployment Data** tab of the **Home Deployment Type View** will be displayed, listing deployment data that will be used by Casper when deploying the selected package.
- **App-V Server**—For App-V 5.0 packages, the **App-V Deployment Data** tab of the **Home Deployment Type View** will be displayed, listing deployment data that will be used by the App-V Server when deploying the selected package.
- **Workspace ONE**—For iOS, Google Android, MSI, and EXE packages, the **Workspace ONE Deployment Data** tab of the **Home Deployment Type View** will be displayed, listing deployment data that will be used by Workspace ONE Server when deploying the selected package.

For instructions on setting up a named connection to a distribution system, see:

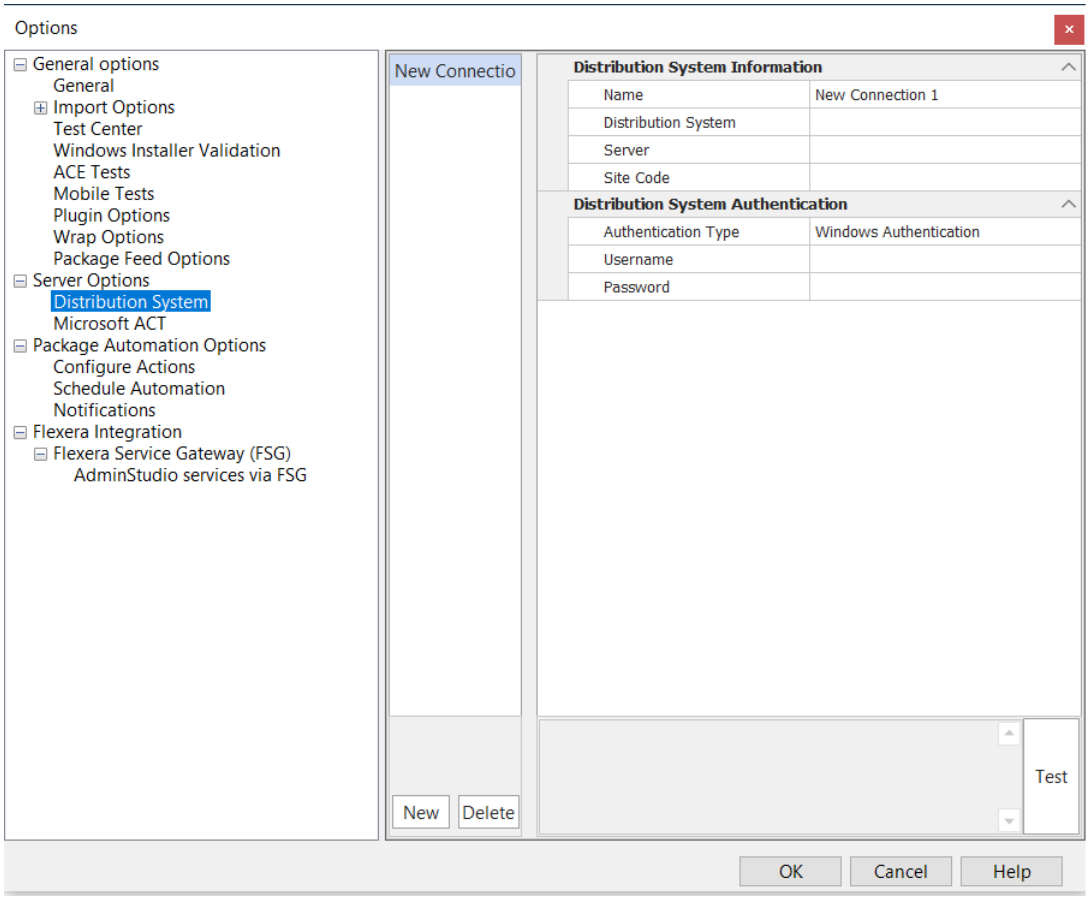
- Creating a New Distribution System Connection Setting
- Editing an Existing Distribution System Connection Setting

Creating a New Distribution System Connection Setting

To create a new named connection to a distribution system, perform the following steps:

Task ***To create a new named connection to a distribution system:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Servers Options**, select **Distribution System**. The **Distribution System** tab opens, and lists all defined connections.



3. Click **New**. A new set of empty connection setting fields is displayed.
4. In the **Name** field, enter a name to identify this new named connection to a distribution system.
5. From the **Deployment Type** list, select one of the following to identify the distribution system technology of this new named connection:
   - **Workspace ONE Distribution Plugin**
   - **Altiris Distribution Plugin**
   - **App-V Server Distribution Plugin**
   - **Casper Distribution Plugin**
   - **Custom Distribution Plugin**
   - **Intune Distribution Plugin**
   - **ConfigMgr Deployment Plugin**
   - **XenApp Distribution Plugin**
6. In the **Server** field, enter the name of your distribution system server.
7. In the **Site Code** field, enter the code that identifies your distribution system site.

Note • *If you are creating a named connection to a XenApp, Altiris, or App-V server, leave the* ***Site Code*** *field blank. If you are creating a named connection to Casper, this field will not be displayed.*

8. Under **Distribution System Authentication**, set the **Authentication Type** field to either **Windows Authentication** or **Server Authentication** to identify the authentication type you are going to use to access the specified distribution system.

   If you selected **Server Authentication**, you need to also enter a **Username** and **Password**.

9. Under **Publish Location Information**, set the **Authentication Type** field to either **Windows Authentication** or **Server Authentication** to identify the authentication type you are going to use to access the shared location where you will be publishing packages during distribution.

   If you selected **Server Authentication**, you need to also enter a **Username** and **Password**.

10. (Casper only) In the **Distribution Point** field, enter the Casper distribution point you want to distribute packages to.

Note • *Casper supports multiple server infrastructures, but AdminStudio only supports the File Share Distribution Points infrastructure, and copies packages to a UNC File Share Distribution Point in Casper. AdminStudio currently does not support copying packages to JDS Instances, Cloud Distribution Points, Software Update Servers, or NetBoot Servers.*

11. (Custom Distribution Plugin only) Under **Script Information**:

    - In the **Script File** field, click **Browse** and select the script file (.ps1) that you want to add. This is an optional field.

Note • *It is required that the specified PowerShell script file must be present in the AdminStudio installed machine.*

Note • *The specified PowerShell script file will be executed locally on the AdminStudio installed machine during publish.*

    - In the **Script Parameters**, enter the parameters used in the PowerShell script with the appropriate AdminStudio placeholders. This is an optional field.

      The **Script Parameters** field must be used to pass the AdminStudio placeholders value to the parameters in the PowerShell script file. During the execution of the script, AdminStudio will replace the specified parameters with the value of the placeholders for the package being published. The below Table lists all supported AdminStudio placeholders with their values.

Placeholders	Values
[ProductName]	Name of the application node as it appears in the Application tree in the Catalog.
[Vendor]	Value of the **Manufacturer** property in the **Package Information** tab.
[Version]	Value of the **Software Version** property in the **Package Information** tab.
[PackagePublishPath]	Path to the package file after the copy.
[InstallCmdLine]	Value of the **Install command line** property in the **Programs** tab under **Deployment Data** tab.

Placeholders	Values
[UnInstallCmdLine]	Value of the **Uninstall command line** property in the **Programs** tab under **Deployment Data** tab.
[Platform]	Operating system platform supported by the package.
[RepairCommandLine]	Value of the **Repair command line** property in the **Programs** tab under **Deployment Data** tab.
[Language]	Value of the **Language** property in the **Package Information** tab.
[PackagePublishFolder]	Path to the folder where the package copied during publish.
[TransformFilePath]	Value of the **Transforms** property in the **Package Information** tab.
[IconPath]	Value of the **Icon file** property in the **General Information** tab.
[SetupType]	Package type extension. For Examples: MSI, EXE, etc

Note *• Enter the script parameters in comma-separated format with no gap:*

For Example: -ProductName [ProductName],-Vendor [Vendor],-Version [Version]

Note *• If you use any other delimiter than comma, an error message will be generated.*

12. In the **Location to Publish Package** field, enter or browse to the shared location where you will be publishing packages during distribution.

Note *• The fields in the* ***Publish Location Information*** *section are not required when setting up a connection to Workspace ONE Server. Applications are published directly to the Workspace ONE Server, not to a shared location.*

Note *• You can add the following placeholders in the shared location path to publish packages. These placeholders are applicable only for* ***Custom Distribution Plugin****.*

- *[ProductName]*
- *[Vendor]*
- *[Version]*
- *[Platform]*
- *[Setuptype]*
- *[Language]*

For Example:

When publishing 7-Zip v19.00, if the following shared path is specified in the **Location to Publish Package** field:

\\10.XXX.XXX.XXX\PackageRepo\[Vendor]\[ProductName]\[Version]

then the placeholders will be resolved to create the shared path as shown below:

\\10.XXX.XXX.XXX\PackageRepo\Igor Pavlov\7-Zip\19.00

Important • *The **Custom Distribution Plugin** can be used to publish applications to an endpoint management system that is not supported out-of-the-box by AdminStudio. During publish to a Custom Distribution System, the selected package will be first copied to the shared path specified in the **Location to Publish Packages** field, and then the PowerShell script file specified in the **Script File** field will be executed. The PowerShell script can perform any action, which may include importing of the copied package into any end point management system.*

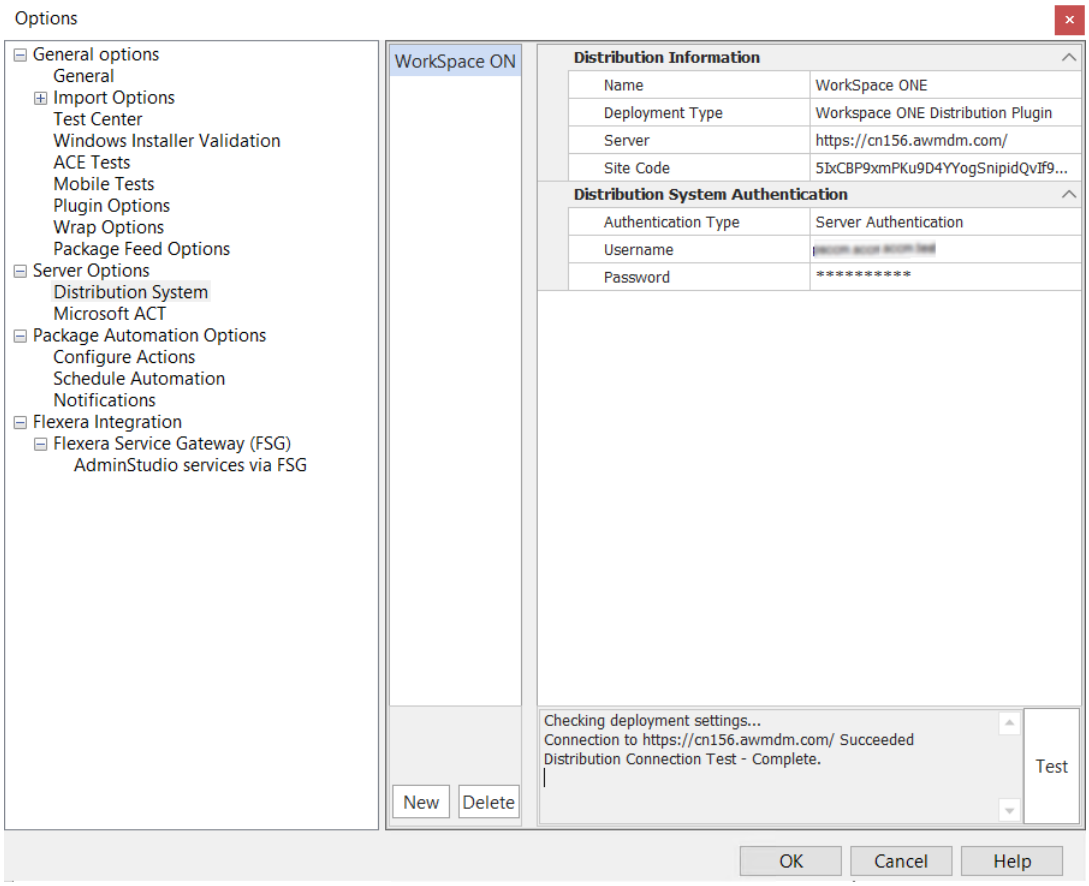
13. If you want to add another named connection, click **New** and repeat the above process.
14. Click **OK** to close the dialog box.

Editing an Existing Distribution System Connection Setting

To edit the settings of an existing connection to a distribution system, perform the following steps.

Task ***To edit an existing named connection to a distribution system:***

1. On the Application Catalog tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Servers Options**, select **Distribution System**. The **Distribution System** tab opens, and lists all defined connections.



3. In the left pane of the **Distribution System** tab, select the name of the connection that you want to edit. The selected connection's settings are displayed.
4. Make any desired edits, as described in Creating a New Distribution System Connection Setting.

Note • *You cannot edit the **Name** and **Deployment Type** fields. If you want to change those fields, you would have to delete the connection and recreate it.*

5. Click **OK**. Your edits will be saved.

Note • *If you want to delete a connection, select its name and then click **Delete**.*

Setting Up a Distribution Connection for Microsoft Intune

To obtain the required credentials to set up a distribution connection in AdminStudio for Microsoft Intune, perform the following steps.

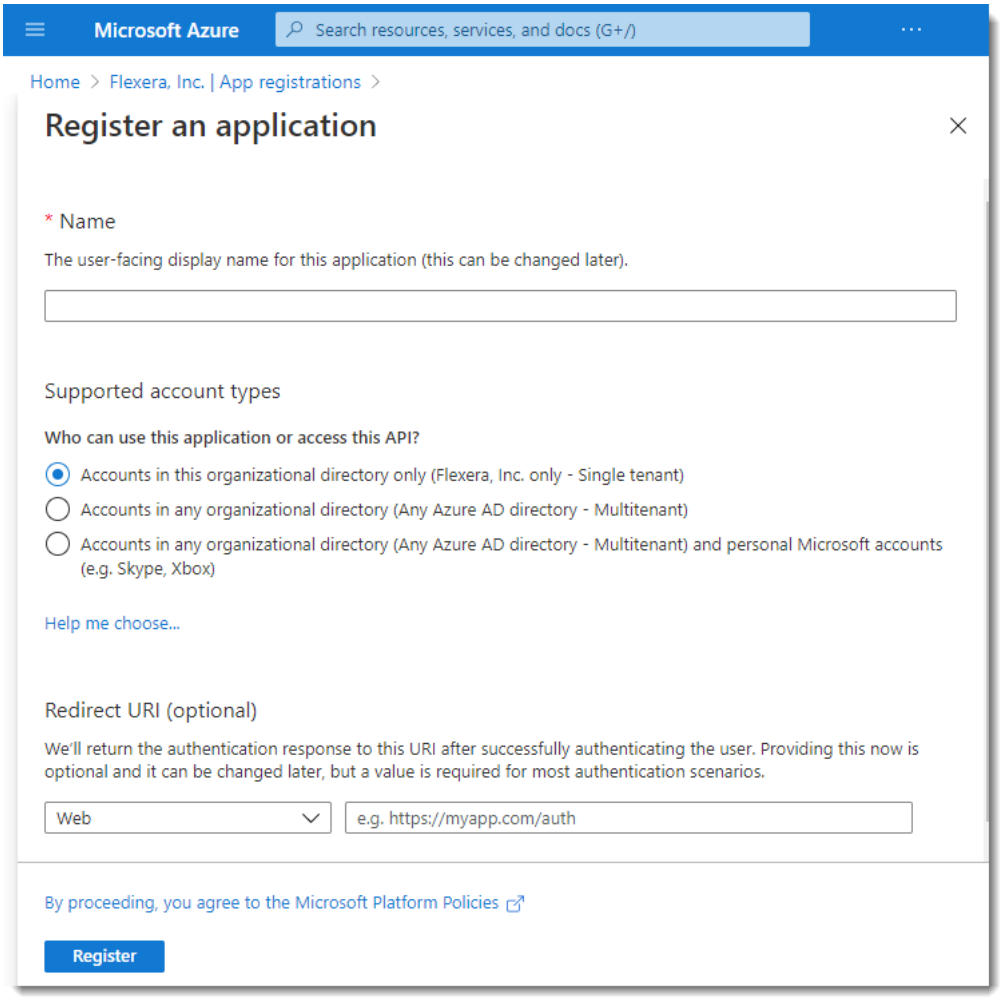
Note • *For Microsoft Azure support, the following dependencies are required:*

- *Microsoft.Graph.dll*
- *Microsoft.Graph.Core.dll*
- *Microsoft.Identity.Client.dll*
- *IntunewinAppUtil.exe*

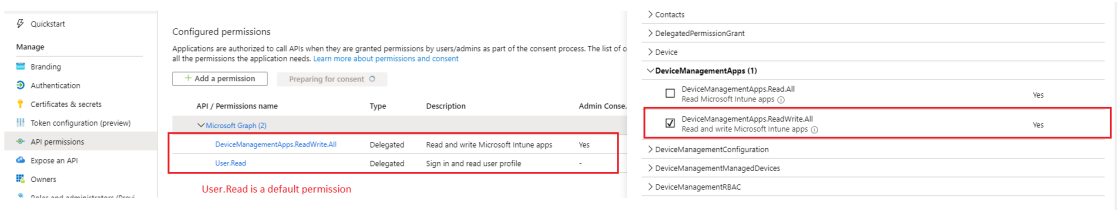
Task ***To obtain required credentials to set up a distribution connection for Microsoft Intune:***

1. Obtain login credentials for Intune services on Microsoft Azure.
2. Log in to Microsoft Azure using the obtained credentials:

   https://azure.microsoft.com/en-in/account/
3. Once logged in, click on **Portal** in the right top corner next to **My account**.
4. From the list of available Azure services, choose **Azure Active Directory**.
5. Click on **App registrations** in the left pane.
6. Click on **New registration** and enter values to complete the registration.



7. Click **Register**. The new registered application will be listed.
8. Copy the **Application (client) ID** and the **Directory (tenant) ID**, which will be used later for authentication purposes.
9. Click **API permissions** from left navigation panel.
10. Click **Add a permission**. The **Request API permissions** panel appears.
11. Click **Microsoft Graph**.
12. Click **Delegated permissions**.
13. Under **DeviceManagementApps** select **DeviceManagementApps.ReadWrite.All** and click **Add permissions**.



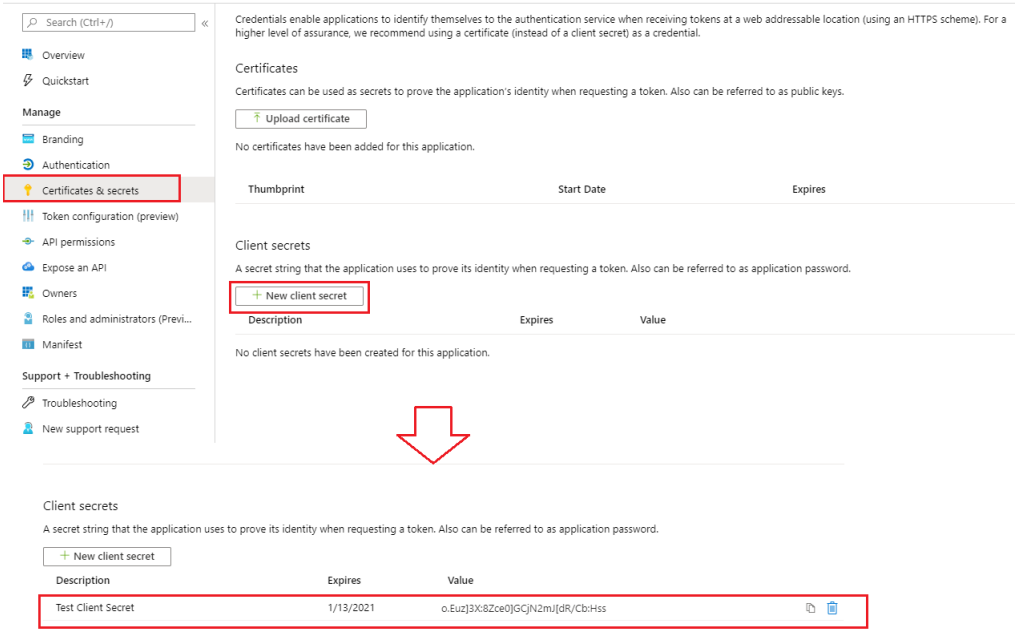
14. Click on **Grant admin consent for Flexera Development Account** and consent to the permissions. The status will turn to Granted. Make sure the permissions are of type **Delegated**.

15. If the customer wishes to authenticate using a client secret instead of providing User Name and Password interactively, perform the following steps:

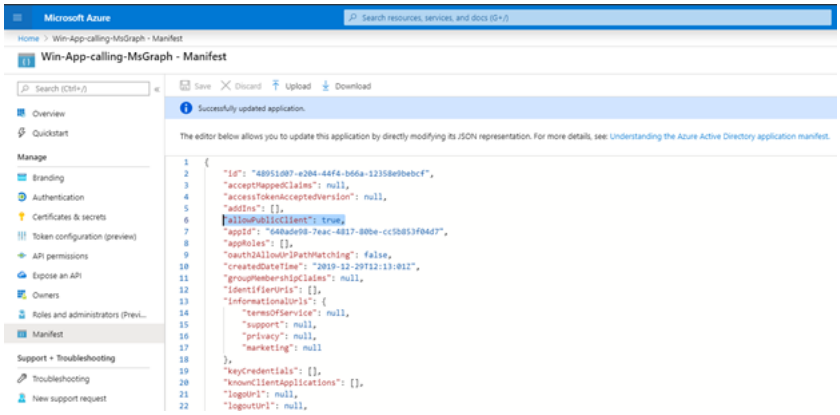
    a. Navigate to **Certificates & secrets** and click **New client secret**.

    b. Provide a description to describe your client secret, choose the expiration date you wish to set, and click **Add**. The **Client secret** will then be listed in the **Client Secrets** section.

    c. Copy the client secret **Value**, as it will only be visible and available to the user now.



16. This step is optional as it is only required when AdminStudio wishes to save the user's credentials for authentication. Under **Manage** in left navigation panel, click **Manifest**. Make sure the application's **allowPublicClient** property in the manifest file is set to **true** (in order for authentication with User name and Password to work).



For more information on Intune, see the following:

- Acquiring Tokens
- How to use Azure AD to access the Intune APIs in Microsoft Graph

See Also
Creating a New Distribution System Connection Setting

Integrating with Other Flexera Applications via the Flexera Service Gateway

AdminStudio can communicate with App Portal, FlexNet Manager Suite, and Workflow Manager via the Flexera Service Gateway.

- Overview of Unified Application Management Workflow
- Enabling Communication Between Products
- Setting Up AdminStudio Accounts
- Synchronizing Applications with App Portal and FlexNet Manager Suite
- Flexera Service Gateway Messages
- App Portal Only Integration

Overview of Unified Application Management Workflow

Because AdminStudio can communicate with App Portal, FlexNet Manager Suite, and Workflow Manager via the Flexera Service Gateway, you can implement a unified application management workflow.

In this integrated solution, a Flexera Identifier—a unique identifier assigned to applications by the FlexNet Manager Suite and stored in its Application Recognition Library (ARL)—is used to maintain application identity across products.

Note • *The ARL uniquely identifies over 110,000 applications (including multiple versions and editions) from over 14,000 publishers.*

Because AdminStudio, App Portal, and FlexNet Manager Suite use the same ID to identify an application, you can implement a unified application management workflow, which enables you to automatically and efficiently manage your application licenses:

- **AdminStudio obtains the Flexera Identifier from FlexNet Manager Suite**—When an application is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and obtains the application's Flexera Identifier.
- **AdminStudio creates catalog item in App Portal**—When you publish an application from AdminStudio to System Center 2012 Configuration Manager, Symantec Altiris Server, or Casper Suite Server, you can choose to have a catalog item for that application automatically created in App Portal, identified by the same Flexera Identifier (assuming that App Portal is also connected to that distribution system).
- **App Portal and FlexNet Manager Suite share license information**—Because an application's identity is maintained between App Portal and FlexNet Manager Suite, automatic license management can be performed.

The following diagram gives you an overview of how the integrated Flexera applications communicate—via the Flexera Service Gateway—when performing the tasks involved in a single application's life cycle.

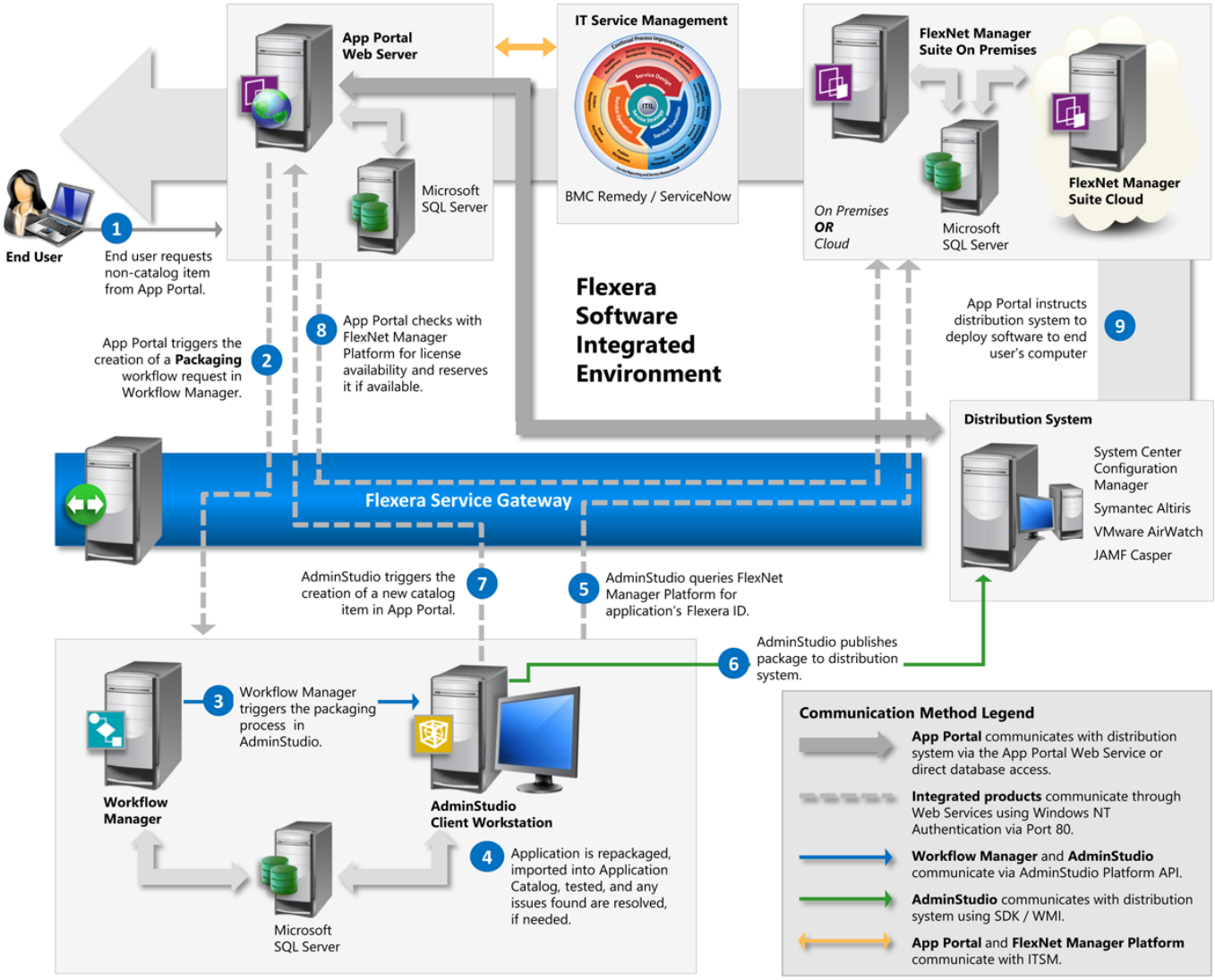


Figure 7-14: Flexera Integrated Environment

This application life cycle workflow includes the following steps:

Table 7-4 • Flexera Integrated Environment Workflow

#	Step	Description
1.	**End user requests non-catalog item from App Portal.**	Using the App Portal website, an end user submits a request for a software item that is not currently in the App Portal catalog.
2.	**App Portal triggers the creation of a Packaging workflow request in Workflow Manager.**	After the end user's request for new software is approved, App Portal triggers the creation of a **Packaging** workflow request in Workflow Manager.

Table 7-4 • Flexera Integrated Environment Workflow

#	Step	Description
3.	**Workflow Manager triggers the packaging process in AdminStudio.**	Workflow Manager can be configured to trigger AdminStudio tasks such as repackaging, importing a package into the Application Catalog, performing tests, and resolving issues that are found, if needed.
4.	**Application is repackaged, imported into Application Catalog, tested, and any issues found are resolved, if needed.**	The package is ready for deployment.
5.	**AdminStudio queries FlexNet Manager Suite for application's Flexera Identifier.**	When the package is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and obtains the application's Flexera Identifier.
6.	**AdminStudio publishes package to supported distribution system.**	After the software has been repackaged and tested, AdminStudio publishes the application to ConfigMgr (Formerly called as System Center Configuration Manager), Symantec Altiris Server, JAMF Casper Suite Server, or VMware Workspace ONE.
7.	**AdminStudio triggers the creation of a new catalog item in App Portal.**	When AdminStudio publishes the application, a catalog item for that application is automatically created in App Portal, identified by the same Flexera Identifier. End user can now request this software in App Portal.
8.	**App Portal checks with FlexNet Manager Suite for license availability and reserves it if available.**	App Portal queries FlexNet Manager Suite to obtain entitlement and usage data for that application including available license count and the number of licenses used. If a license is available, App Portal will automatically reserve it for the end user.
9.	**App Portal instructs distribution system to deploy software to end user's computer.**	App Portal instructs distribution system to deploy the software to the end user's computer.

Enabling Communication Between Products

You enter the login credentials for your Flexera Service Gateway server on the **Flexera Service Gateway (FSG)** tab of the Application Manager Options dialog box.

To enable AdminStudio to communicate with additional Flexera applications via the Flexera Service Gateway, perform the following steps:

Task ***To enter Flexera Service Gateway connection settings:***

1. On the Application Catalog tab menu, select **Options**. The **Options** dialog box opens.
2. Under select **Flexera Service Gateway (FSG)**. The **Flexera Service Gateway (FSG)** tab opens.

3. Enter the following information:

Property	Description
Gateway Host Name	Enter the name or URL of your Flexera Service Gateway server. • **Port number**—If your System Administrator has installed Flexera Service Gateway using a different port than the default port, enter the appropriate port number at the end of the URL, preceded by a colon, such as: `172.300.40.501:8484` • **DNS name vs. IP address**—You can use a DNS name or an IP address. You should specify a DNS name if all clients are on the same domain and can resolve it; otherwise, use an IP address. • **HTTPS**—You should always use `https`. ***Note*** • *The Flexera Service Gateway installer is downloaded from the Flexera Product & License Center.*
User Name	Enter the FSG user name. ***Note*** • *Unless your System Administrator has provided you with a specific User Name to use, enter the default value of* **`admin`**.
Password	Enter the valid password. ***Note*** • *Unless your System Administrator has provided you with a specific User Name to use, enter the default value of* **`admin`**.

4. Click **Test Connection** to validate the Flexera Service Gateway connection information.
5. For FlexNet Manager Suite configuration, see FNMS Configuration.
6. Click **OK** to exit the dialog box.

Setting Up AdminStudio Accounts

The Flexera Integrated Solution includes AdminStudio, App Portal, FlexNet Manager Suite, the Flexera Service Gateway, Workflow Manager, and ConfigMgr (Formerly called as System Center Configuration Manager) (SCCM). All of these products communicate over a company network to provide a complete packaging and deployment solution that tracks usage and reports licensing.

When setting up these integrated Flexera products, you need to give certain accounts enhanced permissions to other products. The following table lists the required permissions for AdminStudio accounts:

Table 7-5 • AdminStudio Accounts and Privileges in the Integrated Solution

AdminStudio Account	Product/Machine Account Needs Access To	Required Privileges
AdminStudio user accounts	**Local workstation**	Require administrator privileges on the workstation where they are running AdminStudio.
	System Center Configuration Manager	Require the Application Administrator role on System Center Configuration Manager.
	FlexNet Manager Suite	See the Accounts for Integration of AdminStudio and AppPortal with FlexNet Manager Suite help topic on the Enterprise Product Integration HelpNet Site: https://docs.flexera.com/epi/
	App Portal	Need to make sure that AdminStudio users have access to the App Portal Web Site using Windows Authentication.
	SQL Server	In order to connect to an AdminStudio Application Catalog database, users require `db_datareader`, `db_datawriter`, `execute`, and `alter` permissions on the AdminStudio database. For detailed instructions on assigning these permissions, see Required Permissions on Application Catalog Databases. ***Tip*** *• The AdminStudio user account does not require these permissions if connecting to SQL Server using an SQL Server user account (which already has the appropriate permissions).*

Synchronizing Applications with App Portal and FlexNet Manager Suite

After you have entered Flexera Service Gateway credentials on the **Flexera Service Gateway (FSG)** tab of the **Options** dialog box, as described in Enabling Communication Between Products, you can then synchronize the applications in the Application Catalog with App Portal and FlexNet Manager Suite.

Task ***To synchronize applications:***

1. On the Application Catalog tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Server Options**, select **Flexera Service Gateway (FSG)**. The **Flexera Service Gateway (FSG)** tab opens.

3. Under **Synchronize Flexera Products**, click the **FlexNet Manager Platform** button to search the FlexNet Manager Suite Application Recognition Library (ARL) to locate and obtain the **Flexera Identifier** for the Application Catalog's existing applications.

   Note • *After valid Flexera Service Gateway connection information is entered, each time you import an application into the Application Catalog, the* ***Flexera Identifier*** *for that application will be automatically obtained from FlexNet Manager Suite.*

4. Click the **App Portal** button to create a catalog item in App Portal for all of the applications in the Application Catalog that were published to a distribution system that App Portal is also connected to before the Flexera Service Gateway connection information was entered.

   Note • *After valid Flexera Service Gateway connection information is entered, each time you publish a supported application to a distribution system that App Portal is also connected to, a catalog item for that application will automatically be created in App Portal.*

5. Click **OK** to exit the dialog box.

Flexera Service Gateway Messages

When AdminStudio is connected to the Flexera Service Gateway, additional output messages appear on the **Summary** panel of the Import Wizard each time you import an application into the Application Catalog or publish an application to System Center 2012 Configuration Manager:

- **Importing a package**—When you import a package into the Application Catalog, the following messages are listed:

  ```
  Extracting Flexera Identifier from FlexNet Manager Platform...
      Done with extracting Flexera Identifier from FlexNet Manager Platform
  ```

- **Publishing an application**—When you publish an application to System Center 2012 Configuration Manager, the following messages are listed:

  ```
  Sending publish notification to Flexera Gateway Service.
  Publish for Group/Application {0} has successfully notified AppPortal.
  ```

App Portal Only Integration

If you have purchased App Portal but have not purchased FlexNet Manager Suite, you can still benefit from just integrating AdminStudio with App Portal.

When both AdminStudio and App Portal are connected to the Flexera Service Gateway, whenever you publish a supported application from AdminStudio to a distribution system that App Portal is also connected to (System Center 2012 Configuration Manager, Symantec Altiris, or Casper Suite), a new catalog item will automatically be created in App Portal (in the category or categories specified on the **App Portal Information** tab of the Application View), making the application available for purchase in the App Portal storefront.

Since App Portal doesn't support desktop applications from Workspace One, the Catalog item creation and its deployment collection which is linked with the published application to Workspace One fails.

Note • *Both AdminStudio and App Portal use the same SCCM ID for the application.*

Note • *The Flexera Identifier is not used in this scenario.*

Managing an Application's Flexera Identifier

A Flexera Identifier—a unique identifier that is assigned to applications by FlexNet Manager Suite and is stored in its Application Recognition Library (ARL)—is used to maintain application identity across Flexera products.

Note • *The ARL uniquely identifies over 110,000 applications (including multiple versions and editions) from over 14,000 publishers.*

A Flexera Identifier is used to link application information from Application Manager with application information in AdminStudio Inventory and Rationalization, FlexNet Manager Suite, and App Portal. An application's Flexera Identifier is listed on the **General Information** tab of an application's **Application View**.

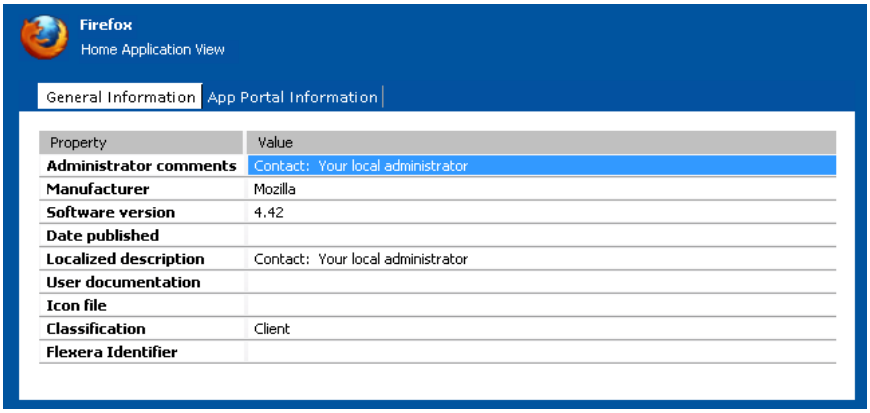


Figure 7-15: Flexera Identifier Field on the General Information Tab of Application View

When an application is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and attempts to obtain the application's Flexera Identifier. For applications that were already imported into the Application Catalog prior to connecting to the Flexera Service Gateway, you can sync applications with the Application Recognition Library, as described in Synchronizing Applications with App Portal and FlexNet Manager Suite.

However, sometimes a Flexera Identifier is not found, and you are required to either perform a manual search for an existing Flexera Identifier or create a new local entry.

- Searching an Application Catalog for Unrecognized Applications
- Performing a Manual Search for a Flexera Identifier
- Creating Local Flexera Identifier Entries for Internal or Repackaged Applications

Searching an Application Catalog for Unrecognized Applications

If both Application Manager and FlexNet Manager Suite are connected to the same Flexera Service Gateway, each time you import an application into the Application Catalog, a search for the application's Flexera Identifier is performed, and if it is found, it is listed on the **General Information** tab of the **Application View**. If the application was imported into the Application Catalog prior to connecting to the Flexera Service Gateway, you can attempt to identify its Flexera Identifier by syncing all imported applications with the Application Recognition Library, as described in Synchronizing Applications with App Portal and FlexNet Manager Suite

However, sometimes an application's Flexera Identifier is not found, such as when:

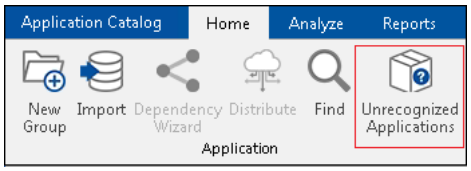
- **Incorrect information**—The value of the information in the application's Product Name, Version, Edition, or Publisher fields is either incorrect or too specific.
- **Internally developed applications**—The application has been developed internally.
- **Repackaged applications**—The application has been repackaged.

You can quickly identify all of the applications in your Application Catalog that do not have an assigned Flexera Identifier by clicking the **Unrecognized Applications** button in the toolbar of the Application Manager **Home** tab. From this **Application Search Results** list, you can search for and assign an existing Flexera Identifier to the application or create a new local Flexera Identifier.

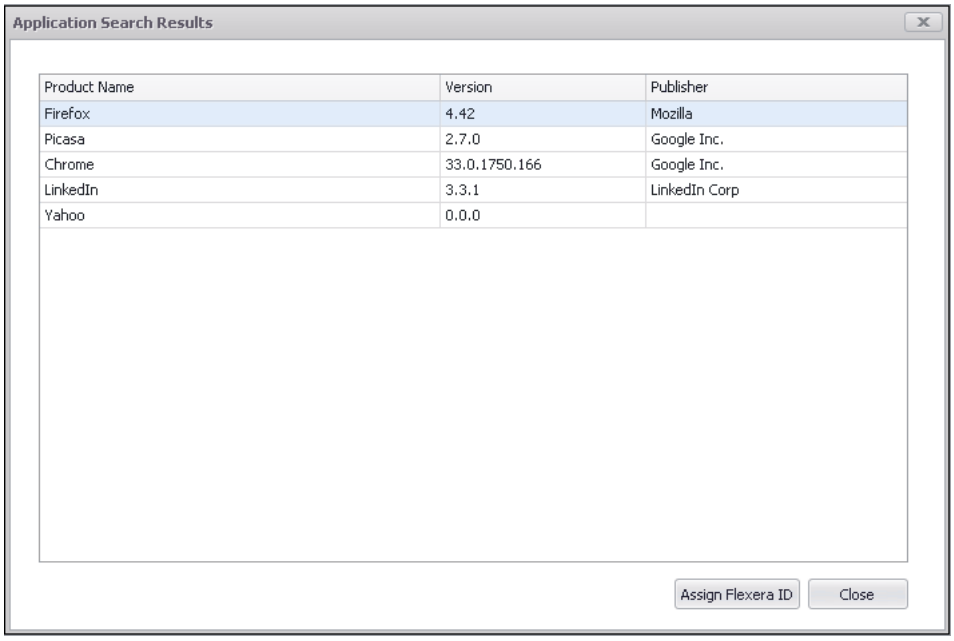
Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

Task ***To search Application Catalog for unrecognized applications:***

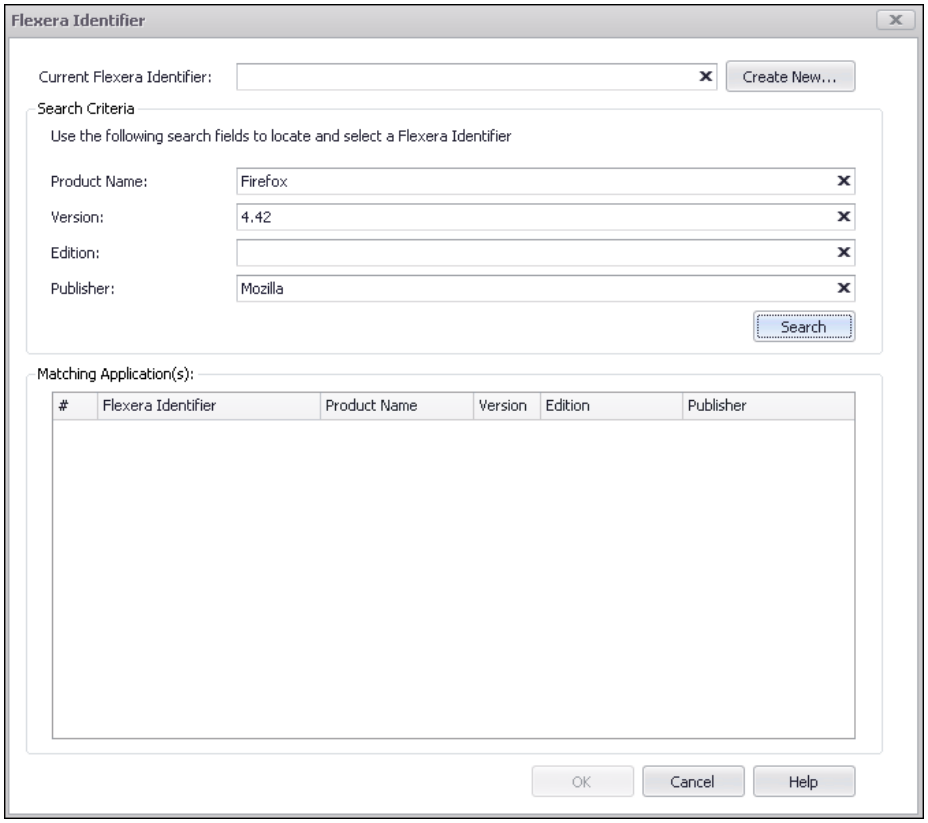
1. Click **Unrecognized Applications** in the toolbar of the Application Catalog **Home** tab.



The **Application Search Results** dialog box opens, listing all applications in the Application Catalog that do not have an associated Flexera Identifier.



2. Select an application in the list and click **Assign Flexera ID**. The **Flexera Identifier** dialog box opens.



3. Proceed with the steps in Performing a Manual Search for a Flexera Identifier, and if a Flexera Identifier is not found, proceed with the steps in Creating Local Flexera Identifier Entries for Internal or Repackaged Applications.

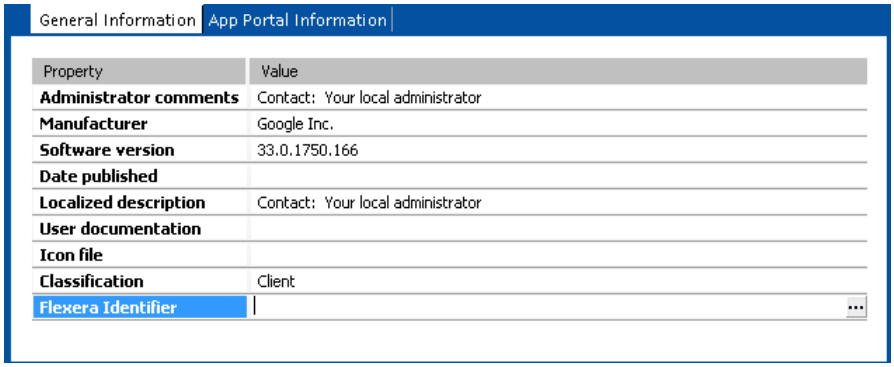
Performing a Manual Search for a Flexera Identifier

When an application is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and attempts to obtain the application's Flexera Identifier. If the application was imported into the Application Catalog prior to connecting to the Flexera Service Gateway, you can attempt to identify its Flexera Identifier by syncing all imported applications with the Application Recognition Library, as described in Synchronizing Applications with App Portal and FlexNet Manager Suite.

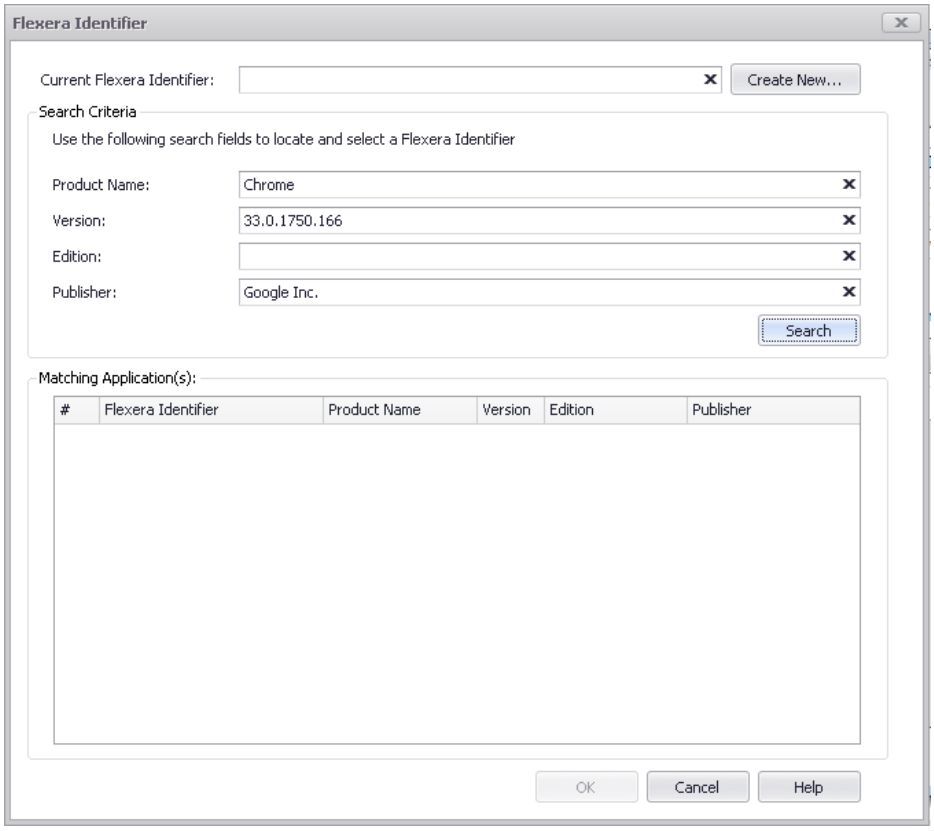
However, if an application still does not have an assigned Flexera Identifier, you can perform a manual search of the FlexNet Manager Suite Application Recognition Library.

Task ***To perform a manual search for a Flexera Identifier:***

1. Open the application's **General Information** tab of the **Application View**. The **Flexera Identifie**r field will be set to `Flexera Identifier not found`.
2. Click the browse button in the **Flexera Identifier** field.



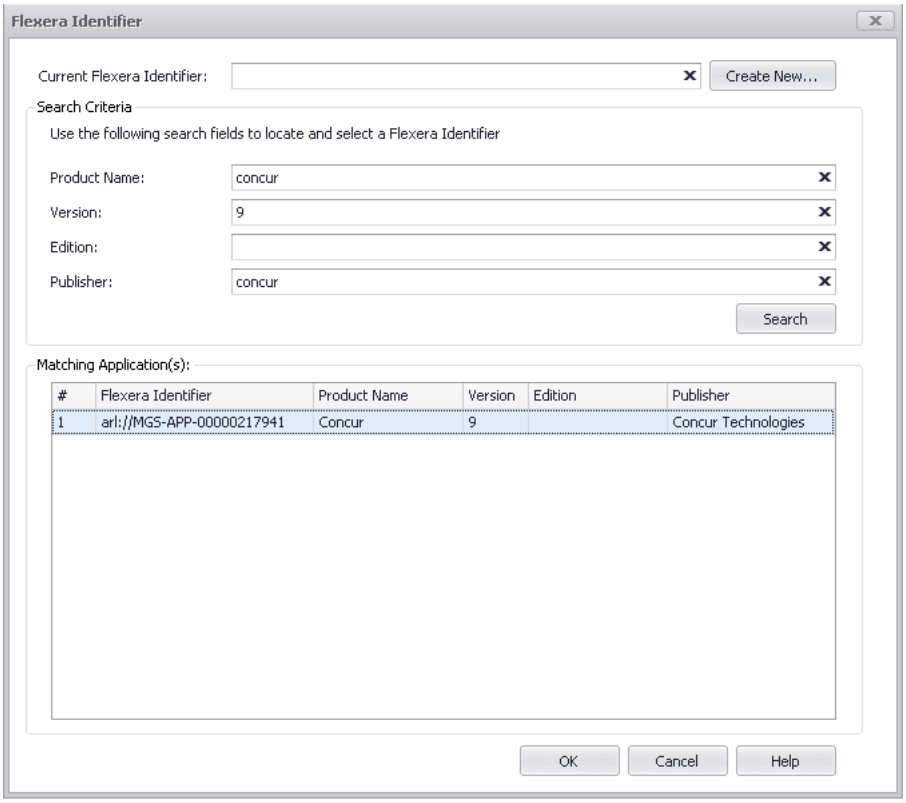
The **Flexera Identifier** dialog box opens.



3. Edit the text in the **Search Criteria** fields to either correct the information or make it less specific, and then click **Search**. A list of possible matching applications will be generated and will be listed in the **Matching Application(s)** list.

4. Do one of the following:

   - **Matching application found**—If the correct matching application is listed, select it from the list and click **OK** and then confirm that you want to associate the application with the selected Flexera Identifier. The Flexera Identifier will be saved in the Application Catalog and will be listed on the **General Information** tab of the **Application View** for that application.



- **Matching application not found**—If a matching application is not listed, continue with the steps in Creating Local Flexera Identifier Entries for Internal or Repackaged Applications to create a new local Flexera Identifier.

Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

Creating Local Flexera Identifier Entries for Internal or Repackaged Applications

Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

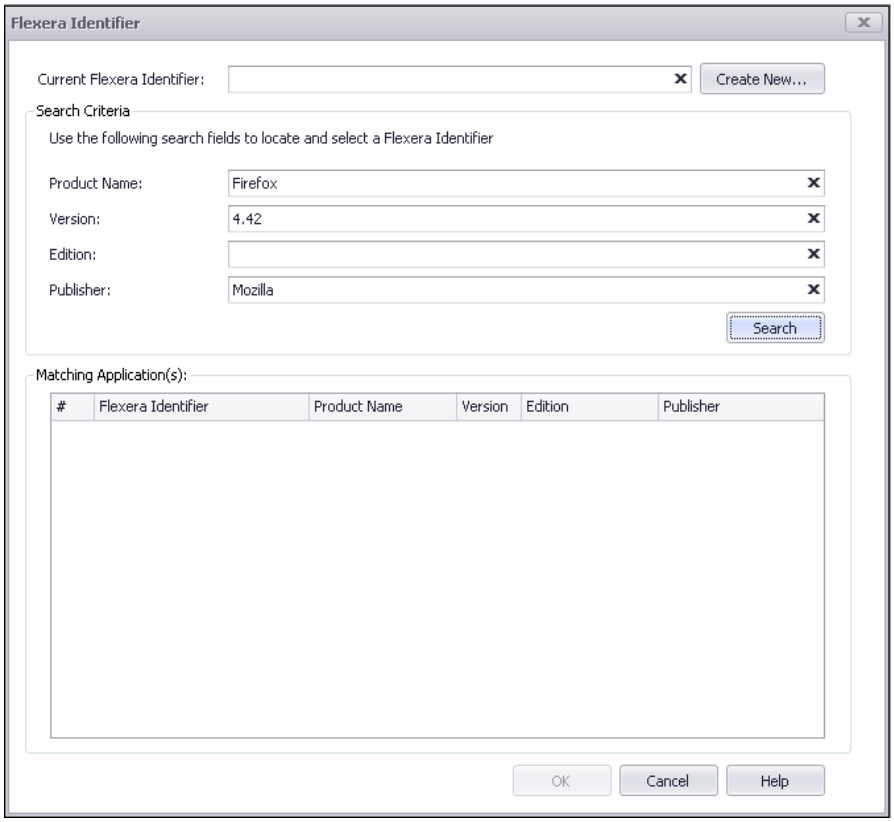
When an application is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and attempts to obtain the application's Flexera Identifier. If the application was imported into the Application Catalog prior to connecting to the Flexera Service Gateway, you can attempt to identify its Flexera Identifier by syncing all imported applications with the Application Recognition Library, as described in Synchronizing Applications with App Portal and FlexNet Manager Suite.

If an application still does not have an assigned Flexera Identifier, you can perform a manual search of the FlexNet Manager Suite Application Recognition Library, as described in Performing a Manual Search for a Flexera Identifier to attempt to identify an existing entry.

However, if you cannot locate an existing entry, you can create a new local Flexera Identifier entry for the FlexNet Manager Suite Application Recognition Library. These may be required for internally developed applications, repackaged applications, and other applications that are not recognized by FlexNet Manager Suite.

Task ***To search Application Catalog for unrecognized applications:***

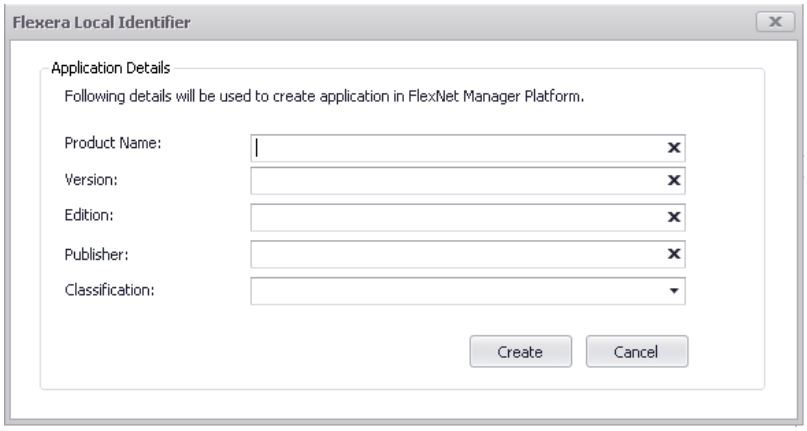
1. Open the **Application View** of the unrecognized application and click the browse button in the empty **Flexera Identifier** field. The **Flexera Identifier** dialog box opens.



2. Use the search fields to locate and select a Flexera Identifier, as described in Performing a Manual Search for a Flexera Identifier.

3. If no Flexera Identifier is found, click **Create New**. The **Flexera Local Identifier** dialog box opens.

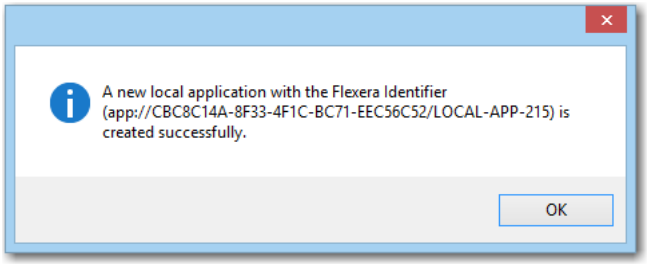
Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*



4. Enter the following information:

Property	Description
Product Name	The basic name of the application, excluding references to versions or editions, and without mentioning the publisher.
Version	The release number (or release identifier) of an application.
Edition	Enter the edition of this application.
Publisher	The name of the publisher of this software, responsible for its development and distribution.
Classification	To indicate how this application is classified, select one of the following options: • **Beta**—A pre-release application (covers such items as beta releases, alpha releases, or release candidates) that you have under some special arrangement. • **Commercial**—The application requires a license to be purchased for use in a commercial setting. • **Freeware**—Licensed for use in a commercial environment free-of-charge. • **Malware**—A potentially harmful application (a virus, Trojan, and the like), and should be treated as malware. If installations of this application are identified, you need to address the corresponding incidents or security issues. • **Shareware**—The application is available for downloading from web sites, and typically uses a “try-before-you-buy” licensing model that might include reminder messages, functional limitations, or other restrictions until a full license is purchased. • **X Rated**—The application contains potentially objectionable or sexually explicit material. You might want to consider whether corporate policies require any action. • **Update**—The application represents an update, for example, a service pack, to another application, and is issued for free to all customers regardless of purchasing agreements or support contracts (a “minor” update).

5. Click **Create**. A confirmation message appears stating that a new local Flexera Identifier has been created.



Entering Microsoft ACT Database Connection Settings

To enable AdminStudio to display data from your Microsoft ACT (Application Compatibility Toolkit) database in Analyze views and reports, you need to enter connection information for your Microsoft ACT database. You specify these settings on the **Server Options / Microsoft ACT** tab of the Application Manager **Options** dialog box.

To enable AdminStudio to communicate with your ACT database, perform the following steps.

Task ***To enter ACT database connection settings:***

1. On the Application Catalog tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Servers Options**, select **Microsoft ACT**. The **Microsoft ACT** tab opens.
3. Enter the following information:

Property	Description
Server	Enter the name of the server that contains your ACT database.
Database	Enter the name of your ACT database.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use database server login identification to log into this server. Then enter the appropriate **User Name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this database server.

Searching an Application Catalog

You can search for data in Application Catalog tables by using **Find** on the **Home** tab of the Application Manager ribbon. You can search all tables in all packages in a group, or just search one column in one table of one package.

Note • *This search is limited to string type columns.*

The tables that are searched depend upon what is selected when the **Find** dialog box is opened:

Table 7-6 • Application Catalog Search Options

If you select...	and specify these options...	this will be searched
Application Catalog	**All Tables** and **All Columns**	All tables and all columns in all of the packages in the Application Catalog.
Group	**All Tables** and **All Columns**	All tables and all columns in all of the packages in the selected group.
Package	**All Tables** and **All Columns**	All tables and all columns in the selected package.
Package	A table and **All Columns**	All columns of a specific table in the selected package.
Package	A table and a column	A specific column in a specific table in the selected package.

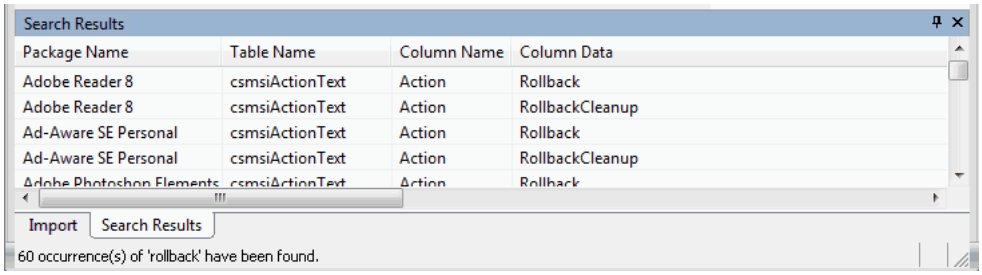
Task ***To search the Application Catalog:***

1. Open Application Manager and connect to an Application Catalog.
2. Select the node in the tree (Application Catalog, Groups, a specific group, a specific package, an OS snapshot, etc.) that you want to search.
3. On the **Home** tab of the ribbon, click **Find**. The **Find** dialog box opens.

   You could also use Ctrl + F or choose **Find** from the selected package or group's shortcut menu.
4. In the **Find What** text box, enter the text that you want to search for.

   Note • *This search is limited to string type columns.*
5. On the **Look In Table** list, select the table that you would like to search, or select **<All Tables>**. When you select a table from this list, the **Look In Columns** list is populated with all of the columns in that table.
6. If you selected a table from the **Look in Table** list, all of the columns in that table are listed. Select the column that you would like to search, or select **<All Columns>**.
7. If you want to search for a partial match rather than an exact match, select the **Partial Match** option.
   - **If this option is not selected**, Application Manager will search for an exact match of the text you entered in the **Find What** text box. The search will be case sensitive.
   - **If this option is selected**, then Application Manager will use appropriate wild card characters so that a partial data match is performed. The search will be case insensitive
8. Click **Find** to initiate the search.

   The **Find** dialog box will close, and the data that is found is displayed in the **Search Results** tab of the Output Window, in the following format:



Package Customization

Edition •

Clicking on the **Customize** button in the ribbon or **Customize** option from the context menu of an MSI package launches the **Customization Wizard**. The panels in the customization wizard are dynamic depending on the number of customization options available for the selected MSI package. Options checked/unchecked in the Customization Wizard will be saved. When a new version of the package is executed manually or during automation, all the saved customization options are used to generate a transform (.mst) file and is imported along with the MSI package into the catalog. For more information, see Customization Wizard.

Disconnecting from an Application Catalog

To disconnect from the currently open Application Manager, select **Disconnect** from Application Catalog tab menu (or from the AdminStudio **Catalog** menu).

When you have disconnected from an Application Catalog, a message appears instructing you to connect to a Microsoft SQL Server Application Catalog database.

Organizing Your Application Catalog Using Groups

Within Application Manager, you can create groups to organize your applications, patches, and OS Snapshot images in the Application Catalog. This is especially useful for organizing your Application Catalog in ways consistent with how your company is organized.

For example, you could create a group representing a certain department's base image including the proper operating system and necessary applications. When you perform conflict analysis on new packages you are integrating into your environment, you can run only the relevant comparisons—saving you the time it would take to run the analysis against all packages in the Application Catalog, or the effort of manually determining the set of packages against which you want to run the analysis each time.

Tasks relating to groups include:

- Adding Groups
- Organizing Applications in Application Manager
- Deleting Application Manager Groups
- Editing Group Properties

- Copying and Sharing Packages in the Application Catalog
- Moving Applications, OS Snapshots, and Groups

Adding Groups

To add additional groups or subgroups to an Application Catalog, perform the following steps.

Task ***To add a group to Application Manager:***

1. Open the **Home** tab of Application Manager.
2. In the tree, right-click on the group to which the new group should belong and select **New Group**.
3. Provide a name for the new group.
4. Press **Enter**.

Organizing Applications in Application Manager

To move applications or groups into different groups, perform the following steps:

Task ***To organize individual packages and groups of packages in Application Manager:***

1. Open the **Home** tab of Application Manager.
2. In the tree, select the application or group that you want to move.
3. Drag the application or group onto a new group.

Note • *The following rules apply to drag and drop operations in Application Manager:*

- *You cannot drop a node on itself.*
- *You cannot drop a node on its parent. It is already a child of the parent.*
- *You cannot drop a group on its child groups.*

Deleting Application Manager Groups

To delete a group from an Application Catalog, perform the following steps:

Task ***To delete a Application Manager group:***

1. Open the **Home** tab of Application Manager.
2. Right-click on the group you want to delete in the tree and select **Delete** from the shortcut menu.
3. From the resulting message box, confirm the deletion.

Caution • *When you delete a group, all subgroups and applications within that group are also removed from the Application Catalog.*

Editing Group Properties

For each group, you can modify its name, and add a description or other comments. To edit this information, perform the following steps.

Task ***To edit group properties:***

1. Open the **Home** tab of Application Manager.
2. Right-click on the group in the tree and select **Properties** from the shortcut menu. The **Group Properties** dialog box opens, displaying the group **Name**, **Description**, and **Comments**.
3. Make any desired edits.
4. Click **OK**.

Copying and Sharing Packages in the Application Catalog

You can choose to import multiple copies of a package into the Application Catalog or share a single package between multiple groups.

Having Multiple Copies of a Package in the Application Catalog

More than one copy of the same package can exist in an Application Catalog. To accomplish this, you can use the Import Wizard to import the same package into multiple groups. Each time you import the package, Application Manager will create a new application node in that group (if one does not already exist) to store that package, and a new entry will be made for that package in the database. When testing is performed, the package will be tested multiple times.

Sharing a Package in the Application Catalog

You can share the same package between multiple groups in the Application Catalog. An application node for that package will exist in multiple groups, but all will point to the same package; the package itself is not copied and a new entry is not made in the database. Also, when testing is performed, the package will only be tested one time.

To share a package, perform the following steps:

Task ***To share a package between groups in the Application Catalog:***

1. Open the **Home** tab of Application Manager.
2. In the first group, right-click the application node containing the package that you want to share and select **Copy** from the shortcut menu (or press Ctrl + C).
3. Right-click on a different group and select **Paste** from the shortcut menu (or press Ctrl + V). A new application node is created in the selected group for the package.

Note • *You cannot use the* ***Copy*** *function to copy a package into the same group.*

Moving Applications, OS Snapshots, and Groups

You can move OS Snapshots, applications, or groups into other groups.

Task ***To move an application, OS Snapshot, or group:***

1. Open the **Home** tab of Application Manager.
2. Select the application or group you want to move in the tree.
3. Drag the item into a new group.

Deleting Packages and Applications

You can delete both packages (deployment types) and applications from the Application Manager tree.

Deleting Packages

You can delete packages that have been imported into the Application Catalog.

Task ***To delete a package from the Application Catalog:***

1. Open the **Home** tab of Application Manager.
2. Right-click on the package in the tree, point to **Delete** on the shortcut menu, and click **Package** to delete the package from the selected application.
3. Confirm the deletion.

Note • *If you delete a package from an application that only has one deployment type, the application is also deleted. However, if the application has other associated deployment types, it is not deleted.*

Deleting Applications

If you delete an application from the Application Manager tree, you are also deleting all of that application's deployment types.

Task ***To delete an application from the Application Catalog:***

1. Open the **Home** tab of Application Manager.
2. Right-click on an application in the tree, and select **Delete** from the shortcut menu.

3. Confirm the deletion.

Note • *If you delete an application, all of that application's deployment types (packages) are also deleted.*

Browsing to Package Location from Application Manager Tree

You can quickly browse to the directory location of your source package files by right-clicking on the package in the Application Manager tree and then selecting **Open File Location** from the shortcut menu. This new option is available on both the **Home** and the **Analyze** tabs of Application Manager.

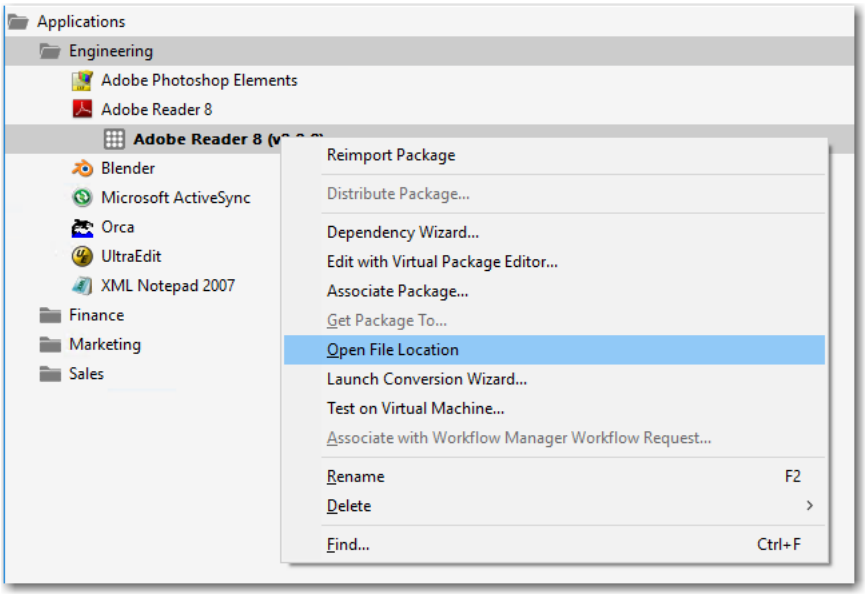


Figure 8: New "Open File Location" Command on Shortcut Menu

A new File Explorer window opens, listing the package's source files.

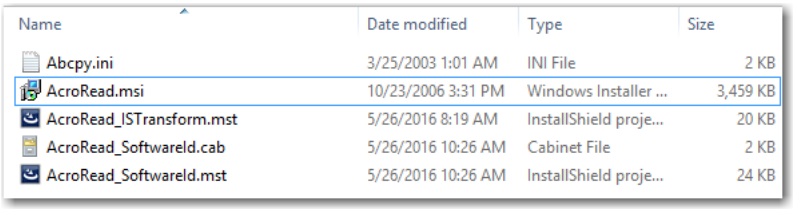


Figure 9: File Location of Imported Package's Source Files

Importing

You can use the Import Wizard to import multiple application deployment types either one at a time, all of the packages in a directory, or one or multiple packages directly from a deployment system.

- Package Types Supported By the Import Wizard
- Package Sources Supported by the Import Wizard
- Importing an Application Using The Package Feed Module
- Importing a Single Package File
- Importing Links to Public Store Applications
- Importing a Folder of Multiple Applications
- Importing From ConfigMgr (Formerly called as System Center Configuration Manager)
- Importing Merge Modules
- Importing OS Snapshots
- Using Duplicate Package Identifiers
- Generating Software ID Tag Files During Package Import
- Viewing Bundled Packages of Complex Installer Executables

Package Types Supported By the Import Wizard

You can use the Import Wizard to import the following package types into the Application Catalog:

Table 7-7 • Data Types Supported By the Import Wizard

Data Type	Description
Windows Installer Packages, Transforms, and Patches	You can import Windows Installer packages (`.msi`) with any associated transforms (`.mst`) and patches (`.msp`) into the Application Catalog. You have several available options when importing Windows Installer packages: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager)
Microsoft UWP App Packages (.appx)	You can import Microsoft UWP app packages (`.appx`) into the Application Catalog. UWP is the packaging format used to distribute and install apps on Windows 8.x and 10, and is the only format allowed for Universal Windows Platform (UWP) apps. You have several available options when importing Microsoft UWP app packages: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) ***Note*** • *You can also upload Windows 8 apps (`.appx`) using these methods.*

Table 7-7 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description
Microsoft MSIX Package (.msix)	You can import Microsoft MSIX packages (`.msix`) into the Application Catalog.
Virtual Packages: Microsoft App-V, Citrix XenApp, VMware ThinApp	You can import virtual packages in Microsoft App-V (4.x and 5.0), Citrix XenApp, VMware ThinApp (4.x and 5.0). You have several available options when importing virtual packages: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) Virtual packages are associated with their source Windows Installer package by matching Package Codes. Virtual packages that were created by AdminStudio include a `metadata.ami` file that identifies the Package Code of the virtual package's source Windows Installer package. To manually associate a virtual package with its source Windows Installer package, you can use the **Associate Package** function, as described in Associating a Virtual Package with its Source Windows Installer Package.
PowerShell Wrapped Package Files (.ps1)	You can import PowerShell wrapped package (`.ps1`) files into the Application Catalog. When a PowerShell script is imported into AdminStudio, the underlying package is also imported into the Application Catalog and can be tested. You have several available options when importing PowerShell wrapped package files: • Importing a Single Package File • Importing a Folder of Multiple Applications

macOS Desktop Applications	You can import the following macOS desktop applications into the Application Catalog, which will enable you prepare those applications for deployment, and then deploy them to JAMF Casper Suite.
	• **Local file**—The following types of macOS desktop applications can be imported:
	  • **Apple disk image package (.dmg)**—When you double-click a `.dmg` file, an Apple disk image is "mounted" as a volume within the Finder. To install the application, you usually drag the application icon from the disk image into the Applications folder.
	  • **Apple installer package (.pkg)**—Double-clicking a `.pkg` file launches the Apple installer application, where the package is installed by proceeding through an installation wizard.
	• **Public store link**—Link to macOS desktop application in the Apple Mac App Store.
	For more information, see:
	• Importing a Single Package File
	• Importing a Folder of Multiple Applications
	• Importing Links to Public Store Applications

Table 7-7 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description
Legacy Applications	You can import non-MSI legacy setup types (such as InstallShield Professional or ISMP installations) into the Application Catalog using any of the following methods: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) When you import a legacy installer (`.exe`), you are prompted to select a setup configuration file (`.ini`) to include with the imported package. For more information, see About Legacy Installer Packages.

Table 7-7 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description
Complex Installer Executables	You can import complex installer executable files that contain bundled Windows Installer packages, including: • InstallShield InstallScript `.exe` files • InstallShield Basic MSI installers that are compressed into a `setup.exe` file • InstallShield Suite Installer `.exe` files • Wise Package Studio `.exe` files • Other executable file types that can be uncompressed by 7-ZIP • InnoSetup • Nullsoft • WiseScript • WiX Burn • Advanced Installer • InstallAnywhere ***Note*** *• If AdminStudio is unable to identify the* ***.exe*** *type of an imported package, by default it is categorized as a legacy installer.* You can import these executables using any of the following methods: • Importing a Single Package File • Importing a Folder of Multiple Applications • Importing From ConfigMgr (Formerly called as System Center Configuration Manager) When you import one of the complex installer executables, you can view the names of bundled Windows Installer packages and perform operating system compatibility, application virtualization compatibility, and best practices testing on those bundled packages. For more information see: • Viewing Bundled Packages of Complex Installer Executables • Viewing Combined Test Results of Child Windows Installer Packages of Complex Installer Executables

Table 7-7 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description

Mobile Apps	You can import the following mobile apps into the Application Catalog, which will enable you prepare those applications for deployment, and then deploy them to System Center 2012 Configuration Manager or Workspace ONE.
	• **Apple iOS mobile app**
	  • **Local file**—Mobile app file (`.ipa`).
	  • **Public store link**—Link to mobile app in the Apple App Store.
	• **Google Android mobile app**
	  • **Local file**—Mobile app file (`.apk`).
	  • **Public store link**—Link to mobile app in the Google Play Store.
	• **Microsoft Windows mobile apps**
	  • **Local file**—UWP app package (.appx) or Windows 8 package (.appx).
	  • **Public store link**—Link to mobile app in the Microsoft Windows Store.
	You have several available options when importing mobile apps:
	• Importing a Single Package File
	• Importing Links to Public Store Applications
	• Importing a Folder of Multiple Applications
	• Importing From ConfigMgr (Formerly called as System Center Configuration Manager)
	
	Edition • *Support for mobile apps is included when you purchase AdminStudio Professional or Enterprise Edition with Mobile.*
	
	Note • *To import a link to a mobile app in a public store, see Importing Links to Public Store Applications.*

Merge Modules	A merge module (`.msm`) is a package containing all of the logic and files needed to install distinct pieces of application functionality such as run-time `.dll` files and virtual machines. For optimal performance, Merge modules should be imported into an Application Catalog database prior to importing Windows Installer packages. This ensures that conflicts resulting from not using available merge modules are correctly identified. For more information, see Importing Merge Modules.
OS Snapshots	You can import OS Snapshot (`.osc`) files into the Application Catalog to use to determine conflicts between an operating system and a package. See Importing OS Snapshots and About Legacy Installer Packages.

Note • *In previous releases, if you were connected to a Software Repository-enabled Application Catalog, it was possible to use the Import Wizard to perform an ad-hoc import of a transform or patch file (importing a transform or patch file after their associated Windows Installer package had already been imported). Beginning with AdminStudio 11.5, this option is no longer available. Transform and patch files always have to be imported along with their associated Windows Installer package.*

Package Sources Supported by the Import Wizard

You can import packages or a directory of packages from your local network. You can also import packages from System Center 2012 Configuration Manager or System Center 2007 Configuration Manager.

You specify the source of the package(s) that you want to import by making a selection on the **Source** panel of the Import Wizard:

Table 7-8 • Package Sources Supported by the Import Wizard

Source	Description
Download & Import using Package Feed Module	Select to download and import a single application into the Application catalog by using the Package Feed Module. See Importing an Application Using The Package Feed Module
Single application	Select to import a single application into the Application Catalog. See Importing a Single Package File.
Folder of multiple applications	Select to import a directory of packages (containing multiple deployment types, if desired) into the Application Catalog. See Importing a Folder of Multiple Applications.
Link to a public store app	Select to import a link to an Apple iOS mobile app in the Apple App Store, a Google Android mobile app in the Google Play Store, or a Windows Store mobile app in the Windows Store. See Importing Links to Public Store Applications.
Applications from a deployment system	Select this option to import applications or packages from a ConfigMgr (Formerly called as System Center Configuration Manager) server. See Importing From ConfigMgr (Formerly called as System Center Configuration Manager).

Importing an Application Using The Package Feed Module

Edition •

The AdminStudio Package Feed Module can save countless hours by leveraging a curated setup database covering thousands of products. This helps you to more quickly add new applications to the AdminStudio application catalog. From here you can generate reports, wrap it with a script template you have created, and pass it off to any of the several popular deployment solutions to which integration is provided.

- Network Requirements for Package Feed Module
- About the Package Feed Module
- Package Feed Module Authentication
- Ensuring the Security of the Package Feed Module
- Using the Package Feed Module to Perform Bulk Import

About the Package Feed Module

The **Package Feed Module** helps you to do the following:

- Download and import the latest version of a product
- Identifying the silent installation parameters for the particular version of the product

Note • *The installation parameter available in the* ***Silent Command Line Arguments*** *panel of the* ***Package Feed Module*** *is validated*

The **Package Feed Module** supports the below package types:

- EXE
- MSI
- PKG
- DMG
- MSU
- MSIX

To learn more about the **Package Feed Module**, click here.

Important • *Note the following to activate the Package Feed Module:*

- *To avail complete use of the Package Feed Module, please contact your sales representative or contact us online at: https://www.flexera.com/about-us/contact-us.html*

Package Feed Module Authentication

All of the packages in Package Feed Module are hosted on Flexera's secured cloud infrastructure (`https://dl.sci7.secunia.com`). AdminStudio authenticates itself against this URL through a token-based system, so there is no scope for impersonation. It does not matter if AdminStudio (and Package Feed Module) is installed on a VM hosted on Azure or on a physical machine, as long as it has access to this URL and the token for authentication it will be able to download packages. The token and URL are embedded within AdminStudio.

Ensuring the Security of the Package Feed Module

Practices are in place to ensure the security of Flexera's cloud infrastructure site to prevent malicious content.

Packages in the Package Feed Module are downloaded directly from the software vendor's download links. These packages are then put through anti-virus checks and MD5 Checksum and SHA validations to make sure there is no malicious content injected during download. These checks make sure that all of the packages are fully safe and secured before they are hosted on Flexera's secured cloud infrastructure site.

Beyond these security checks, AdminStudio runs QA automation tests on the packages to test the install command line switches by installing the packages and validating successful installation.

Using the Package Feed Module to Perform Bulk Import

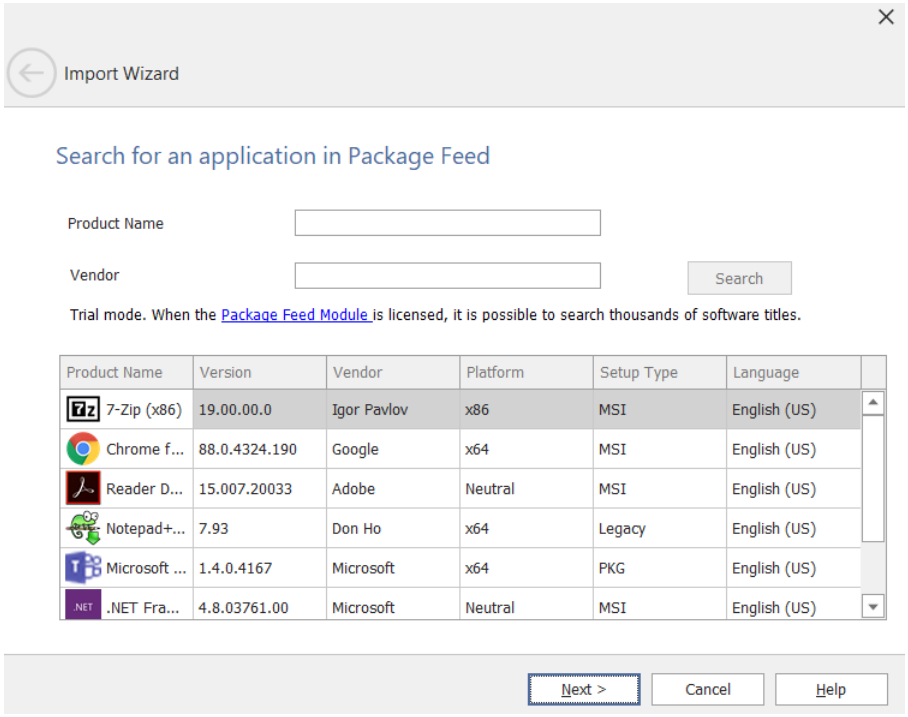
To use the Package Feed Module to perform bulk import, perform the following steps.

Task ***To download and import an application into the Application Catalog using the Package Feed Module:***

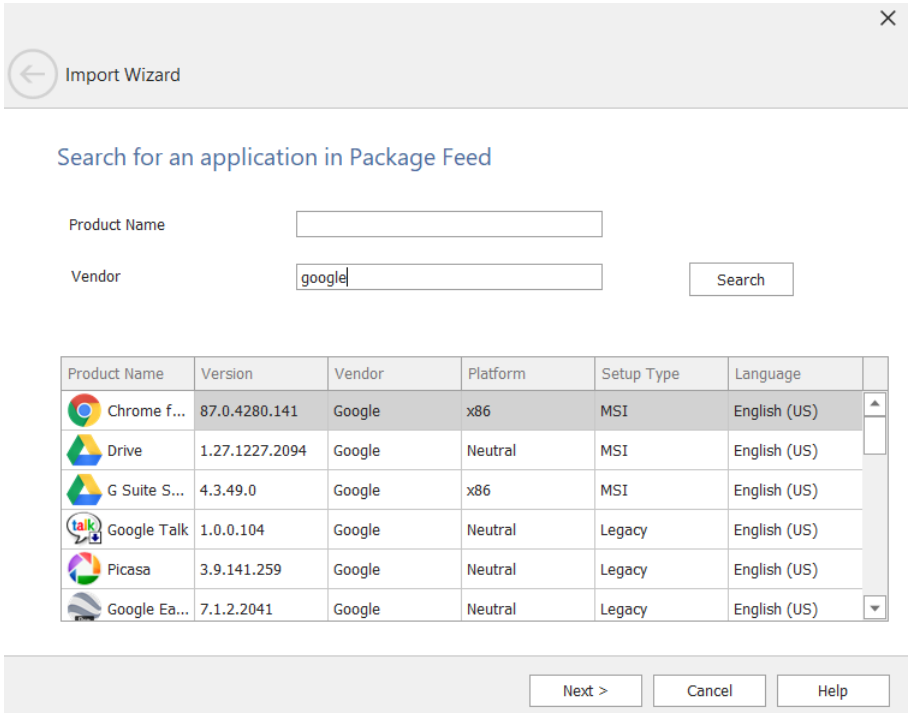
1. On the **Home** tab of the Application Manager ribbon, click the **Import** button. The **Source** panel opens.
2. Select **Import using Package Feed module** and click **Next**. The **Search for an application in Package Feed** panel opens.

Tip • *You can directly select the **Package Feed** button on the **Home tab** of the Application Manager ribbon. The **Search for an application in Package Feed** panel opens.*

When the Package Feed Module is in evaluation mode (AdminStudio evaluation activation, Enterprise/Professional License), then only evaluation packages will be listed in the Package Feed Search Grid.

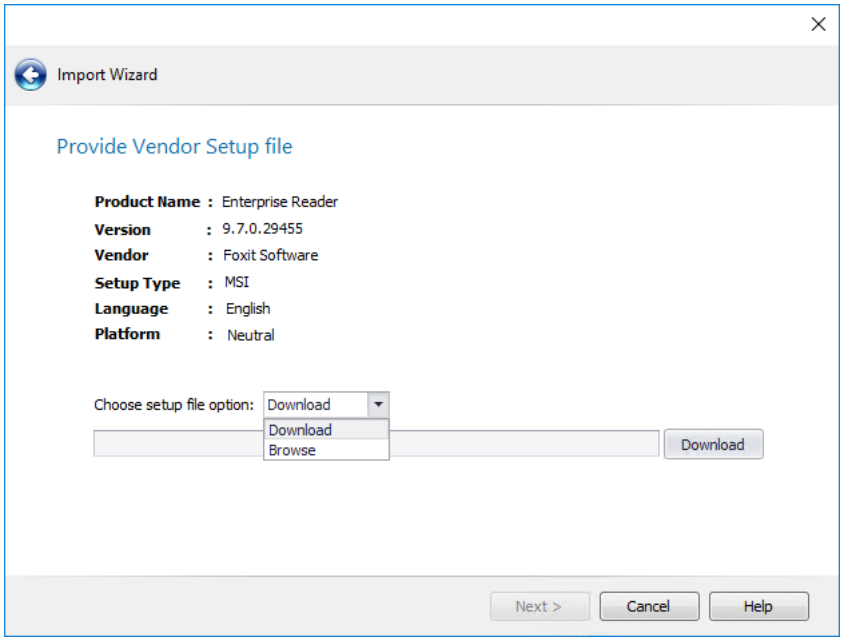


3. When package feed module is in activation mode, you can search for the packages in the search grid.

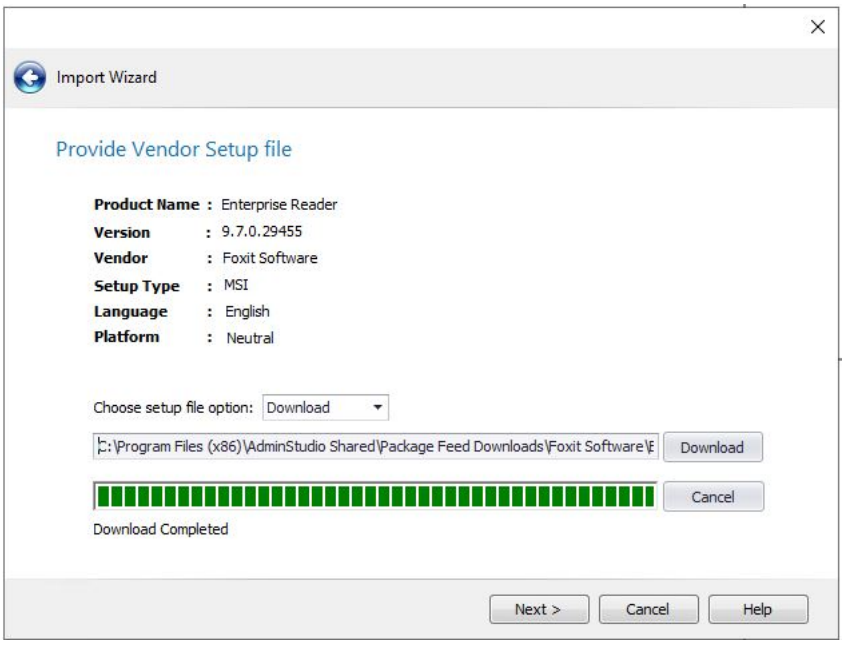


Note • *During the search for an application, do not use asterisk (*) along with the name; for example, Adobe**.*

4. Click **Next**. The **Provide Vendor Setup file** panel loads.

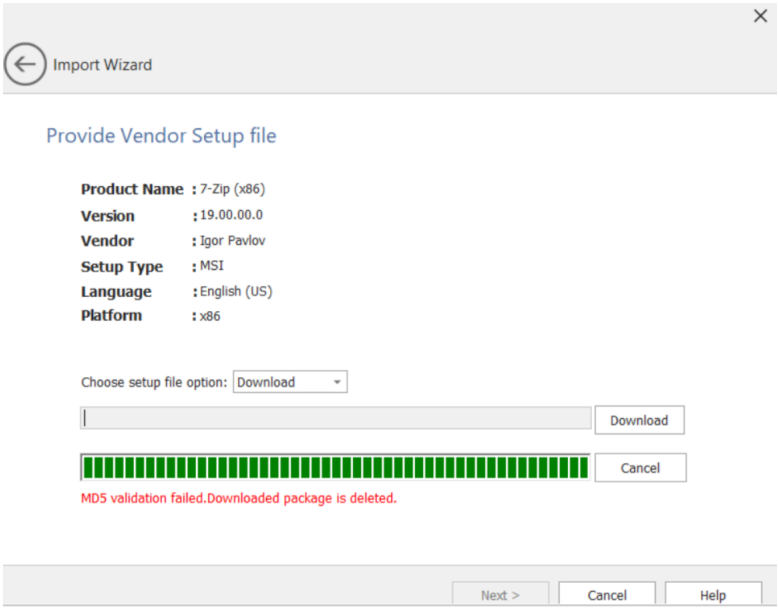


5. Click **Download** to download the setup file or you can **Browse** to selected the setup file.

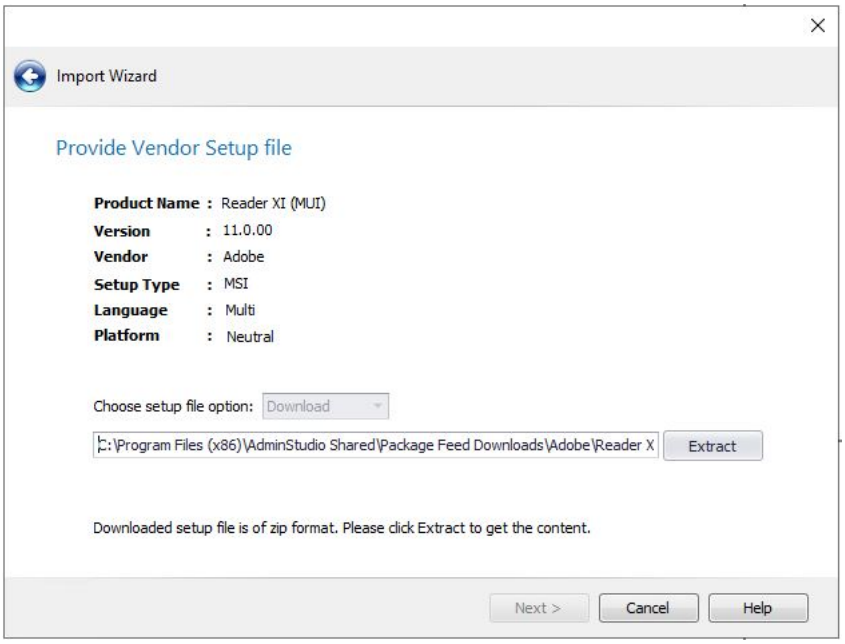


Note • *Package downloaded from Package Feed Module will get validated against the MD5 checksum.*

- *If MD5 value is matching, then the downloaded package will be retained in the configured Package Feed download directory.*
- *If MD5 value is not matching, then downloaded package will get deleted from the configured Package Feed download directory.*



6. If the downloaded file is of **zip** format, click the **Extract** button, and then browse and open the extracted setup file, and click **Next.**



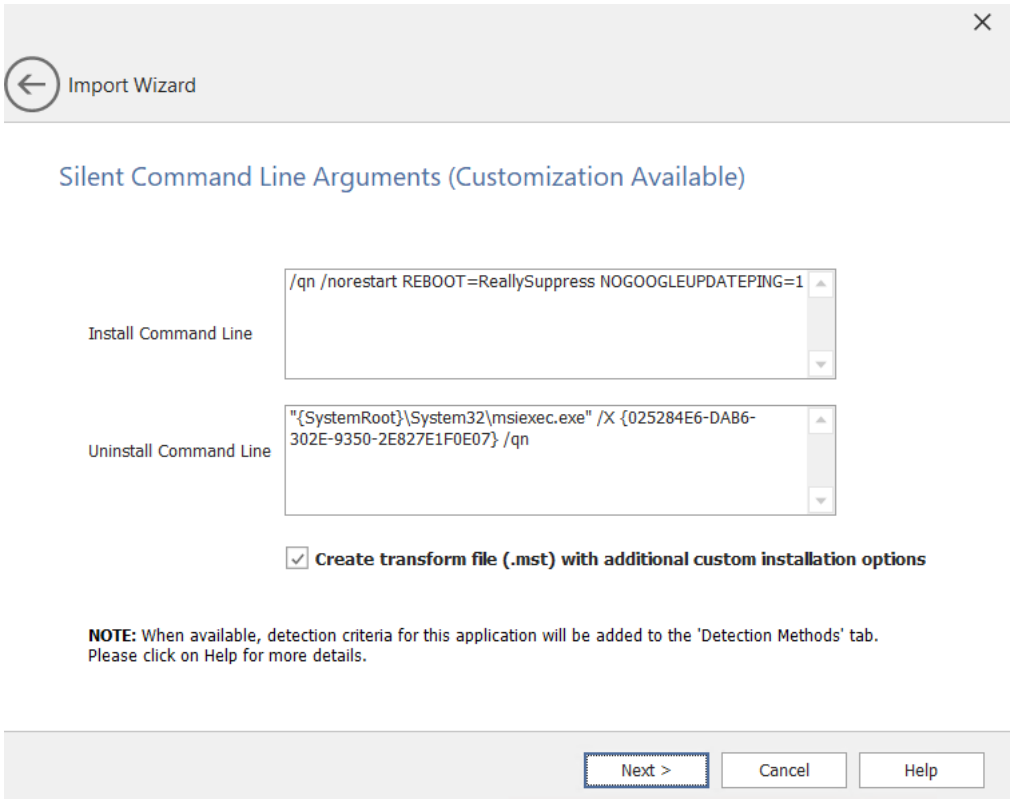
Note • *If you don't* ***Browse*** *and click* ***Next****, an error message will appear.*

7. When you have finished downloading the setup file to the package feed configured download path, click **Next**. The **Silent Command Line Arguments panel** loads.

Note • *To define your desired download folder path see Package Feed Options Tab*

8. In the **Silent Command Line Arguments panel**, Install Command Line and Uninstall Command Line fetched from the package feed module of the package. If package can be customized then **Create transform file mst file with additional custom installation options** will be enabled and checked by default if deployment type is msi.

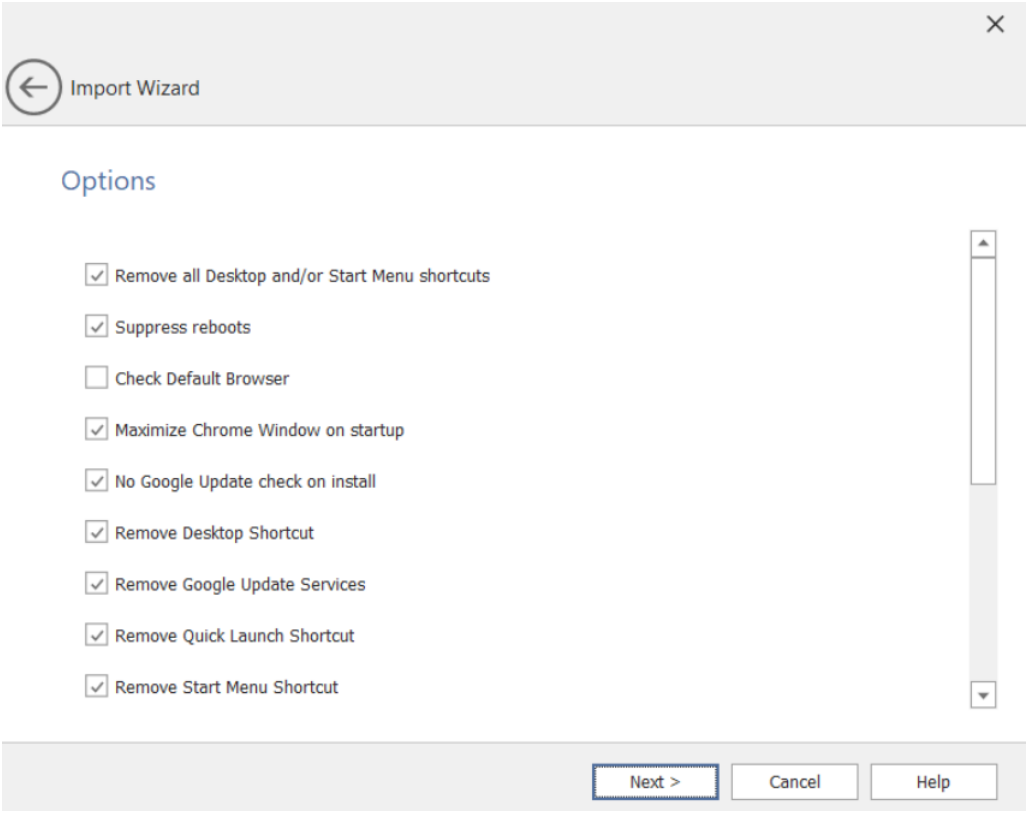
Note • *Customization option is available for only MSI packages.*



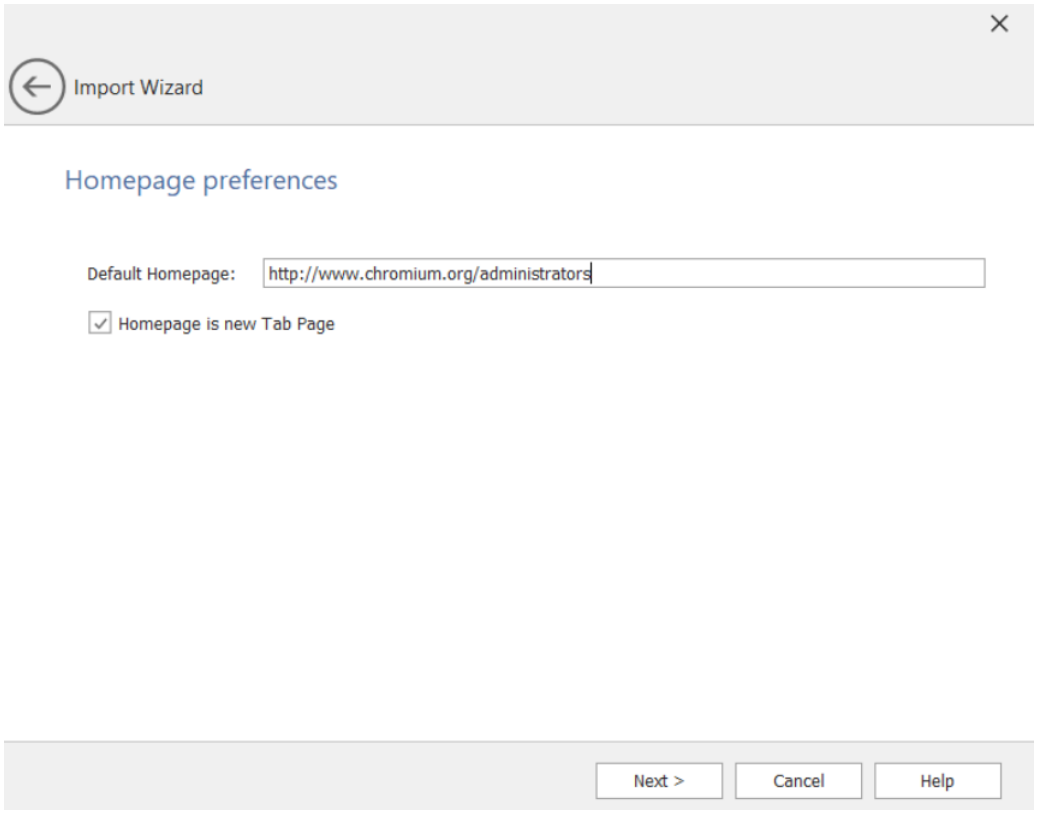
If customization options are not available, then check box will be disabled and unchecked.

Note • *If the customization is available, then the wizard will appear as* ***Silent Command Line Arguments (Customization Available)****.*

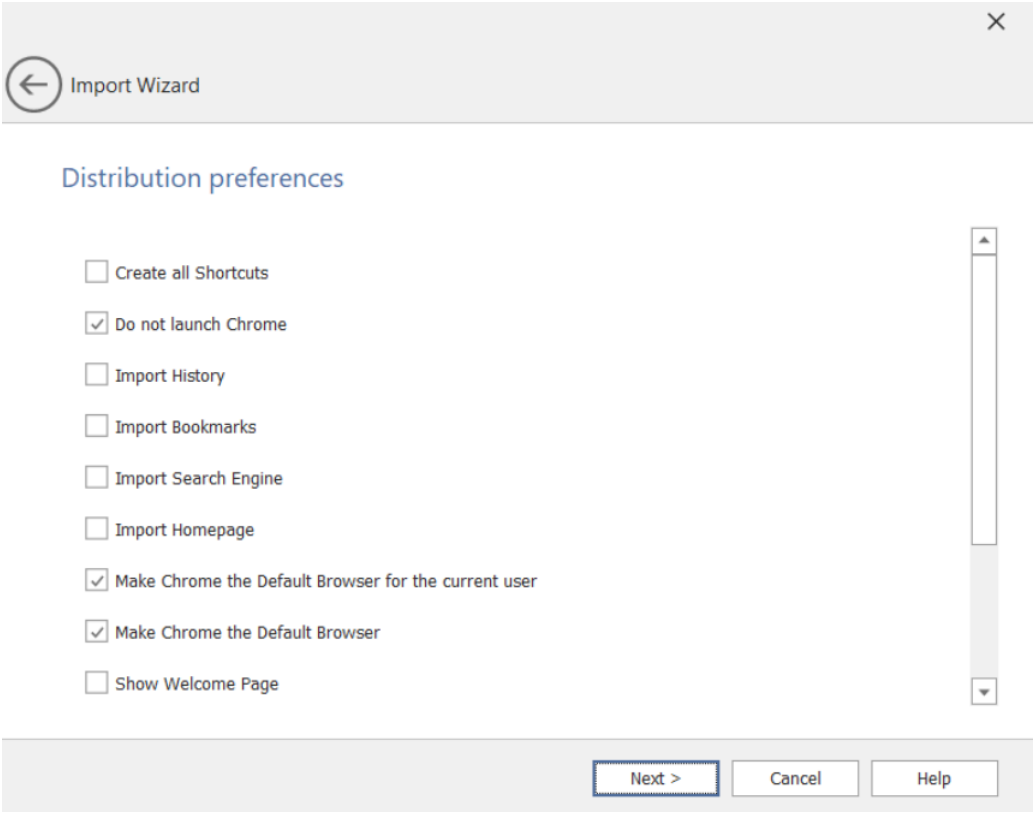
9. Click **Next**. When using the Package Feed Module, dynamic custom panels will appear in the Import Wizard based on the settings defined for each package. For example, when importing the Google Chrome package, several dynamic custom panel will appear, including the **Options** panel.



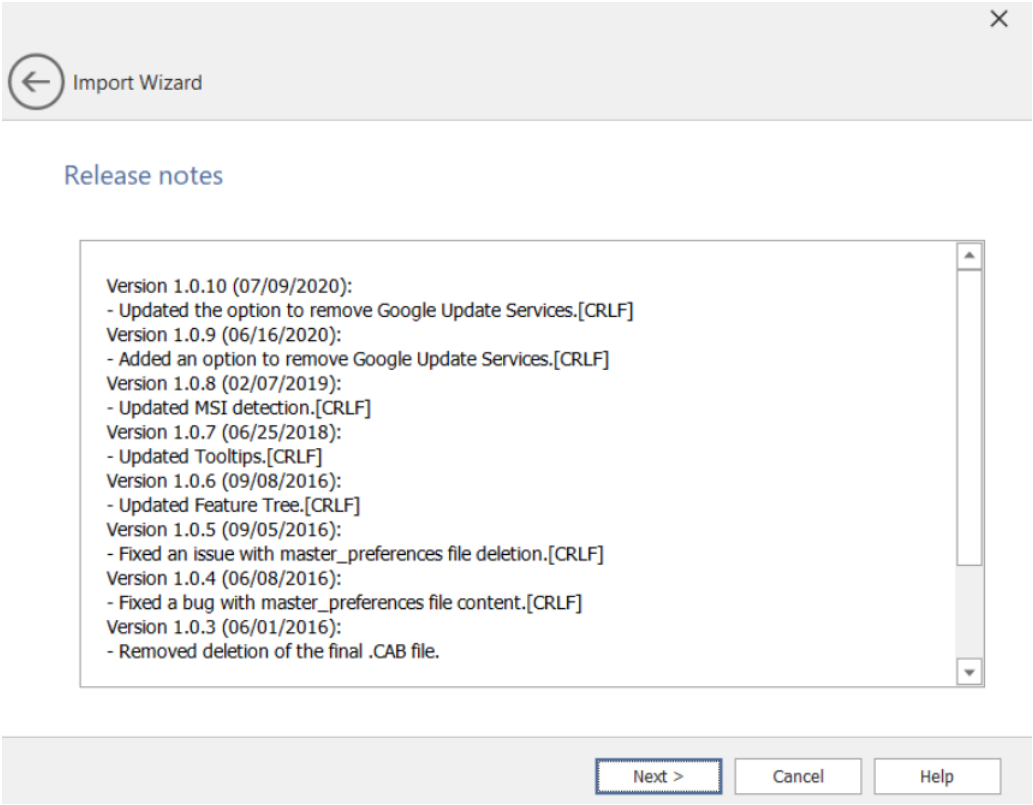
10. Edit the settings in the **Options** panel and click **Next**. The **Home Page Preferences** panel opens (for the Google Chrome package).



11. In the **Default Homepage** text box, you can add or edit the default home page. If you want to open the default home page when you create a new tab, then select **Homepage is new Tab Page** check box button.
12. Click **Next**. The **Distribution preferences** panel loads (for the Google Chrome package).



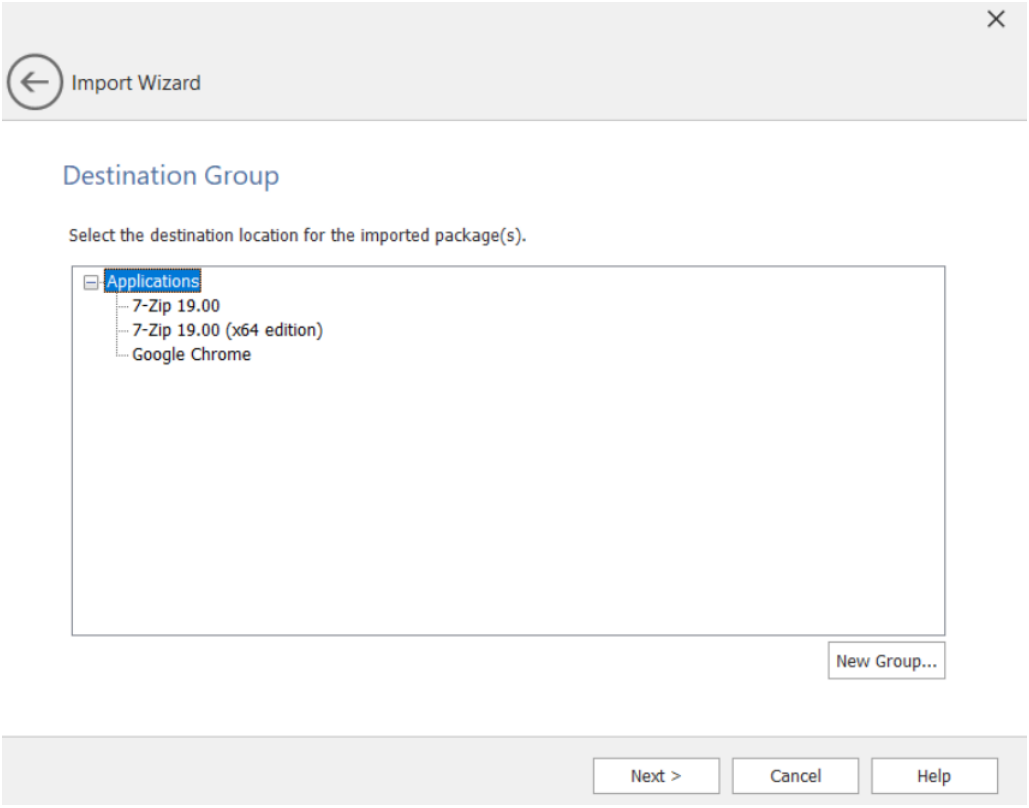
13. Select the desired preferences by clicking on the check box button.
14. Click **Next**. The **Release notes** panel loads (for the Google Chrome package), which displays release updates of the product.



15. Click **Next**. The **Destination Group** panel opens.
16. Select a destination group into which your package will be imported.

Note • *If you launched the Import Wizard by selecting a group in the tree and then clicking **Import**, that group will be selected by default on the **Destination Group** panel*

Note • *If you want to import the package into a new group, click the **New Group** button to create a new group.*



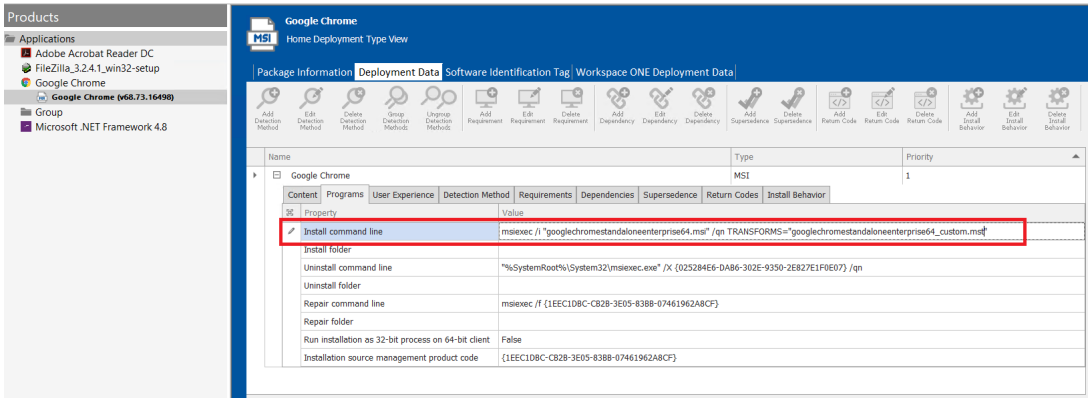
17. Click **Next**. The **Summary** panel opens.

18. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Next** to start the import.

    Progress messages are displayed. Depending on whether options have been set on the **Import Options / General** tab of the Application Manager **Options** dialog box (available from the Application Catalog tab menu), testing may be performed during import.

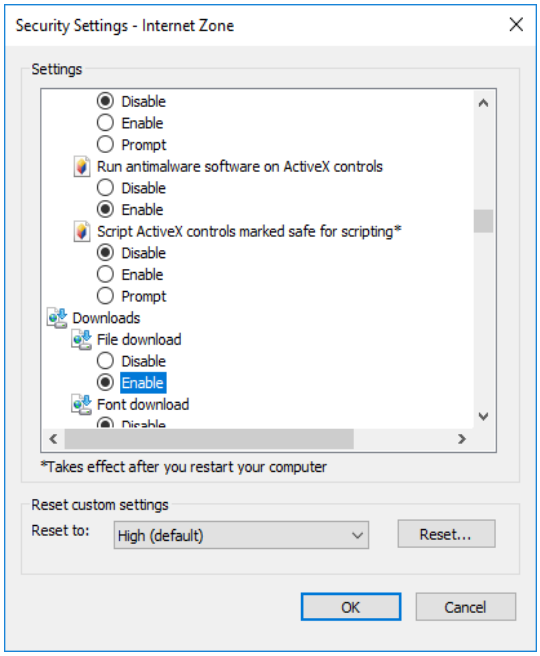
    Command line value gets updated for _custom.mst transform file.

    The package will then appear under an **Application** node in the Application Catalog tree:



Note • *On successful import of an application into the Application Catalog Tree, you can see the* ***Install Command line*** *and* ***Uninstall Command Line*** *for the selected application in* ***Deployment Data >> Programs >> Install Command Line and Uninstall Command Line****.*

Tip • *Make sure download option is enabled in your IE browser. Navigate to* ***Tools (Alt+X) >> Internet options >> Security*** *tab* ***>> Custom level... >> Downloads >>*** *select* ***Enable***



Network Requirements for Package Feed Module

To download packages from the package feed, your system should meet the following network requirements:

Table 7-9 • Network Requirements for Package Feed Module

Item	Description
Internet connection	You should have an active internet connection to work on the Package Feed Module.
Whitelist URL	The following URL is used to download packages from the package feed. `https://dl.csi7.secunia.com`
Protocols and port numbers	TCP Port 443

Importing a Single Package File

You can import the following types of package files:

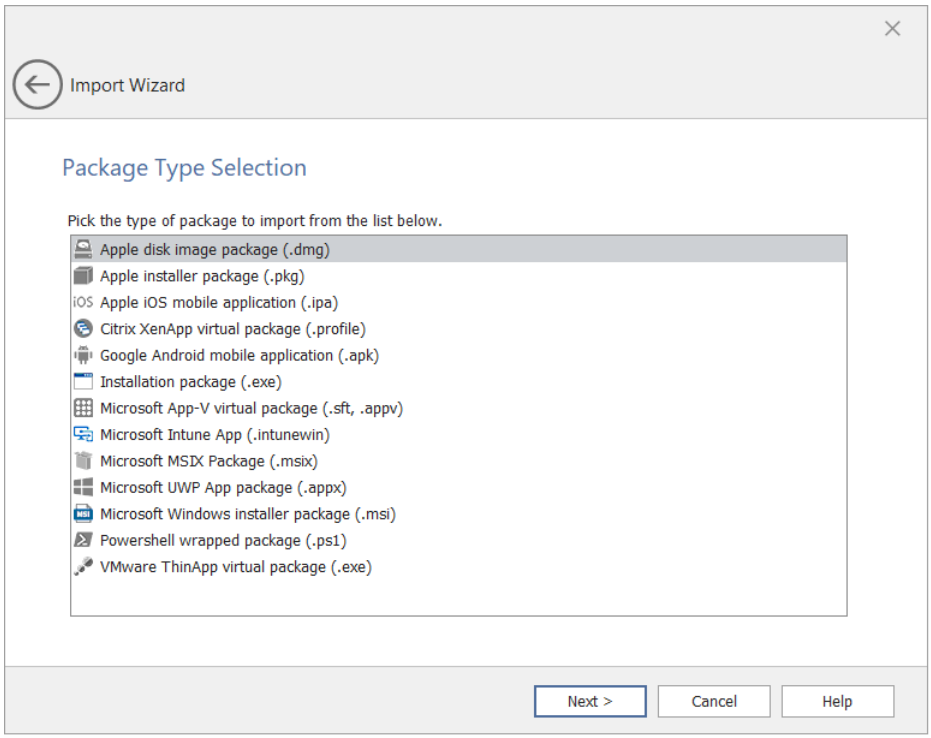
- **Windows Installer package**—You can import a Windows Installer package with all of its associated transform files and patches into the Application Catalog at the same time.
- **Virtual package**—You can import a single virtual package in Microsoft App-V, Citrix XenApp, VMware ThinApp.

  ***Note** • AdminStudio supports the import of App-V 4.x packages (.sft files) as well as App-V 5.0 packages (.appv files).*
- **macOS App**—You can import a single macOS desktop application in `.pkg` or `.dmg` format.
- **Mobile app**—You can import a single mobile app in Apple iOS, Google Android, or Windows Store format.
- **Microsoft Intune**—you can import a Microsoft Intune package into the Application Catalog.

To import a single Windows Installer package, virtual package, or mobile app, perform the following steps:

Task ***To import a single package into an Application Catalog:***

1. On the **Home** tab of the Application Manager ribbon, click the **Import** button. The **Source** panel opens.
2. Select **Single application** and click **Next**. The **Package Type Selection** panel opens.



3. Select the package type of the package that you want to import, and click **Next**. The **Package File Selection** panel opens.
4. Click **Browse** and select the package that you want to import.

5. Click **Next**. For some package types, the **Package Support Files** panel opens, where you may optionally select any additional files to be imported along with the package, such as:

Package Type	Support File	Description
Windows Installer (.msi)	Transform files (.mst)	All of the `.mst` files that are in the same directory as the Windows Installer file you are importing are automatically listed, but only those `.mst` files that AdminStudio determines are probably applicable to this Windows Installer package are selected to be included in the import. If you do not want to import a selected `.mst` file, clear the selection. ***Note*** • *You an add additional transform files and specify the order that they will be applied, as described in Adding Additional Package Support Files and Ordering List.*
	Patch files (.msp)	If a patch file is in the same directory as the Windows Installer file you are importing, that patch file will automatically be listed. If you do not want to import it, clear the selection. ***Note*** • *You an add additional patch files and specify the order that they will be applied, as described in Adding Additional Package Support Files and Ordering List.* ***Note*** • *If you specify an update.exe patch file that was created by Developer/DevStudio/InstallShield Editor, Application Manager will extract the .msp file in the Temp folder and then perform the import.* ***Note*** • *See About the Administrative Installation of Patches.*
Legacy packages (.exe)	Setup configuration files (.ini)	Contains setup and configuration information for a legacy installation package.

6. If the **Package Support Files** panel opens, do the following to add and select this package's support files, and to modify the order in which they are applied:

   - **Selecting a support file**—If a support file is already listed, select it to include it in the import.
   - **Adding support file**—To add an additional support file, click the **Add** button and browse to the location of the support file. If the package requires multiple support files, you can repeat the procedure as necessary.
   - **Reordering support files**—The order in which the support files are applied may be important, and can be changed by selecting a support file in the list and clicking the **Up** and **Down** buttons.

Note • *More than one **Package Support Files** panels may open. For Windows Installer packages, you first see a **Package Support Files** panel prompting the import of transform files, and next a **Package Support Files** panel that prompts the import of patch files opens.*

Note • *If you specify an `update.exe` patch file that was created by Developer/DevStudio/InstallShield Editor, Application Manager will extract the `.msp` file in the `Temp` folder and then perform the import.*

7. When you have finished adding files to the **Package Support Files** panel, click **Next**. The **Destination Group** panel opens.
8. Select a destination group into which your package will be imported.

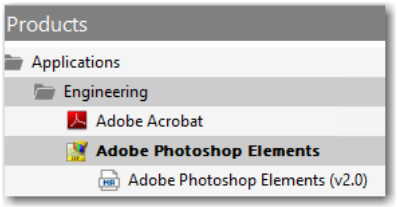
Note • *If you launched the Import Wizard by selecting a group in the tree and then clicking **Import**, that group will be selected by default on the **Destination Group** panel.*

Note • *If you want to import the package into a new group, click the **New Group** button to create a new group.*

9. Click **Next**. The **Summary** panel opens.
10. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Next** to start the import.

    Progress messages are displayed. Depending on whether options have been set on the **Import Options / General** tab of the Application Manager **Options** dialog box (available from the Application Catalog tab menu), testing may be performed during import.

    The package will then appear under an **Application** node in the Application Catalog tree:



Creation of Application Nodes During Package Single Package Import

Application nodes are created in the Application Manager tree using the package's associated Package Code. If multiple packages of different deployment types (such as Windows Installer, App-V, and ThinApp) of the same software product are all imported into the same Group and all have the same Package Code, all of the deployment types will be automatically listed under the same application node.

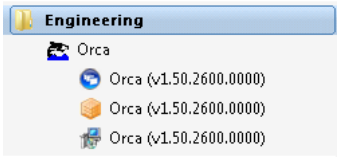


Figure 7-1: Three Deployment Types Under One Application Node

However, if multiple packages of different deployment types of the same software product have different Package Codes, and are all imported into the same Group, an additional node for that application will be created for each Package Code, incremented by a number, such as: `ABC Application` and `ABC Application [1]`.

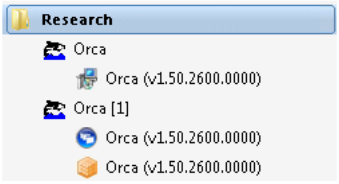


Figure 7-2: Three Deployment Types Under Two Different Application Nodes

Note • *Virtual packages that were created by AdminStudio include a `metadata.ami` file that identifies the Package Code of the virtual package's source Windows Installer package.*

Forcing Packages With Different Package Codes Under Same Application Node

You also have the option of forcing packages with different Package Codes to be listed under the same application node. To do this, import the package that has the different Package Code using the **Single application** option of the Import Wizard, and select the application node that you would like it to appear under on the **Destination Group** panel.

Associating a Virtual Package with its Source Windows Installer Package

You can import Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual packages into the Application Catalog and associate them with their source Windows Installer package.

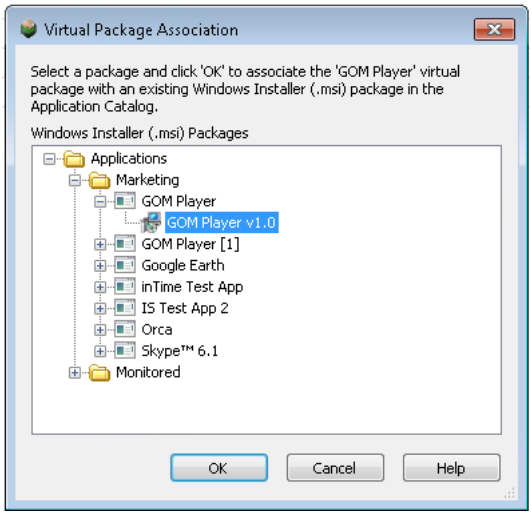
Virtual packages are self-contained entities which ordinarily cannot be modified after they are created. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package.

Application Manager uses a package's Package Code to associate packages and to automatically group them under an application node (if they are imported into the same group).

If a Windows Installer package and a virtual package that AdminStudio created by converting that Windows Installer package are imported into the same Group, the virtual package will be associated with its source Windows Installer package. However, if you want to manually associate a virtual package to its source Windows Installer package, perform the following steps:

Task ***To manually associate a virtual package with a Windows Installer package:***

1. Open the **Home** tab of Application Manager
2. Right-click on the virtual package you want to make an association for, and select **Associate Package** from the shortcut menu. The **Virtual Package Association** dialog box opens, listing Windows Installer packages in the Application Catalog.



3. Select the virtual package's source Windows Installer package and click **OK**. The virtual package is now associated with the selected Windows Installer package.

Caution • *After you have imported a virtual package into the Application Catalog, you are permitted to use the* ***Associate Package*** *function to associate it with any Windows Installer package in the Application Catalog, even one that is not its source package. Therefore, use this feature with caution.*

Deleting a Virtual Package Association

You can delete a virtual package's association with a Windows Installer package by performing the following steps:

Task ***To delete a virtual package association:***

1. Open Application Manager.
2. Right-click on a virtual package, point to **Delete** and click **Package Association** on the shortcut menu. The **Delete Virtual Package Association** dialog box opens, prompting you to confirm the deletion.
3. Select the association that you want to delete and click **OK**. The association is deleted.

About Windows Installer Packages (.msi)

Application Manager supports the import of Windows Installer packages (`.msi`). A Windows Installer package contains all of the information that the Windows Installer requires to install or uninstall an application or product and to run the setup user interface. The `.msi` file can also contain one or more transform files (`.mst`) and one or more patches (`.msp`).

A Windows Installer package is organized around the concepts of components and features:

- A feature is a part of the application's total functionality that a user may decide to install independently.
- A component is a piece of the application or product to be installed.

The Windows Installer always installs or removes a component from a user's computer as a coherent piece. Components are usually hidden from the user. When a user selects a feature for installation, the installer determines which components must be installed to provide that feature.

About Microsoft MSIX Packages (.msix)

Microsoft has introduced a new installer technology, MSIX packages (`.msix`), to support platform independent installations. AdminStudio supports the import of Microsoft MSIX packages into the Application Catalog.

MSIX is the next generation software deployment model for the Windows platform, bringing the best of MSI, AppX and App-V together in a single package. MSIX has the capability to run a traditional Win32 application in a native container, achieving application isolation, ease of deployment, fully clean uninstallation, and seamless software updates all at the same time. MSIX promises to be the alternative to MSI and become the gold standard of Windows deployments. You can choose to continue to use MSI, but the benefits of MSIX will greatly outweigh the need to maintain MSIs long-term, especially given that most enterprises and consumers are increasing their adoption of Windows 10.

About Transforms (.mst)

Application Manager supports the import of Windows Installer packages (`.msi`) with associated transforms (`.mst`). A transform is a collection of changes applied to an installation. By applying a transform to a base installation package, the installer can add or replace data in the installation database. The installer can only apply transforms during an installation.

The installer registers a list of transforms required by the product during the installation. The installer must apply these transforms to the product's installation package when configuring or installing the product.

A transform can modify information that is in any persistent table in the installer database. A transform can also add or remove persistent tables in the installer database. Transforms cannot modify any part of an installation package that is not in a database table, such as information in the summary information stream, information in substorages, information in nested installations, or files in embedded cabinets.

About Patches (.msp)

AdminStudio supports the import of Windows Installer packages (`.msi`) with associated patches (`.msp`). A Windows Installer patch (`.msp` file) is a file used to deliver updates to Windows Installer applications. A patch is a self-contained package that contains all the information required to update an application.

A patch package contains the actual updates to the application and describes which versions of the application can receive the patch. A patch package does not include a database like a regular installation package (`.msi` file). Patches contain at minimum two database transforms. One transform updates the information in the installation database of the application. The other transform adds information that the installer uses for patching files.

About the Administrative Installation of Patches

For patches to be applied to a Windows Installer package, it is necessary to perform an administrative install of the Windows Installer package and then perform an administrative install of each patch package one by one. This way, the content of each patch package is appended to the Windows Installer package at the administrative install location.

In previous releases, when you imported a patch into the Application Catalog, you were prompted to specify a location for an administrative install. However, starting with AdminStudio 2013, you no longer have to specify a location for an administrative install if your Windows Installer package includes patches. Instead, the administrative install operation is automatically performed in a TEMP folder.

About Legacy Installer Packages

Application Manager supports the import of non-MSI legacy setup types (such as InstallShield Professional or ISMP installations) into the Application Catalog. By importing these legacy setup formats, you allow AdminStudio to manage these setups in a manner consistent with other MSI based packages.

When you import a legacy installer (`.exe`), you are prompted to select a setup configuration file (`.ini`) to include with the imported package.

Note • *Testing using Analyze is not supported by the* ***Legacy installer package (.exe)*** *package type.*

Importing Links to Public Store Applications

You can import links to public store apps in the Apple iOS App Store, Apple Mac App Store, Google Play Store, or Windows Store into the Application Catalog. This enables you to prepare and manage them in conformance with your standard application readiness processes. The following features are supported for public store apps:

Table 7-10 • Support for Public Store Apps

Feature	Description
Obtain and view metadata	When a public store app is imported, metadata is automatically extracted from the application's property files and its public store, and that data can be viewed in Application Manager, both on the **General Information** tab of the **Application View** and on the **Tables** view. See Managing Mobile App Metadata.

Table 7-10 • Support for Public Store Apps

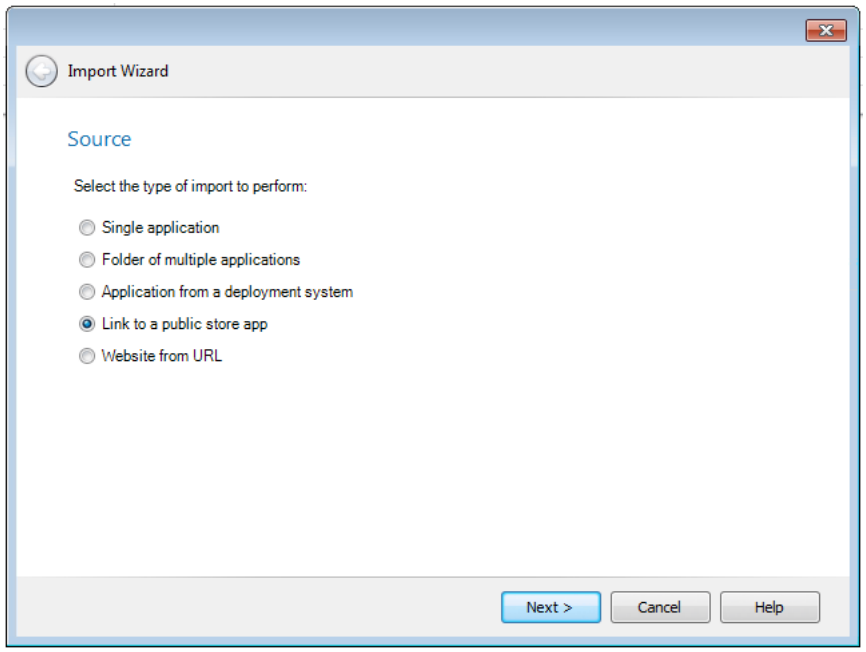
Feature	Description
Perform testing	After you import a public store app, you can perform OS compatibility and risk assessment testing to determine whether deployment of these public store apps will be successful. Test results are displayed in Analyze. See Performing Compatibility, Best Practices, and Risk Assessment Testing. ***Note*** *• Operating system compatibility testing is only available for Apple iOS, macOS, and Google Android public store apps.*
Detailed reporting	In Reports, you can view device compatibility, OS compatibility, and feature use reports for iOS, macOS, Android, and Windows Store apps (local and public store). See Viewing Application Testing and Analysis Reports on the Reports Tab.
Policy compatibility reporting	If you import iOS Enterprise Policy Configuration files (`.mobileconfig` or `.plist`), you can view policy compatibility reports in Reports for both iOS local and public store mobile apps. See Importing Enterprise Policy Configuration Files.
Distribution	You can use the Distribution Wizard to deploy iOS, macOS, Android, and Windows Store mobile apps (both local and public store) to Microsoft System Center 2012 Configuration Manager, Casper Server, and Workspace ONE Server. See Distributing Applications Using the Distribution Wizard.

Note *• For information on importing local mobile apps, see Importing a Single Package File or Importing a Folder of Multiple Applications.*

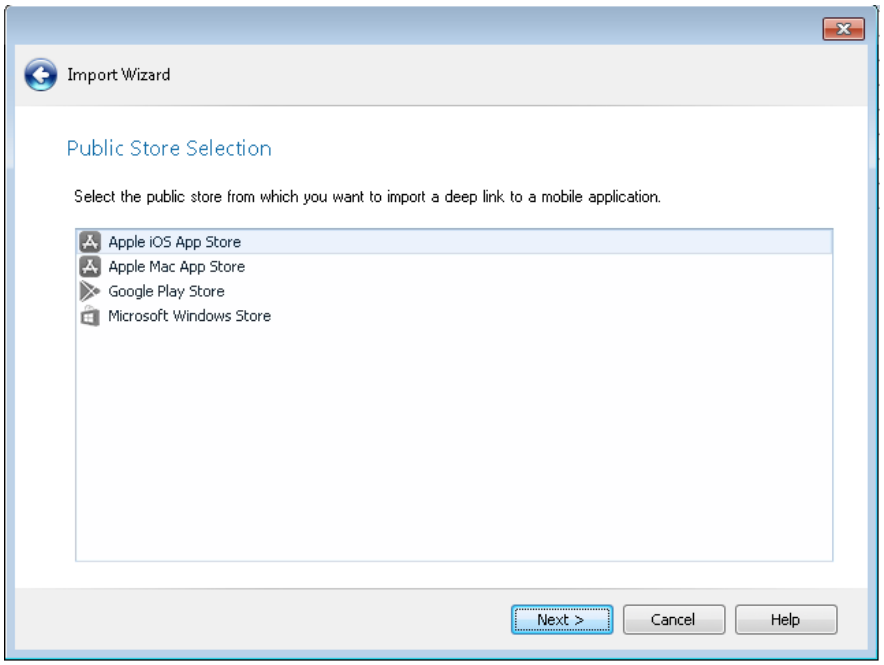
To import a link to public store application, perform the following steps:

Task ***To import a link to a public store mobile app into the Application Catalog:***

1. On the **Home** tab of the Application Manager ribbon, click the **Import** button. The **Source** panel opens.



2. Select **Link to a public store app** and click **Next**. The **Public Store Selection** panel opens.

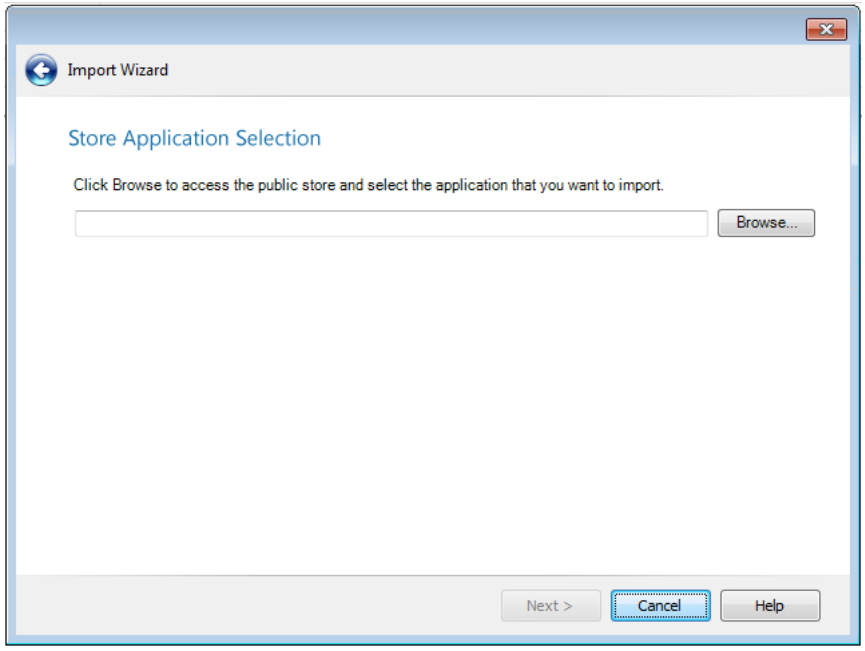


3. Select one of the following public stores:

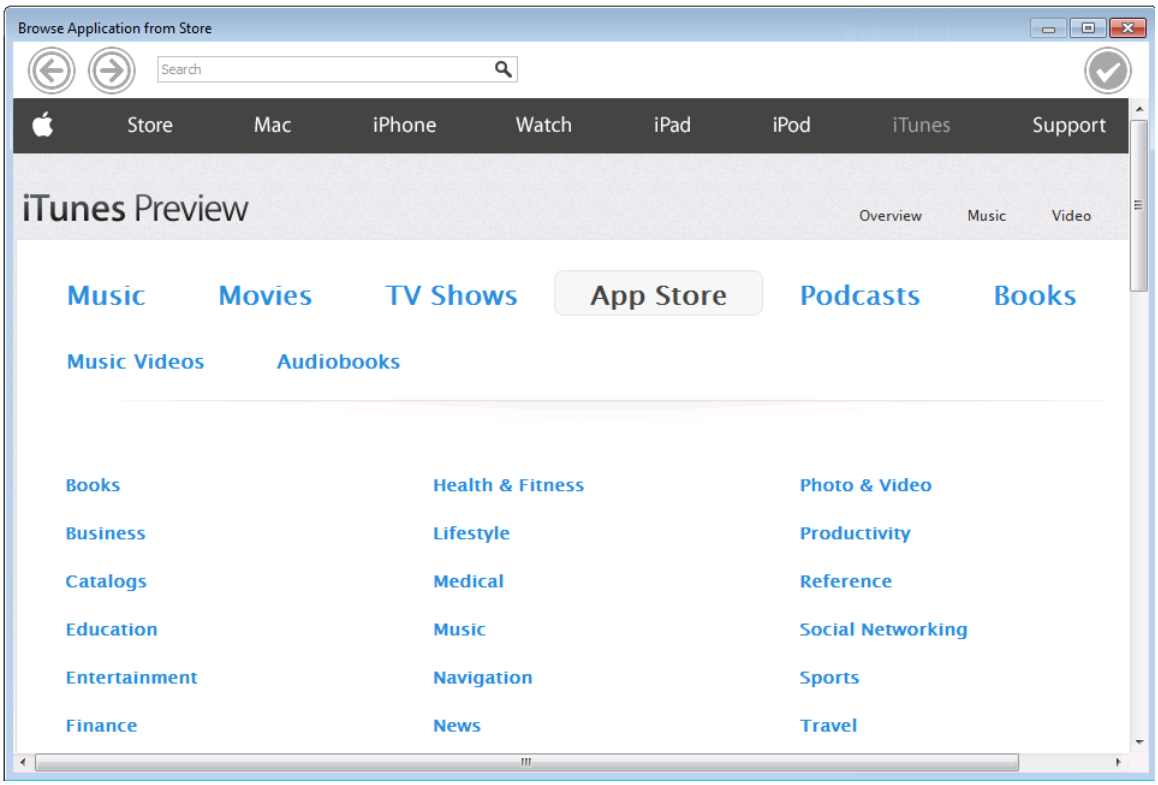
   - **Apple iOS App Store**

- **Apple Mac App Store**
- **Google Play Store**
- **Microsoft Windows Store**

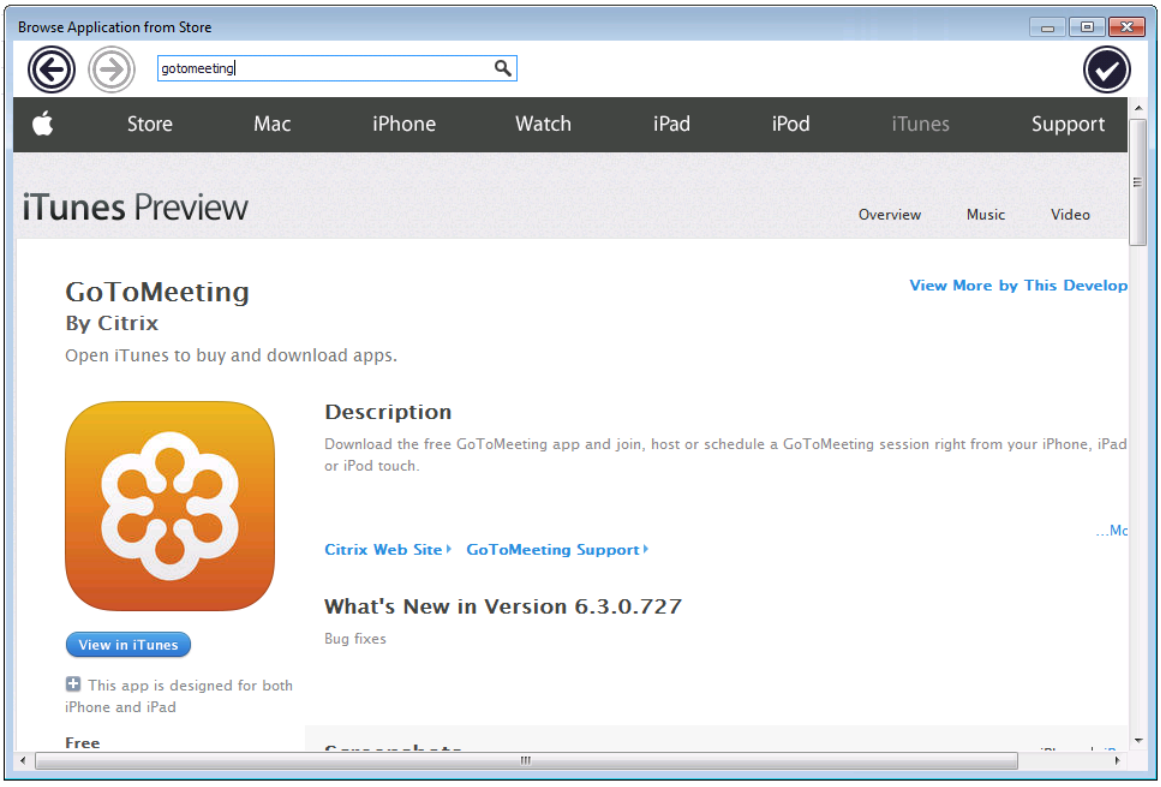
4. Click **Next**. The **Store Application Selection** panel opens.



5. Click **Browse**. The **Browse Application from Store** dialog box opens which displays the browser window of the selected public store.



6. Use the links in the browser window and the search functionality to locate the desired mobile app and open its informational page.



***Note** • You can also use the arrow keys at the top left of the dialog box to navigate through the public store.*

7. When you have opened the informational page of the public store app that you would like to import, click the checkmark button at the top right of the dialog box.

***Tip** • When importing Microsoft Windows Store mobile apps, It is possible that the Windows Store app may open and obscure this checkmark button. If this happens, minimize the Windows Store app, and then return to the browser window and click the checkmark button.*

The link to the selected public store app is now listed on the **Store Application Selection** panel, such as:

`https://itunes.apple.com/us/app/gotomeeting/id424104128?mt=8`

8. Click **Next**. The **Destination Group** panel opens.
9. Select a destination group into which your public store app will be imported.

***Note** • If you launched the Import Wizard by selecting a group in the tree and then clicking **Import**, that group will be selected by default on the **Destination Group** panel.*

***Note** • If you want to import the package into a new group, click the **New Group** button to create a new group.*

10. Click **Next**. The **Summary** panel opens.
11. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Next** to start the import. Progress messages are displayed on the **Running the Import** dialog box.
12. When the import is complete, click **Finish** to close the wizard.

The mobile app is now listed in the Application Manager tree using one of the following icons:

Public Store App Type	Icon	URL
Apple iOS App Store **Apple Mac App Store**		`https://itunes.apple.com/us/app`
Google Play Store		`https://play.google.com/store/apps`
Microsoft Windows Store		`http://windows.microsoft.com/en-us/windows/search#q=productivity+buisness &s=Store`

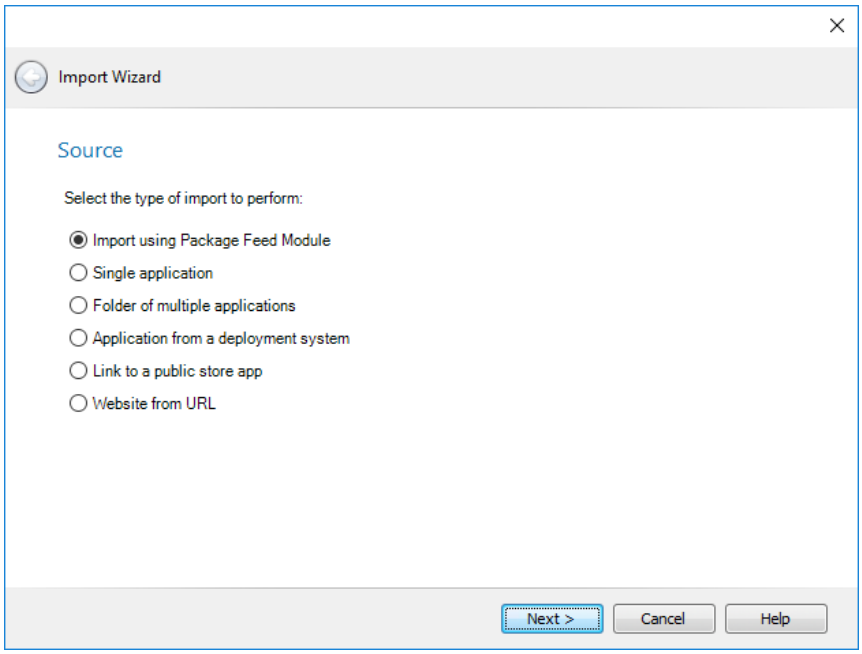
Importing a Folder of Multiple Applications

You can import multiple packages (of selected package types) from a directory of packages into the Application Catalog using the **Folder of multiple applications** option of the Import Wizard. You can use this method to import all of the package types described in Package Types Supported By the Import Wizard.

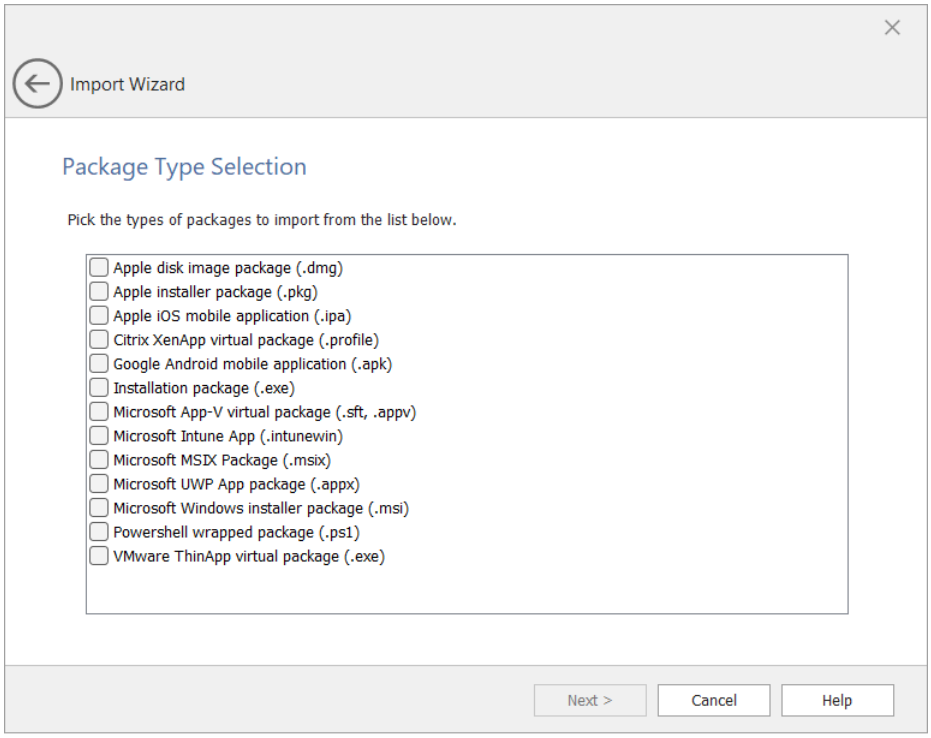
To import a directory of packages, perform the following steps.

Task ***To import a directory of packages:***

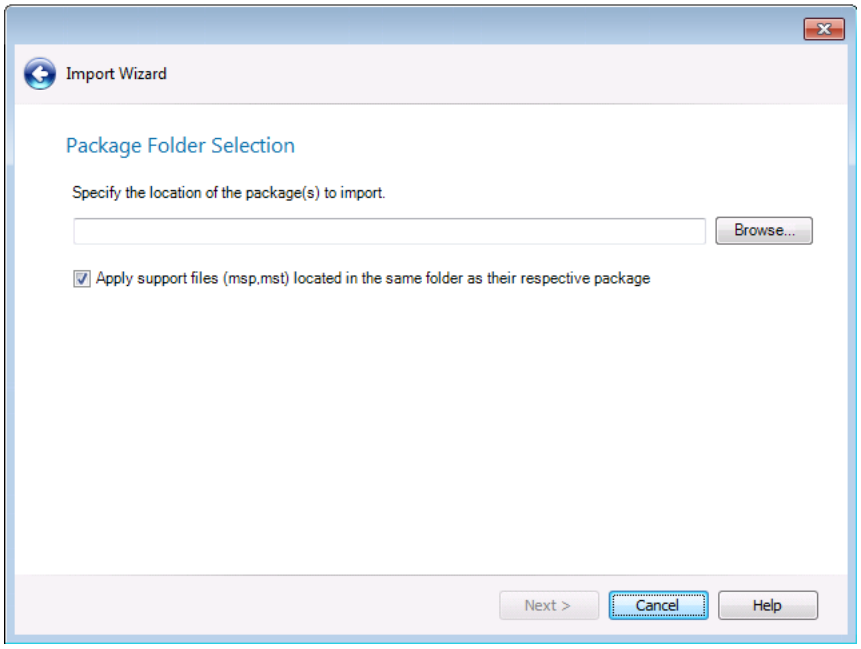
1. Open **Application Manager**.
2. On the **Home** tab of the Application Manager ribbon, click **Import**. The **Source** panel of the Import Wizard opens.



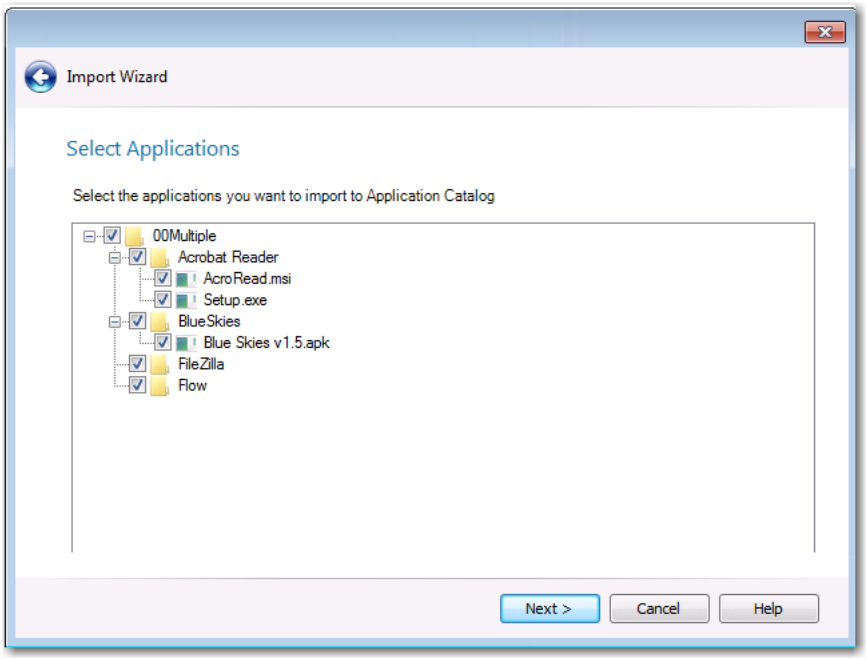
3. Select **Folder of multiple applications** and click **Next**. The **Package Type Selection** panel opens, prompting you to select the package deployment types that you want to import.



4. Select the package types that you want to import and click **Next**. The **Package Folder Selection** panel opens, prompting you to select the directory containing the packages you want to import.



5. Click **Browse** and select the directory containing the packages that you want to import.
6. Optionally, if you also want to import package support files (such as transforms or patch files), select the **Apply support files (.msp, .mst) located the same folder as their respective package** option.
7. Click **Next**. The **Select Applications** panel opens.



8. By default, all applications containing the selected package type are selected. You can clear the selection of any packages you do not want to import.
9. Click **Next**. The **Destination Group** panel opens.
10. Select the destination group into which the packages will be imported.

Note • *If you launched the Import Wizard by selecting a group in the tree and then clicking* ***Import****, that group will be selected by default on the* ***Destination Group*** *panel.*

Note • *If you want to import the packages into a new group, click the* ***New Group*** *button to create a new group.*

11. Set the **Create subgroups based on source folder structure** option to determine the location of the imported packages in the Application Manager tree:
    - **Selected**—Subgroups of the selected group will be created in the Application Manager tree that mimic the directory structure of the selected directory, and the packages will be imported into those subgroups.
    - **Not selected**—All of the packages in the selected directory (and its subdirectories) will be imported into the root of the selected group.
12. Click **Next**. The **Summary** panel opens.
13. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Next** to start the import. Progress messages are listed on the **Running the Import** panel.
14. When import is complete, click **Finish** to close the wizard. The packages will then be listed in the Application Manager tree.

Note • *Depending on whether options have been set on the* ***Import Options / General*** *tab of the Application Manager* ***Options*** *dialog box (available from the Application Catalog tab menu), testing may be performed after import.*

Note • *For information on how the Import Wizard decides which packages to import, see Import Wizard's Selection Rules When Importing Packages from a Directory.*

Creation of Application Nodes During Folder of Multiple Applications Import

Application nodes are created in the Application Manager tree using the package's associated Package Code. If multiple packages of different deployment types (such as Windows Installer, App-V, and ThinApp) of the same software product all have the same Package Code and are all imported using the **Folder of Multiple Package Import** option (without the **Create subgroups based on source folder structure** option), all of the deployment types will be automatically listed under the same application node.

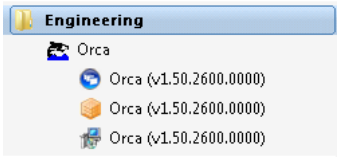


Figure 7-3: Three Deployment Types Under One Application Node

However, if multiple packages of different deployment types of the same software product have different Package Codes, and are all imported into the same Group, an additional node for that application will be created for each Package Code, incremented by a number, such as: `ABC Application` and `ABC Application [1]`.

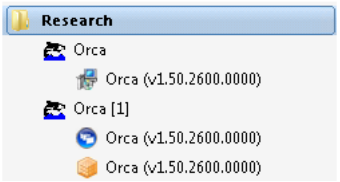


Figure 7-4: Three Deployment Types Under Two Different Application Nodes

Forcing Packages With Different Package Codes Under Same Application Node

You also have the option of forcing packages with different Package Codes to be listed under the same application node. To do this, import the package that has the different Package Code using the **Single application** option of the Import Wizard, and select the application node that you would like it to appear under on the **Destination Group** panel.

Importing From ConfigMgr (Formerly called as System Center Configuration Manager)

AdminStudio supports both Microsoft System Center 2012 Configuration Manager's application data model and the package model used in System Center 2007 Configuration Manager. Therefore, AdminStudio provides the following benefits related to migrating to Microsoft System Center 2012 Configuration Manager:

- **Eases your migration to Microsoft System Center 2012 Configuration Manager**—Because AdminStudio gives you the ability to choose the model which is best for your environment, it allows you to migrate to System Center 2012 Configuration Manager your own pace, minimizing possible disruptions to your functioning processes around packaging and repackaging during your migration. As you prepare your applications for migration to System Center 2012 Configuration Manager by importing them into the Application Catalog and collecting and reviewing metadata, you can continue to deploy packages from AdminStudio to System Center 2007 Configuration Manager.
- **Automates identification of System Center 2012 Configuration Manager metadata**—AdminStudio automates the identification of application metadata required for the population of System Center 2012 Configuration Manager's application model. Data is automatically collected during import into the Application Catalog—which is displayed in an organized, easy-to-navigate tabbed layout—and wizards are provided for you to quickly and easy add additional data. Using these features significantly reduces the time and cost to manually identify this metadata, and helps to speed the adoption of the user-centric deployment of applications using System Center 2012 Configuration Manager. For more information, see Managing ConfigMgr (Formerly called as System Center Configuration Manager) Application Model Data and Managing ConfigMgr (Formerly called as System Center Configuration Manager) Package Deployment Data.

You use the Import Wizard to perform a bulk import of Windows Installer, Microsoft App-V, Apple iOS, Google Android, and Windows Store packages from ConfigMgr (Formerly called as System Center Configuration Manager) server into the Application Catalog.

- Importing Applications, Mobile Apps, and Packages from ConfigMgr (Formerly called as System Center Configuration Manager)
- Package Information Imported from ConfigMgr (Formerly called as System Center Configuration Manager)

Importing Applications, Mobile Apps, and Packages from ConfigMgr (Formerly called as System Center Configuration Manager)

You can import applications and mobile apps from System Center 2012 Configuration Manager or packages from System Center 2007 Configuration Manager into the Application Catalog.

You can import the following deployment types from System Center 2012 Configuration Manager:

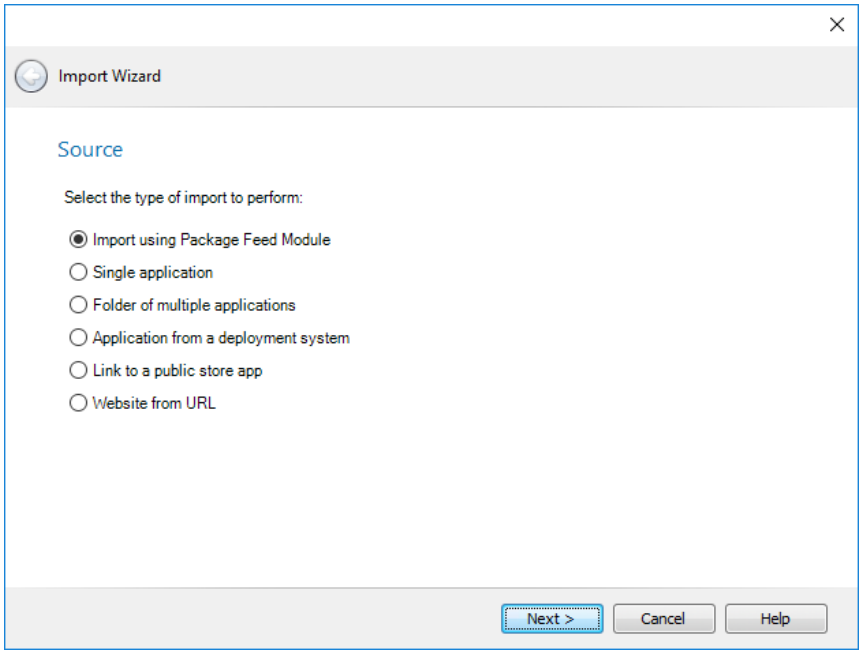
- Windows Installer
- Microsoft App-V 4.x and 5.0
- Apple iOS (local and public store link)
- Google Android (local and public store link)
- Windows Store (local only)

When you import packages from ConfigMgr (Formerly called as System Center Configuration Manager) from versions prior to 2012, Application Manager also imports information that will assist you in migrating that package to System Center 2012 Configuration Manager's application model.

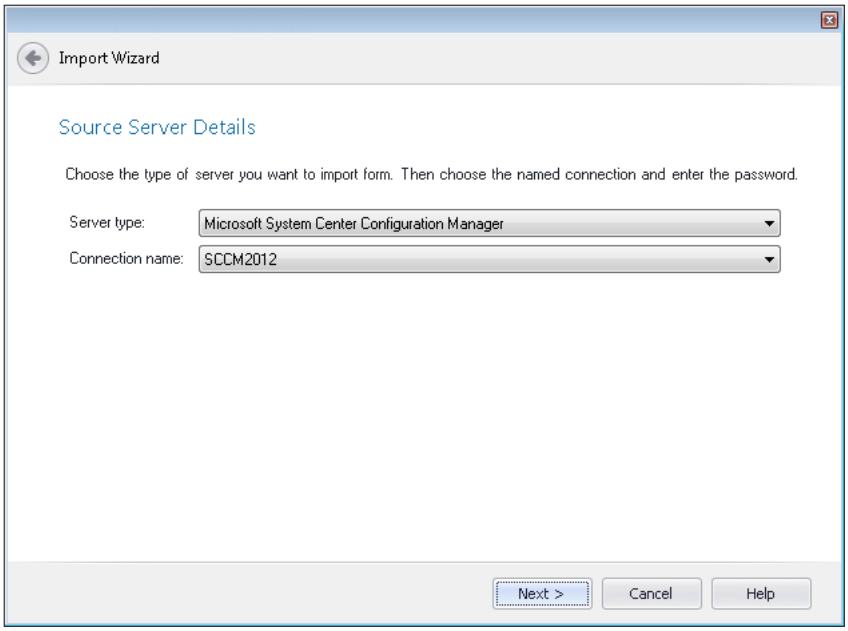
Note • *Before you can perform an import from ConfigMgr (Formerly called as System Center Configuration Manager), you need to define a named connection to a ConfigMgr (Formerly called as System Center Configuration Manager) Server, as described in Creating a New Distribution System Connection Setting.*

Task ***To import applications and mobile apps from Microsoft System Center 2012 Configuration Manager:***

1. Open Application Manager and specify a named connection to a ConfigMgr (Formerly called as System Center Configuration Manager) server, as described in Creating a New Distribution System Connection Setting.
2. On the **Home** tab of the ribbon, click the **Import** button. The **Source** panel of the **Import Wizard** opens.

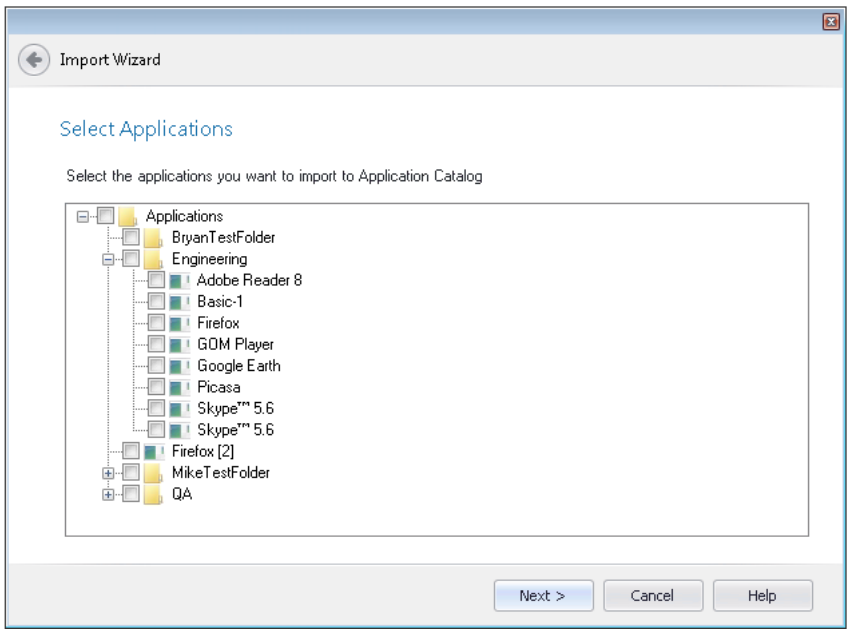


3. Select **Packages from a deployment system** and click **Next**. The **Source Server Details** panel opens.



4. From the **Server type** list, select **ConfigMgr (Formerly called as System Center Configuration Manager)**.

5. From the **Connection name** list, select the named connection you created to your ConfigMgr (Formerly called as System Center Configuration Manager) server.
6. Click **Next**. The **Select Applications** (for System Center 2012 Configuration Manager) or **Select Packages** for System Center 2007 Configuration Manager) panel opens, listing all of the applications or packages in the specified server.



7. Select the applications or packages that you want to import.
8. Click **Next**. The **Destination Group** panel opens.
9. On the **Destination Group** panel, select the group into which you want the selected applications to be imported.

Note • *If you launched the Import Wizard by selecting a group in the tree and then clicking* ***Import****, that group will be selected by default on the* ***Destination Group*** *panel.*

Note • *If you want to import the packages into a new group, click the* ***New Group*** *button to create a new group.*

10. Set the **Create subgroups based on source folder structure** option to determine the location of the imported packages in the Application Manager tree:
    - **Selected**—Subgroups of the selected group will be created in the Application Manager tree that mimic the directory structure of the selected ConfigMgr (Formerly called as System Center Configuration Manager) directory, and the packages will be imported into those subgroups.
    - **Not selected**—All of the packages in the selected ConfigMgr (Formerly called as System Center Configuration Manager) directory (and its subdirectories) will be imported into the root of the selected group.
11. Click **Next**. The **Summary** panel opens.
12. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Next** to start the import.

Progress messages are listed on the **Running the Import** panel.

13. When import is complete, click **Finish** to close the wizard. The applications imported from ConfigMgr (Formerly called as System Center Configuration Manager) will then be listed in the Application Manager tree.

Note • *Depending on whether options have been set on the **Import Options** / **General** tab of the Application Manager **Options** dialog box (available from the Application Catalog tab menu), testing may be performed after import.*

Package Information Imported from ConfigMgr (Formerly called as System Center Configuration Manager)

When you import a package from ConfigMgr (Formerly called as System Center Configuration Manager), information from Configuration Manager is imported along with it and that information is displayed on the **Deployment Data** tab of the package's **Home Deployment Type View**.

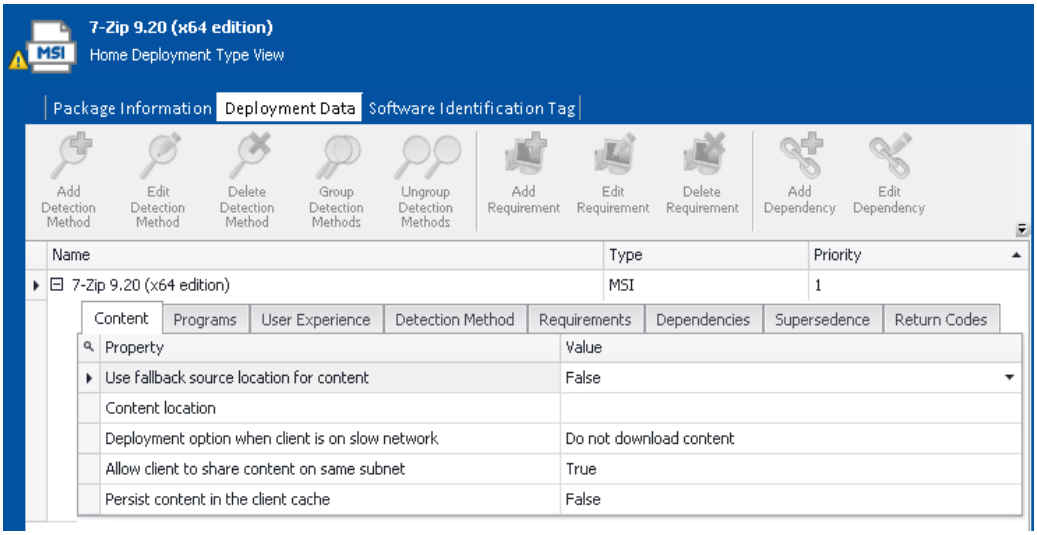


Figure 7-5: Deployment Data Tab of Home Deployment Type View

The following information is imported from ConfigMgr (Formerly called as System Center Configuration Manager):

Table 7-11 • Package Information Imported from ConfigMgr (Formerly called as System Center Configuration Manager)

System Center 2007 Configuration Manager Program Property	Application Model Property on Deployment Data Tab	
Program > General > Command Line	**Programs** subtab	**Install Command Line**
Program > Requirements > Estimated Disk Space	**Requirements** subtab	**Disk Space** condition
Program > Requirements > Client Platforms	**Requirements** subtab	**Operating System** condition
Program > Advanced > Run Another Program First	**Dependencies** subtab	**Dependency**
Program Advertised to Collection	**Requirements** subtab	**Custom Requirement** (derived from collection queries of the advertised collection)

Importing Merge Modules

For optimal performance, Merge Modules should be imported into an Application Catalog database prior to importing Windows Installer packages. This ensures that conflicts resulting from not using available merge modules are correctly identified.

Task ***To import Merge Modules:***

1. Open **Application Manager**.
2. Open the **Merge Modules** tab.
3. Click the **Import** button in the ribbon. The **MSM Source Information** panel of the Merge Module Import Wizard opens.

   Note • *You can also open the Merge Module Import Wizard by right-clicking on the root Merge Module group or one of the imported Merge Modules in the tree and then selecting* ***Import Merge Module*** *from the shortcut menu.*

4. Click the Browse ( ) button in the **Merge Modules** area and select the merge module file that you want to import.
5. To import multiple merge modules, you can repeat the procedure as necessary.
6. The order in which merge modules are applied can be changed by selecting a merge module in the list and clicking the Move Up and Move Down arrows.
7. If you need to delete a merge module you have added, clear its check box.
8. Click **Next**. The **Summary** panel opens.

9. Click **Finish** to accept these options and begin the import.

   A report of the import process appears on the **Import** tab in the Output window.

About Merge Modules (.msm)

Application Manager supports the import of merge modules (.msm). Merge modules are essentially simplified Windows Installer `.msi` files.

A merge module cannot be installed alone because its lacks some vital database tables that are present in an installation database. Merge modules also contain additional tables that are unique to themselves. To install the information delivered by a merge module with an application, the module must first be merged into the application's .msi file.

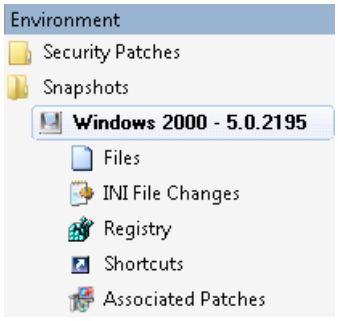
Importing OS Snapshots

You can import OS Snapshot (`.osc`) files into the Application Catalog to use to determine conflicts between an operating system and a package. OS Snapshot files are files representing a particular computer system's contents. To generate an OS snapshot file, use the OS Snapshot Wizard, as described in Capturing an OS Snapshot, to scan a computer's operating system and record the files, INI files, shortcuts, and registry entries present.

Caution • *The OS Snapshot Wizard saves OS Snapshot information in two files: an `.osc` file (a collection of file type information) and an associated `.nir` file (registry information). The `.nir` file must be present in the same directory as the `.osc` file in order for import to be successful.*

Task ***To import an OS Snapshot into an Application Catalog:***

1. Open Application Manager.
2. Click on the **Environment** tab. The tree lists the Security Patches, OS Snapshots, and Enterprise Policy Configuration files that have already been imported into the Application Catalog.



3. In the tree, right-click on the **Snapshots** group and select **Import Snapshot** from the shortcut menu. The **File Selection** panel opens.
4. Click **Browse** and select the OS Snapshot (`.osc`) file that you want to import.
5. Click **Next**. The **Summary** panel opens.
6. Click **Finish** to accept these options and begin the import.

Progress messages are listed on the **Import** tab of the Output window.

About OS Snapshots (.osc)

Application Manager supports the import of OS Snapshot (`.osc`) files, which are files representing a particular computer system's contents. To generate an OS snapshot file, use the OS Snapshot Wizard to scan a computer's operating system and record the files, INI files, shortcuts, and registry entries present. The Wizard saves this information in an `.osc` file (a collection of file type information) and an associated `.nir` file (registry information).

When an OS Snapshot file is imported into Application Manager, it can be used as a reference point for conflict identification. See Taking OS Snapshots for more information.

To provide maximum flexibility during the OS Snapshot process, you can use the Exclusions Editor to create an exclusion list that identifies files, INI files, shortcuts, and registry entries that the OS Snapshot Wizard should disregard during the scan. Using this list, you can eliminate unnecessary files, shortcuts, or registry entries, and reduce the time it takes to perform the OS Snapshot. See Configuring Exclusions Using the Exclusions Editor for more information.

Caution • *OS Snapshots should only be used for comparison in Application Manager. You should never attempt to convert an OS Snapshot into an MSI package.*

Using Duplicate Package Identifiers

When you import a package into an Application Catalog database, Application Manager checks specific identifiers that are selected on the **Import Options / Duplicate Package** tab of the Application Manager **Options** dialog box to determine if that package has already been imported.

- **For Windows Installer files**, the following identifiers are listed on the **Duplicate Package** tab:
  - PackageCode
  - ProductCode
  - Product Language
  - ProductVersion
  - List of Transform Files

  If none of these identifiers are selected, Application Manager will use the Product Name identifier to perform a Duplicate Package check.
- **For App-V packages**, the following identifiers are listed on the **Duplicate Package** tab:
  - PackageGUID
  - VersionGUID

  If neither of these identifiers are selected, Application Manager will use the **Product Name** identifier to perform a Duplicate Package check.

If Application Manager determines that you are attempting to import a duplicate package (based upon the selected identifiers), the package is renamed using the syntax defined in the **Duplicate Package Naming Syntax** field.

Business Case for Importing a Package Multiple Times

You might encounter this situation if you are importing the same package into the same Application Catalog database multiple times, each time with a different set of transforms. One common reason why you might want to import the same package into your Application Catalog database more than once would be if you wanted to use InstallShield Editor to create custom installation SKUs of a common MSI package to distribute to different departments in your organization, each installation including certain features that are appropriate for the department and excluding certain features that are not appropriate. For example, if you were distributing a copy of Microsoft Office, you could add transforms to the Microsoft Office MSI package so that:

- Accounting's installation would include only Word and Excel
- Marketing's installation would include only Word and PowerPoint, and
- Development's installation would include only Word and Access.

Therefore, you might want to import the same package into your database more than once, each time with a different set of transformations. What happens when you import the package the second time depends upon the identifiers you selected on the **Duplicate Package** tab. In this example:

- If you select the **List of Transform Files** and **ProductCode** identifiers on the **Duplicate Package** tab of the Application Manager **Options** dialog box, Application Manager will not identify these two packages as duplicate, even though they have the same **ProductCode**, because they have a different set of transformations. Therefore, the package will be imported with the same display name as the first package.
- If you only select the **ProductCode** identifier on the **Duplicate Package** tab of the Application Manager **Options** dialog box, Application Manager will identify the second package as a duplicate because the two packages have the same **ProductCode**.

Duplicate Product Name Conventions

When it identifies a duplicate package, by default Application Manager generates a new name for that package by pre-pending the Product Name with the Manufacturer's name and, if necessary, appending the Product Name with numbers:

- **1st Package**: PowerPoint
- **2nd Package**: Microsoft Corporation_PowerPoint

You can edit the **Duplicate Package Naming Syntax** on the **Import Options / Duplicate Package** tab of the **Options** dialog box.

When Duplicate Packages are Identified During Bulk Import

If Application Manager is performing a bulk import or reimport, it still identifies duplicate packages using the user-specified criterion and will rename duplicate packages using the syntax in the **Duplicate Package Naming Syntax** field.

Specifying Duplicate Package Identifiers

To specify duplicate package identifiers, perform the following steps.

Task ***To identify duplicate package identifiers:***

1. On the Application Manager tab menu, click **Options**. The Application Manager **Options** dialog box opens
2. Under **Import Options**, select **Duplicate Package**. The **Duplicate Package** tab opens.

3. Under **Duplicate Package Identification Options**, select the identifiers that you would like to use for Windows Installer package imports by selecting one or more of the following options:

   - **PackageCode Property**— Identifier of the package the product was installed from. No two non-identical .msi files should ever have the same package code.
   - **ProductCode Property**—Unique identifier for the particular product release, represented as a string GUID, for example:

     `{12345678-1234-1234-1234-123456789012}`

   - **Product Language**—The language the installer should use for any strings in the user interface that are not authored into the database.
   - **ProductVersion**—Version of the product in string format. The format of the string is: major.minor.build.
   - **List of Transform Files**—A list of the transformations associated with this package.
   - **[None Selected]**—If you do not select any of these five identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.

4. Under **Duplicate Virtual Package Identification Options**, select the identifiers that you would like to use for App-V package imports by selecting one or more of the following options:

   - **PackageGUID**—Unique identifier of App-V package.
   - **VersionGUID**—Unique identifier of App-V package version.
   - **[None Selected]**—If you do not select either of these identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.

5. Click **OK** to save your changes and exit the **Options** dialog box
6. Proceed with the package import, as described in .

Note • *The options that you select on the **Import Options** / **Duplicate Package** tab of the Application Manager **Options** dialog box apply globally to all packages that you attempt to import; you cannot apply different identifiers to different packages. Also, since these options are saved in the AdminStudio Shared Directory, everyone using AdminStudio at your organization will share the same Duplicate Package options.*

Generating Software ID Tag Files During Package Import

AdminStudio includes ISO/IEC 19770-2 tagging as a standard capability, which simplifies the software asset management task by enabling the collection and analysis of the tags to provide an accurate application inventory.

ISO/IEC 19770-2 is an international standard for the creation of software identification tags. A software identification tag is a small, XML-based file that contains descriptive information about the software, such as the product name, product edition, product version, and publisher. Applications with tag files can be easily identified after installation. Software asset management tools collect the data in the tags to provide accurate application identification for software that is installed in an enterprise.

AdminStudio adds tag files—which contain both ISO 19770-2 compliant tag information and AdminStudio's extended tag information—to Windows Installer packages that it processes in two locations:

- **Packages built by Repackager**—By default, whenever Repackager builds a Windows Installer package (even when building one silently), a software ID tag file is created for that package. For more information, see Generating Software ID Tag Files During Repackaging.
- **Packages imported into the Application Catalog**—By default, tag files are created for each package that is imported into the Application Catalog. When Application Catalogs from versions of AdminStudio prior to 11.0 are upgraded, AdminStudio will, upon your approval, create tag files for all packages during upgrade.

In both of these cases, AdminStudio stores the ISO tag file in an external transform file.

Information about generating software ID tag files during package import is provided in the following topics:

- About Software ID Tag File Generation
- Viewing and Editing Package Tag Information in Application Manager

About Software ID Tag File Generation

AdminStudio includes ISO/IEC 19770-2 software tagging support. ISO/IEC 19770-2 is an international standard for the creation of software identification tags, which are small, XML-based files that contain descriptive information about a software package. This information is used by software asset management tools to identify the software that is installed in an enterprise.

AdminStudio adds software ID tag files to Windows Installer packages whenever Repackager builds a Windows Installer package or when packages are imported into the Application Catalog (or when an Application Catalog is upgraded).

Information about software ID tag files is presented in the following sections

- How Tag Files Are Named
- Output Files Created by Tag File Generation: .mst and .cab
- Sample Software ID Tag File
- Creation of Tag Files During Application Catalog Upgrade
- Support for Packages With Multiple Tag Files
- About Software Tagging RegIDs
- About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields

How Tag Files Are Named

AdminStudio uses the following format to name the software ID tag files that it creates:

```
SoftwareCreatorRegID.UniqueID.swidtag
```

The tag file's name is comprised of the following components:

- **SoftwareCreatorRegID**—By default, AdminStudio uses the value of the **Tag Creator RegID** field on the **Software Tagging** tab of the **General Options > Import Options > Software Tagging** tab of the Application Manager **Options** dialog box, which is:

```
regid.2009-06.com.flexera,AdminStudio
```

You may edit this value for a package that has been imported into the Application Catalog on the **Software Identification Tag** subtab of the Application Manager **Home Deployment Type View**. You can also edit this value for a Repackager project on the **Software Identification Tag** view in the Repackager interface.

- **UniqueID**—This portion of the tag file name uniquely identifies the package by using the product GUID, which is the ProductCode of the MSI package or the unique string used for the Add and Remove Programs uninstall key name.
- **.swidtag**—The `.swidtag` file extension is given to this XML file.

Here is an example of a software ID tag file name:

```
regid.2009-06.com.flexera,AdminStudio_6F7CB29F-1319-4816-B345-0856916EB801.swidtag
```

Output Files Created by Tag File Generation: .mst and .cab

When AdminStudio creates a tag file for a Windows Installer package, two files are created in the same directory as the `.msi` file:

- **Compressed (.cab) file**—A compressed `.cab` file is created to contain the tag file. The `.cab` file is named `PackageName_SoftwareID.cab`. It contains the `.swidtag` XML tag file.
- **Transform (.mst) file**—A transform `.mst` file is created to associate the tag file in the `.cab` file with the Windows Installer package. The transform file is named `PackageName_SoftwareID.mst`.

All three files are placed in the same directory:

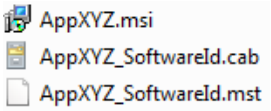


Figure 7-6: Software Tag Transform and CAB File

Sample Software ID Tag File

The following is a sample of a software ID tag file that is created by AdminStudio:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<swid:software_identification_tag xsi:schemaLocation="http://standards.iso.org/iso/19770/-2/2009/
schema.xsd http://standards.iso.org/iso/19770/-2/2009/schema.xsd" xmlns:ds="http://www.w3.org/2000/09/
xmldsig#" xmlns:swid="http://standards.iso.org/iso/19770/-2/2009/schema.xsd" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:fs="http://www.flexera.com/swid/2011/arp">
    <!-- Mandatory elements -->
    <swid:entitlement_required_indicator id="entitlement_required_indicator">true
    </swid:entitlement_required_indicator>
    <swid:product_title id="product_title">ProductABC</swid:product_title>
    <swid:product_version id="version_name">
        <swid:name>4.00.0000</swid:name>
        <swid:numeric>
            <swid:major id="version_major">4</swid:major>
            <swid:minor id="version_minor">0</swid:minor>
            <swid:build id="version_build">0</swid:build>
            <swid:review id="version_review">0</swid:review>
        </swid:numeric>
    </swid:product_version>
    <swid:software_creator>
```

```
        <swid:name id="creator_name">Flexera</swid:name>
        <swid:regid id="creator_regid">regid.1986-12.com.flexera</swid:regid>
    </swid:software_creator>
    <swid:software_licensor>
        <swid:name id="licensor_name">Flexera</swid:name>
        <swid:regid id="licensor_regid">regid.1986-12.com.flexera</swid:regid>
    </swid:software_licensor>
    <swid:software_id>
        <swid:unique_id id="unique_id">ProductABC_4.0.0_D8F6AD25-2351-D3D1-D235-13JSL23HS151
        </swid:unique_id>
        <swid:tag_creator_regid id="tag_creator_regid">regid.2009-06.com.flexera,
        AdminStudio</swid:tag_creator_regid>
    </swid:software_id>
    <swid:tag_creator>
        <swid:name id="tag_name">Flexera</swid:name>
        <swid:regid id="tag_regid">regid.2009-06.com.flexera,AdminStudio</swid:regid>
    </swid:tag_creator>
    <swid:extended_information xsi:schemaLocation="http://www.flexera.com/swid/2011/arp
    http://www.flexera.com/swid/2011/arp">
        <fs:original_arp_guid id="original_arp_guid">D8F6AD25-2351-D3D1-D235-13JSL23HS151
        </fs:original_arp_guid>
        <fs:original_arp_publisher id="original_arp_publisher">Flexera
        </fs:original_arp_publisher>
        <fs:original_arp_display_name id="original_arp_display_name">Product ABC 4.0
        </fs:original_arp_display_name>
        <fs:original_arp_display_version id="original_arp_display_version">4.0.0
        </fs:original_arp_display_version>
        <fs:current_arp_guid>D8F6AD25-2351-D3D1-D235-13JSL23HS151</fs:current_arp_guid>
        <fs:current_arp_publisher>Flexera</fs:current_publisher>
        <fs:current_arp_display_name>Product ABC 4.0</fs:current_arp_display_name>
        <fs:current_arp_display_version>4.0.0</fs:current_arp_display_version>
        <fs:adminstudio_app_catalog_package_id>13</fs:adminstudio_app_catalog_package_id>
        <fs:adminstudio_app_catalog_machine_name>sch101</fs:adminstudio_app_catalog_machine_name>
        <fs:adminstudio_app_catalog_db_name>jan18_1</fs:adminstudio_app_catalog_db_name>
        <fs:adminstudio_app_catalog_guid>9BC14888-65EA-8F03</fs:adminstudio_app_catalog_guid>
    </swid:extended_information>
</swid:software_identification_tag>
```

Extended “ARP” Information in Tag File

The fields in the <swid:extended_information> element contain the package’s original and current ARP (Add or Remove Programs) information. Having this ARP information in the tag file makes it easier to keep track of the original ARP entry information and to preserve it during repackaging. This extended information is added to all software ID tag files created by AdminStudio.

Application Catalog Reference Information in Tag File

Software tag files that are created by importing a package into the Application Manager Application Manager will include an additional field in the <swid:extended_information> element to enable the package to be referenced in the Application Catalog:

```
< swid:extended_information>
    <fs:adminstudio_app_catalog_package_id>APP_ID_FROM_AS CATALOG
    </fs: adminstudio_app_catalog_package_id>
</ swid:extended_information>
```

Creation of Tag Files During Application Catalog Upgrade

The first time you open an Application Catalog database from a version of AdminStudio prior to 11.0, the following message appears to notify you that AdminStudio will reimport all of your Windows Installer packages so that software tag ID files can be created for them:

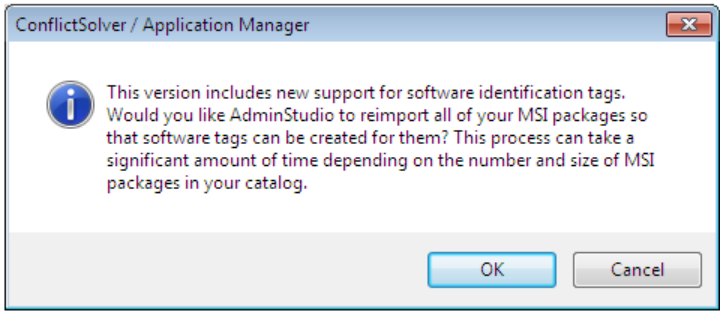


Figure 7-7: Creation of Software Tag File During Application Catalog Upgrade

If you click **Cancel**, software ID tag files will not be created.

Support for Packages With Multiple Tag Files

It is possible for a Windows Installer package to be associated with more than one tag file in Application Manager and in the Repackager interface.

If a Windows Installer package installs more than one product—which results in more than one entry in the list of installed products (Add or Remove Programs list)—it can have more than one associated software identification tag file. Both tag files will be stored in the same location in the `.cab` file.

If a Windows Installer package or a Repackager project includes more than one tag file, there will be two tabs (one for each tag file) displayed on the **Software Identification Tag** subtab of the **Home Deployment Type View** in Application Manager and on the Repackager **Software Identification Tag** view:

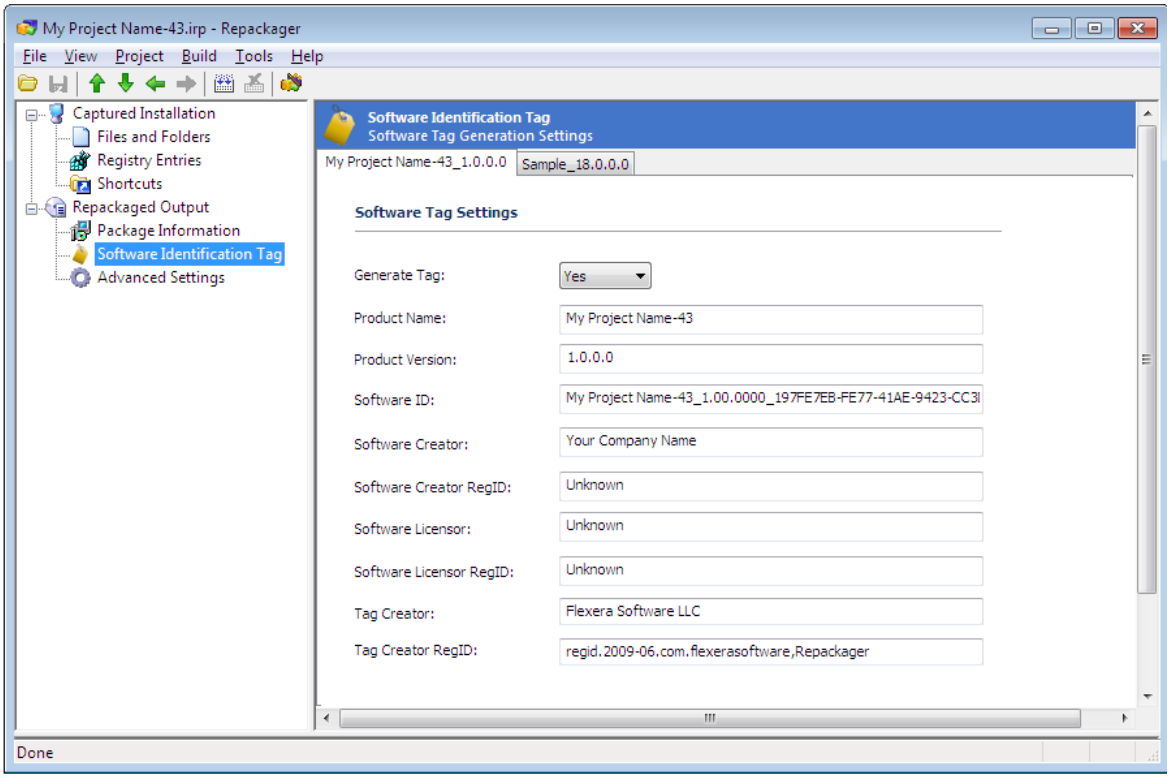


Figure 7-8: Repackager's Software Identification Tag View: Displaying Two Tag Files

If a Windows Installer package has multiple tag files, software asset management tools will treat each tag file as a separate product.

How Existing Tag Information is Incorporated Into the Software ID Tag File

If a Windows Installer project that you are either repackaging or importing into the Application Catalog already has an associated vendor tag file, AdminStudio will incorporate the existing tag information into the new tag file.

AdminStudio makes a copy of the original tag file and modifies it to add the AdminStudio extended information fields. This ensures that custom fields introduced by the vendor are not lost. Also, AdminStudio will not modify any of the standard software tag fields in existing tags. This ensures that any digital signing of these fields will remain valid.

Assuming that you have a Windows Installer package that creates an ARP entry and with an existing vendor tag file, the following scenarios could occur:

- **Repackaging**—Repackager will examine the information in the vendor tag file and determine whether it corresponds to the ARP entry in the Windows Installer package by matching one of the properties, such as Product Name. If there is a match, Repackager will copy the vendor tag information into a new tag file and then add the AdminStudio extended information nodes to that tag file to include information such as original product code, original product name, original product version, original publisher, current product code, current product name, current product version, and current publisher. The final Windows Installer package will have a transform file created along with the updated tag file. The original vendor tag file will not be present in the Windows Installer package.
- **Importing into Application Manager**—Application Manager will detect the existing vendor tag information and add the AdminStudio extended information nodes related to current product code, current publisher, current product name, current product version, AdminStudio application ID, AdminStudio catalog name, AdminStudio catalog ID, and

AdminStudio server name. This updated version of the tag file will be added to the Windows Installer package using a transform file and the original vendor tag file in the original Windows Installer package will be removed by the same transform. If the Windows Installer package is imported into Application Manager after being repackaged by Repackager, there is no need for the transform to remove the original vendor tag file since it is not present.

About Software Tagging RegIDs

A software ID tag file can contain up to three different RegIDs: **Tag Creator RegID**, **Software Creator RegID**, and **Software Licensor RegID**. A RegID has the following format:

```
regid.YYYY-MM.ReversedDomainName,optional_division
```

For example:

```
regid.2009-06.com.yourcompany,GlobalProductDivision
regid.2001-09.com.ABCDcompany,AccountingSystems
regid.2005-11.com.WZYZcompany
regid.2010-02.net.1234company,WordProcessing
```

The following table describes the components of a RegID in more detail:

Table 7-12 • Components of a Software Tag File RegID

Component	Description
Type	A software tag ID begins with the string **regid**. This qualifies the element as a registration ID for software identification tags. The **regid** string is followed by a period.
Date	Enter a date code in YYYY-MM format which is the date during which the naming entity owned the domain, such as **2009-11**. • This date should be the first month in which the domain name was owned by this naming entity at 00:01 GMT of the first day of the month. • This date code uses the Gregorian calendar and must include all four digits of the year and both digits of the month (where January = 01 and December = 12). • The dash must be included. The date string is followed by a period.

Table 7-12 • Components of a Software Tag File RegID

Component	Description
Naming Authority	Enter the reversed domain name of the naming entity (person or organization) that is creating this software identification tag. For example, if your company's domain is `mycompany.com`, you would enter **`com.mycompany`** after the date code. Note the following regarding the naming authority domain name: • A RegID can be created by any individual or organization that owns or has owned the registration for a domain name. • The domain name does not need to be active, nor does it need to resolve to an address. • Domain names by themselves do not constitute a unique identifier because domains can expire and/or be acquired by other entities. This means that the RegID must also include a date indicating when the domain registration was owned by the entity.
Additional Subentity	(Optional) Enter a string that specifies any subentity that may have its own unique naming authority, such as **`AccountingSystems`** or **`MarketingDepartment`**. For example, AdminStudio's default RegID includes AdminStudio as a subentity of Flexera, which is the naming authority: `regid.2009-06.com.flexera,AdminStudio` Using this optional component enables individual business units of large software publishers to manage their own software identification tags independently. This string must be preceded by a comma. ***Note*** *• With the exception of the comma prefix, the owner of the domain name can assign text following the reversed domain name as desired as long as all characters are valid for use in file names on any platform that the tag will be installed on. It is the responsibility of the naming entity to ensure that each subentity reference is unique within their organization.*

About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields

The **Tag Creator Name**, **Software Creator Name**, and **Software Licensor Name** fields could all refer to the same company, or could refer to different companies. It is entirely possible that one company could create the software (Software Creator Name), while a second company could license and distribute the software (Software Licensor Name), and yet another company could create the tag (Tag Creator Name). Any combination is possible.

When specifying these fields on the **Software Identification Tag** view, keep in mind the following:

- **Software Creator Name and Software Creator RegID**—Should be consistent across all software packages created by a company.
- **Software Licensor Name and Software Licensor RegID**—Should be consistent across all software packages licensed by a company.
- **Tag Creator Name and Tag Creator RegID**—Should be consistent across all software identification tags created by the company.

Note • *If the captured installation data for this application does not install any Add/Remove Programs entries (meaning that it will not be displayed on a machine's list of installed programs), the following message will be displayed on the* ***Software Identification Tag*** *view:*

```
No software ID tags will be created because no Add or Remove Program information was detected in the
    captured data.
```

However, if you build this project into a Windows Installer package and then import that package into Application Manager, a software tag file will be created.

Viewing and Editing Package Tag Information in Application Manager

You can view and edit the software ID tag information for an individual Windows Installer package on the **Software Identification Tag** subtab of the **Home Deployment Type View** in Application Manager.

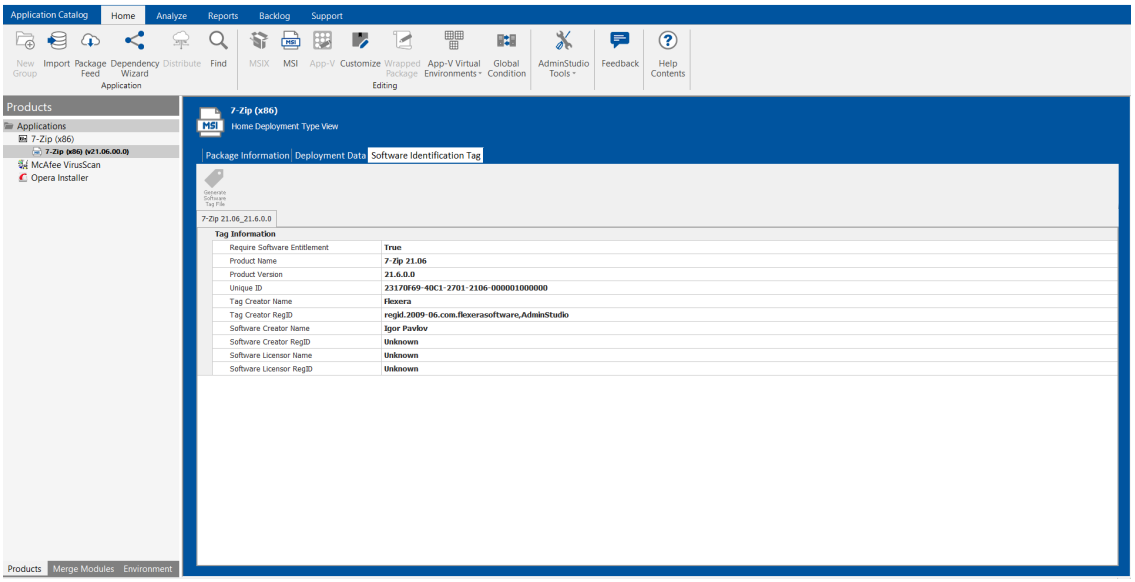


Figure 7-9: Software Identification Tag Subtab of Home Deployment Type View

Task ***To view and edit software ID tag file information for a Windows Installer package:***

1. Open Application Manager.
2. Select the **Home** tab in the ribbon.
3. In the Application Manager tree, select a Windows Installer package. The **Home Deployment Type View** opens.
4. Open the **Software Identification Tag** tab.
5. Review the properties, which are described in Software Identification Tag Tab, and make any desired edits.
6. To save your edits and generate an updated tag file, click the **Generate Software Tag File** button. When this button is clicked, Application Manager will generate a new transform file for this package that includes the newly modified tag file, and will then reimport the package into the Application Catalog along with its updated transform file.

Viewing Bundled Packages of Complex Installer Executables

You can import complex installer executable files that contain bundled Windows Installer packages into the Application Catalog. There are multiple installation executable types that can contain embedded Windows Installer packages, including the following:

- InstallShield InstallScript `.exe` files
- InstallShield Basic MSI installers that are compressed into a `setup.exe` file
- InstallShield Suite Installer `.exe` files
- Wise Package Studio `.exe` files
- PowerShell wrapped package `.ps1` files
- Other executable file types that can be uncompressed by 7-ZIP
- InnoSetup

Note • *See Enabling Support for InnoSetup Executables.*

- Nullsoft
- WiseScript
- WiX Burn
- Advanced Installer
- InstallAnywhere

Note • *If AdminStudio is unable to identify the `.exe` type of an imported package, by default it is categorized as a legacy installer.*

After these complex installer executables have been imported, you can view a list of the child `.msi` packages bundled within them on the **Bundled Packages** tab of the **Home Deployment Type View**.

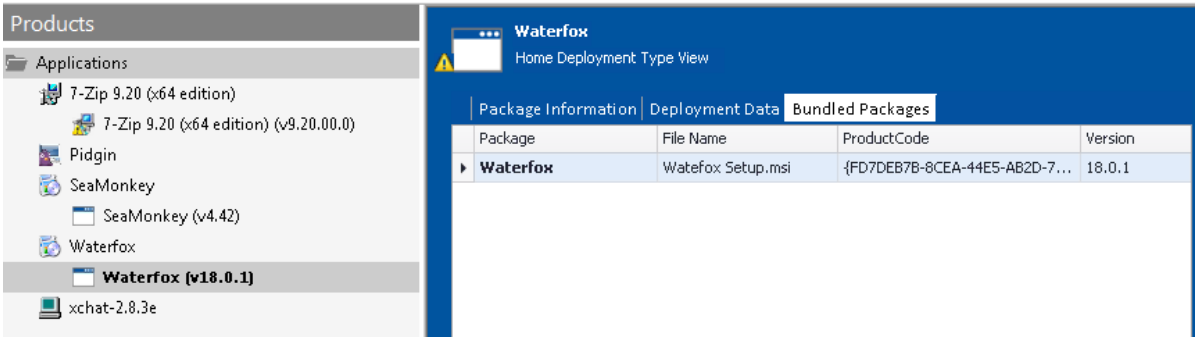


Figure 7-10: Bundled Packages Tab of Home Deployment Type View

When inspecting these child `.msi` packages, Application Manager extracts the information about each package, such as product name and version number. This makes it much more likely that Application Manager will be able to assign a Flexera Identifier to these applications.

You can perform operating system compatibility, application virtualization compatibility, and best practices testing on these bundled packages, and the test results will be combined. For more information, see Viewing Combined Test Results of Bundled Packages.

Note • *AdminStudio will only inspect complex installer (.exe or .ps1) files one level deep. If a complex installer file contains another complex installer file bundled within it, that child file will not be inspected.*

Enabling Support for InnoSetup Executables

AdminStudio supports the latest version of InnoSetup: `innounp.exe` (version 0.46.0.0).

In order for AdminStudio to extract child packages from InnoSetup executables, you need to perform the following steps.

Task ***To enable support for InnoSetup executables:***

1. Download the **InnoSetup Unpacker** RAR file (`innounp046.rar`) from the following location:

   `https://sourceforge.net/projects/innounp/files/innounp/innounp%200.46/innounp046.rar/download`

2. Uncompress the `innounp046.rar` file to obtain the `innounp.exe file`.

3. Copy the `innounp.exe` file to the following location on your AdminStudio machine:

   `C:\Program Files (x86)\AdminStudio\2016\Common\Tools`

Using Package Automation

The Package Automation feature enables you to automate the import, testing, wrapping, conversion, and publishing of packages. The following section describes how to use the Package Automation features:

- Adding a Single Package Request to the Backlog Tab
- Importing a List of Applications for Automation
- Matching the Applications
- Configuring Actions for Automation
- Scheduling Automation
- Specifying Notification Settings for Automation
- Automate Package Feed Module Trial Packages

Adding a Single Package Request to the Backlog Tab

To add a single package request to the Backlog tab, perform the following steps.

Task ***To add a single package request:***

1. Click **Add** on the **Backlog** tab.

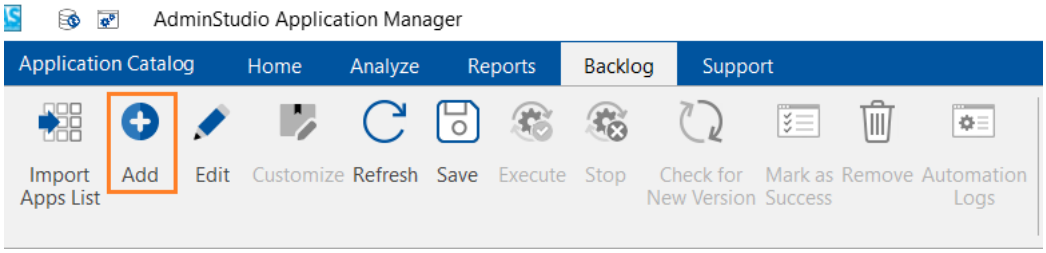


Figure 7-11: Click Add Ribbon

The **Add Package Request** panel opens.

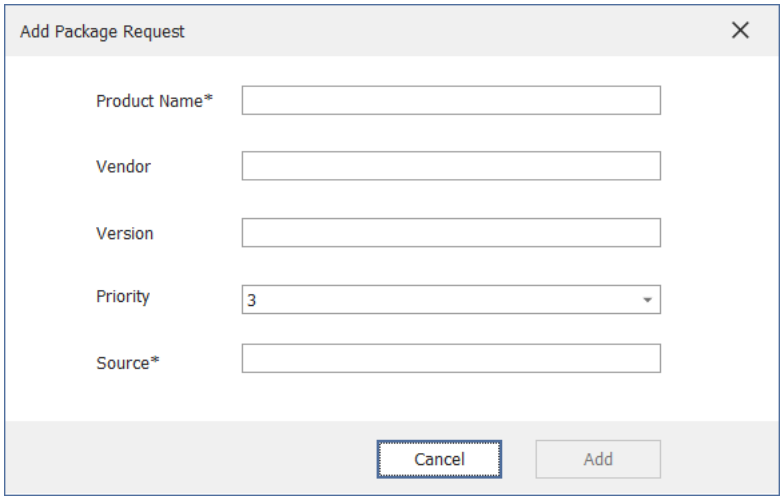


Figure 7-12: Add Package Request Panel

2. Enter the following information:

Property	Description
Product Name	Enter the name of the application for the package request. This is a mandatory field.
Vendor	Enter the name of application vendor for the package request.
Version	Enter the application version for the package request.
Priority	Select the priority from the drop down for the package request. ***Note*** • *The default value of the Priority will be 3.*
Source	Enter source for the package request. This is a mandatory field.

3. Click **Add** to add a single package request in the Backlog tab.

4. Click **Cancel** to close the **Add Package Request** dialog box without saving your changes.

Editing a Package Request in the Backlog Tab

To edit existing package request, perform the following steps.

Task ***To edit an existing package request:***

1. Open **Backlog** tab.
2. Select the package request that you want to edit.
3. Click **Edit** ribbon button or right-click on the package and select **Edit** from the context menu.

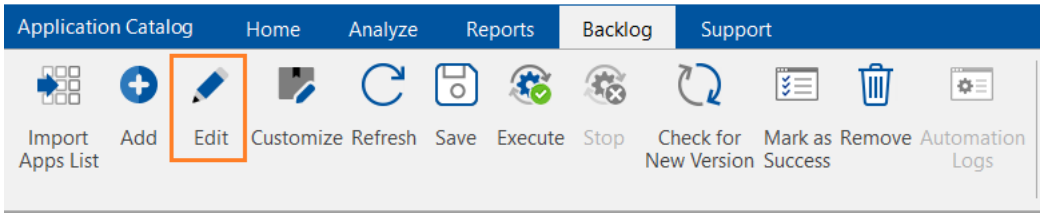


Figure 7-13: Click Edit Ribbon

The **Edit Package Request** panel appears.

4. Modify as necessary and click **Apply**.

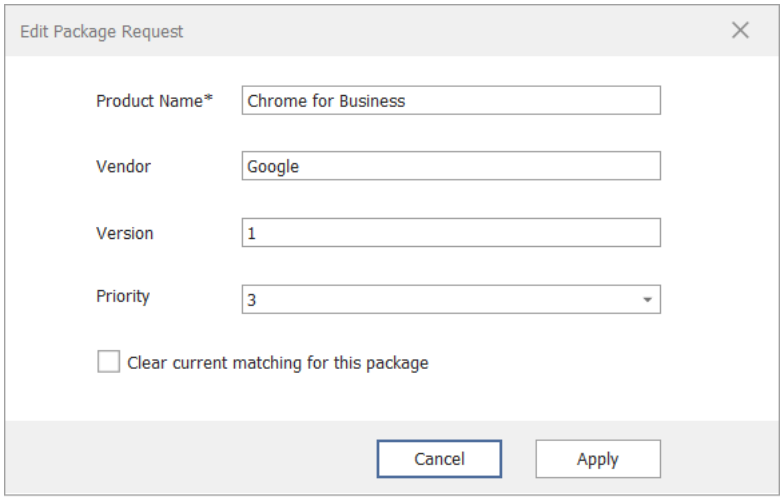


Figure 7-14: Edit Package Request Panel

5. If you select **Clear current matching for this package** while editing a package request, the current matches for this package in the **Version in Catalog** and **Version in Package Feed** will be cleared, and matching task will be triggered to get new matches based on the modified values.

   If the **Clear current matching for this package** option is not selected, then modified values will be updated to the package request in the **Backlog** tab but the current matches will be retained. By default, this option is not selected.

6. Click **Cancel** to close the **Edit Package Request** dialog box without saving your changes.

Customize Package

Clicking on the **Customize** button launches the **Customization Wizard**. The panels in the **Customization Wizard** are dynamic depending on the number of customization options available for the selected package. Options checked/unchecked in the **Customization Wizard** will be saved. When a new version of the package is executed manually or during automation, all the saved customization options are used to generate a transform (.mst) file and is imported along with the MSI package into the catalog.

To customize a package, for example Google Chrome, perform the following steps.

Task ***To customize a package:***

1. In the **Backlog** tab, select a package request.
2. Click the **Customize** button in the ribbon or right-click and select **Customize** from the context menu.

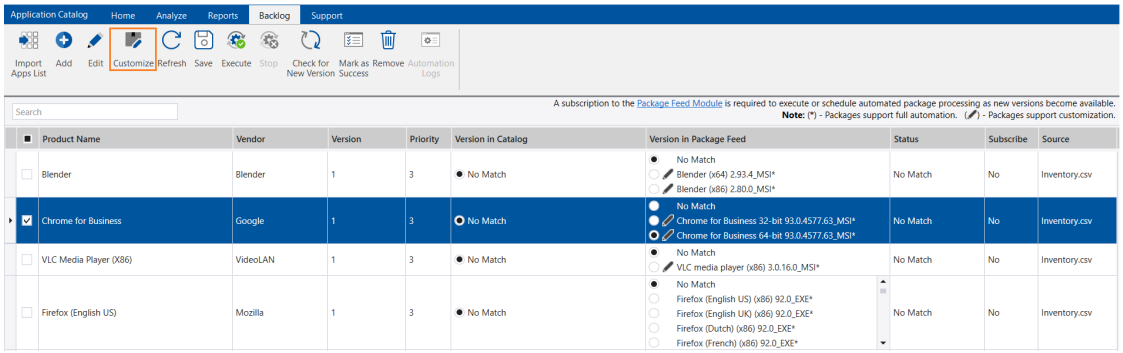


Figure 7-15: Click Customize Ribbon

Note • *You can customize only those packages which have customization options available. You can identify such packages by the ✎ visual indicator.*

3. The Customization Wizard opens. In the Customization Wizard panels check/uncheck your desired customization options for the package. For more information. see Customization Wizard.

Importing a List of Applications for Automation

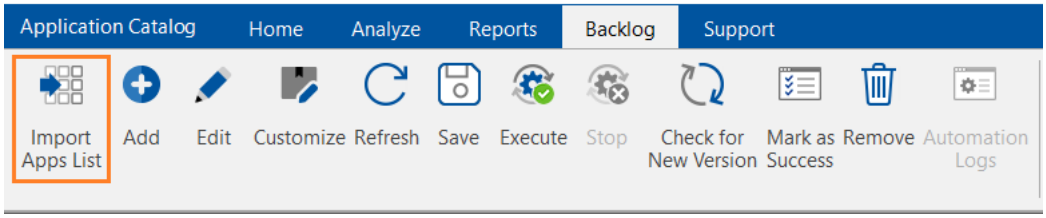
The Application list is a .CSV file, exported from an external inventory system (such as Configuration Manager, FlexNet Manager Suite, or another CMDB system), that contains a list of all of your organization's applications. After successful import, AdminStudio identifies the matching packages in the application catalog and Package Feed Module against the applications in the imported list.

Important • *The apps list is expected to be in the .csv format. This file is expected to have the list of apps deployed in your organization across all the end point devices. Such a file is generally obtained (or exported) from an inventory system like ConfigMgr, Flexera's FlexNet Manager Suite (FNMS) etc.*

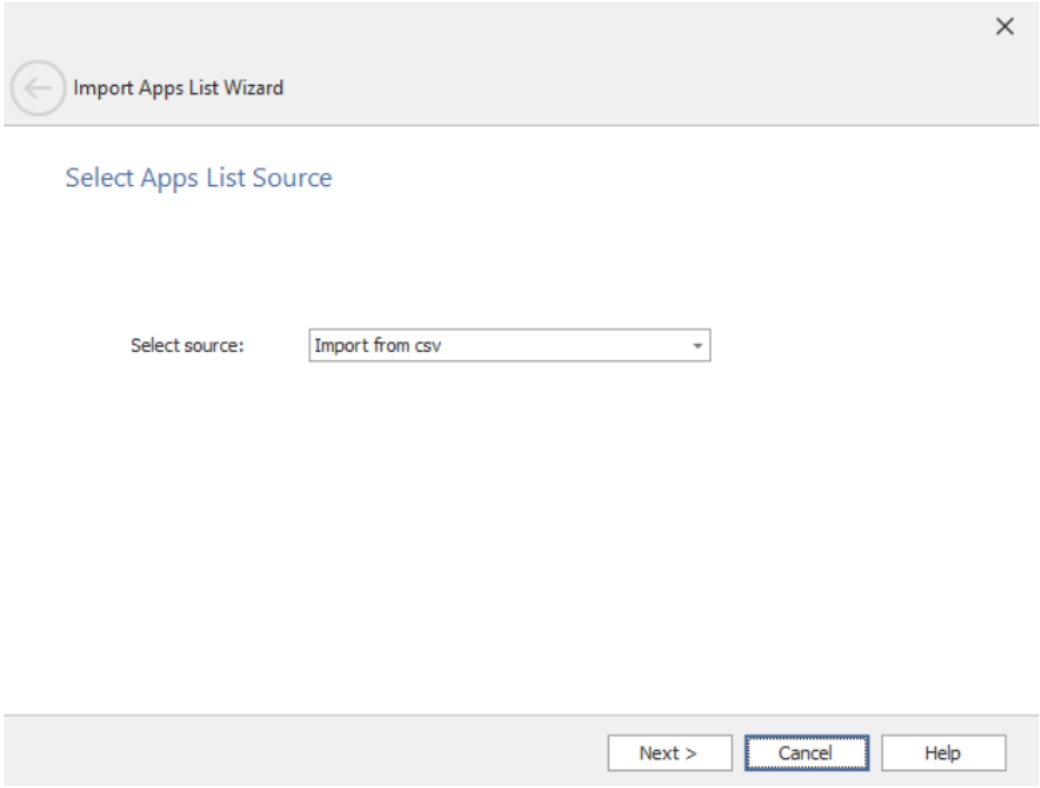
To import application list, perform the following steps:

Task ***To import a list of applications:***

1. Click **Import Apps List** on the **Backlog** tab.

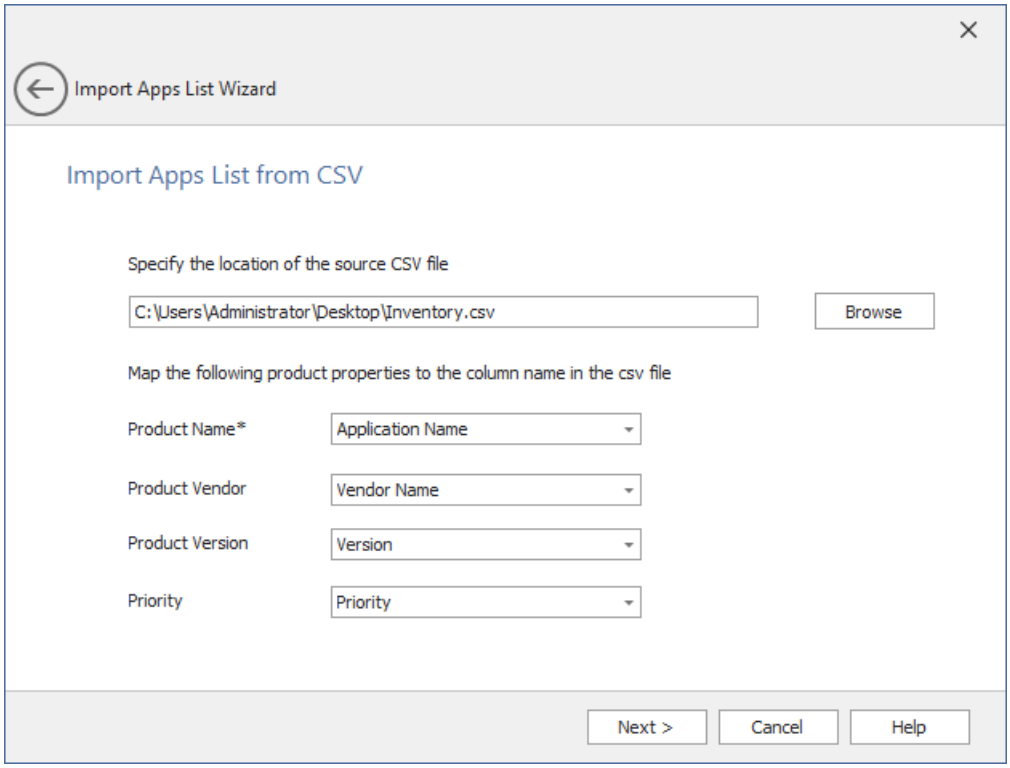


The **Select Apps List Source** panel of the **Import Apps List Wizard** opens.



2. From the **Select source** list, select **Import from csv** and click **Next**. The **Import Apps List from CSV** panel opens.

Note • *In future releases, additional sources will be added to the* ***Select source*** *list.*



3. Click **Browse** and select the CSV file which has the inventory of apps deployed in your organization.

Note • *The CSV file can have any number of columns. However it is mandatory that it has columns representing* ***Product Name****. The* ***Product Vendor****,* ***Product Version****, and* ***Priority*** *are the optional. These four columns will be imported into the AdminStudio Application Catalog.*

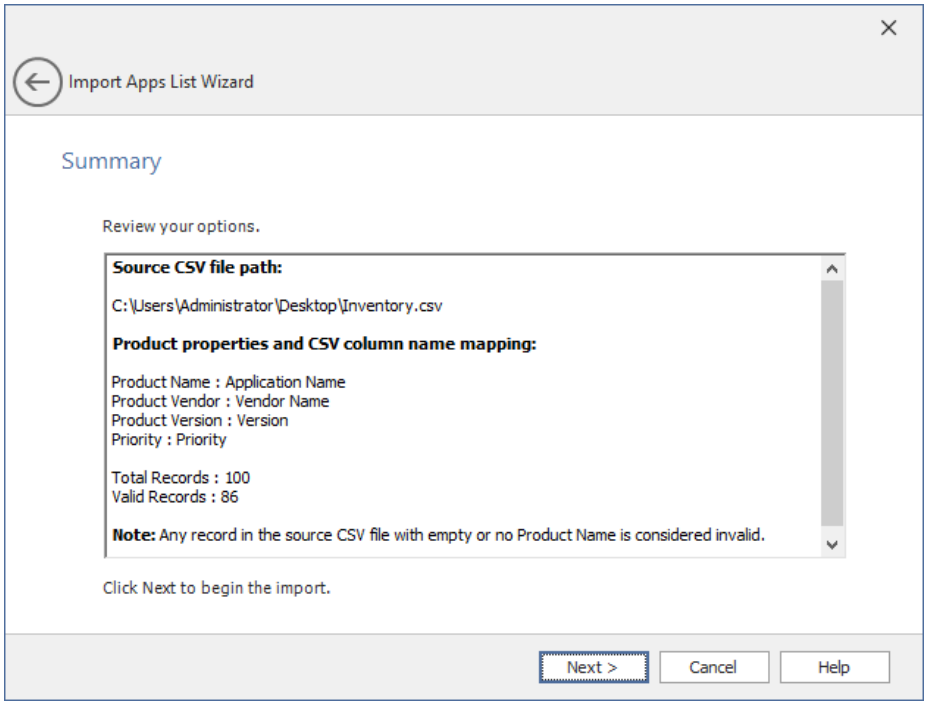
AdminStudio will automatically map the columns from the CSV file to Product Name, Product Vendor, Product Version, and Priority, based on how the columns are named. If the mapping is not done automatically or if it is found to be incorrect, then select the appropriate column name from drop down to map to **Product Name**, **Product Vendor**, **Product Version**, and **Priority**.

For example:

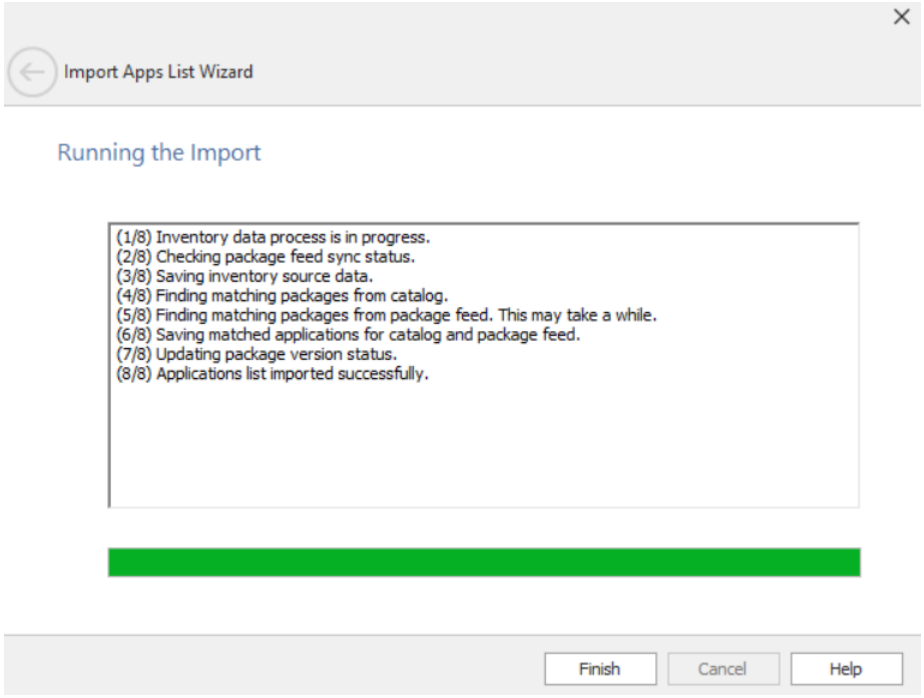
- **Product Name** - Software Title, Application Name, Product, etc.
- **Vendor Name** - Manufacturer, Publisher, or Vendor
- **Product Version** - Software Versions or Version
- **Priority** - Priority

Note • *Any row in the CSV file with blank or empty value for the* ***Product Name*** *will be considered as an invalid record and will not be imported into AdminStudio. Product Vendor Product Version and Priority can be blank or empty. However, it is recommended not to have blank values for Product Name, Vendor and Version to get better matching results. If the value of Priority is empty or invalid, then the default value 3 will be used as a Priority.*

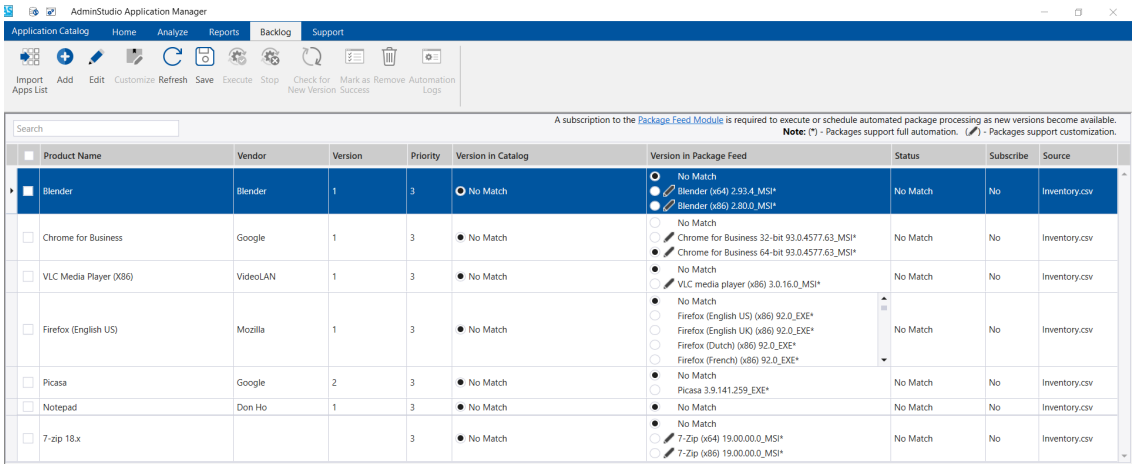
4. Click **Next**. The **Summary** panel opens.



5. Click **Next**. The **Running the Import** panel opens.



6. When import is complete, click **Finish**. Upon successful import, all app entries present in the CSV file will be displayed on the **Backlog** tab.



7. Proceed with the steps in:
   - Matching the Applications
   - Configuring Actions for Automation
   - Scheduling Automation

Matching the Applications

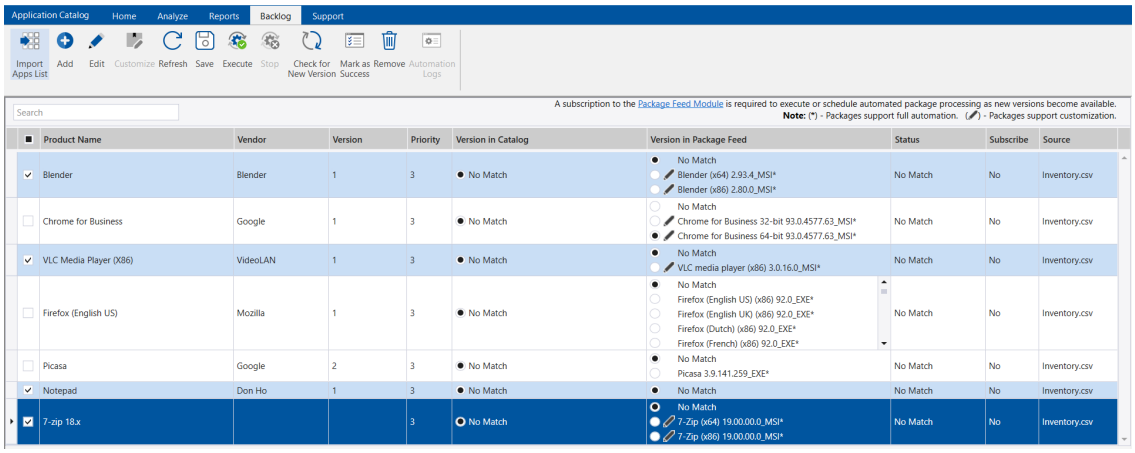
After the applications list is imported into the catalog, AdminStudio performs automatic matching of the applications in the list against packages in catalog and Package Feed Module.

The matching packages in the catalog and Package Feed Module for an application will be displayed in the **Version in Catalog** and **Version in Package Feed** columns respectively. For a given application, depending on how close the **Product Name**, **Version** and **Vendor** matches with the packages in the catalog and Package Feed Module, you may see single or multiple matching entries in the **Version in Catalog** and **Version in Package Feed** columns. If there are no matching packages found, then you will find no package entries in these two columns. After import and matching, AdminStudio by default sets all the applications to **No Match** in **Version in Catalog**, **Version in Package Feed**, and **Status** column.

Task ***To match the applications:***

1. For an application, select the appropriate package entry in the **Version in Catalog** and **Version in Package Feed**, if there are more than one entries found.
2. Click **Save** button after the matching is done, to retain the matching. You will notice the **Status** of the applications will be changed after clicking on **Save**.

Note • *When you select any package node under the **Version in Catalog** and **Version in Package Feed** columns and click **Save** then all other nodes will hide except the selected package node.*



The following information is listed for each application:

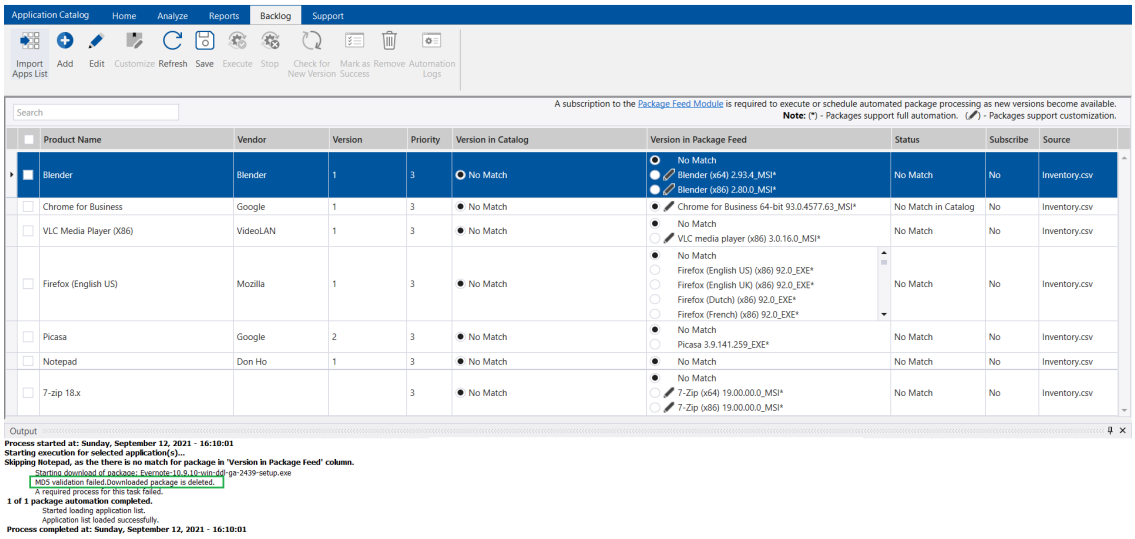
Column	Description
Product Name	Name of application.
Vendor	Vendor of application.
Version	Version of application.
Priority	Priority of application. This column has 5 values, were you can select the value as per the priority of the package. ***Note*** • *The default value of the Priority will be 3.*
Version in Catalog	This column displays the matching package entries in the Application Catalog for the applications in the list. Depending on AdminStudio's automatic matching, this column may have single or multiple matching entries. By default, the matching will be set to **No Match** for all the applications. If no matching packages are found in the catalog, then the column will list no packages. Supported package types include .exe, .msi, and .ps1. The package entry is displayed in the format: ProductName_Version_SetupType, to help easy matching. Example: 7-Zip_19.00.00_MSI. This column has two filter values: • **Matches available**: selects the inventory package having at least one matching package in Version in Catalog column. • **Matches not available**: selects the inventory package having which has no matching package in Version in Catalog column.

Column	Description
Version in Package Feed	This column displays the matching package entries in the Package Feed Module for the applications in the list. Depending on AdminStudio's automatic matching, this column may have single or multiple matching entries. The versions of the packages displayed will be always equal to or greater than the version of the listed application. This helps you know if there is a newer (and latest) version available in Package Feed Module. By default, the matching will be set to **No Match** for all the applications. If no matching packages are found in the Package Feed Module, then the column will list no packages. Supported package types include .exe and .msi. The package entry is displayed in the format: ProductName_Version_SetupType, to help easy matching. Example: 7-Zip_19.00.00_MSI_IgorPavlov. This column has two filter values: ● **Matches available**: selects the inventory package having at least one matching package in the **Version in Package Feed** column. ● **Matches not available**: selects the inventory package having which has no matching package in the **Version in Package Feed** column.

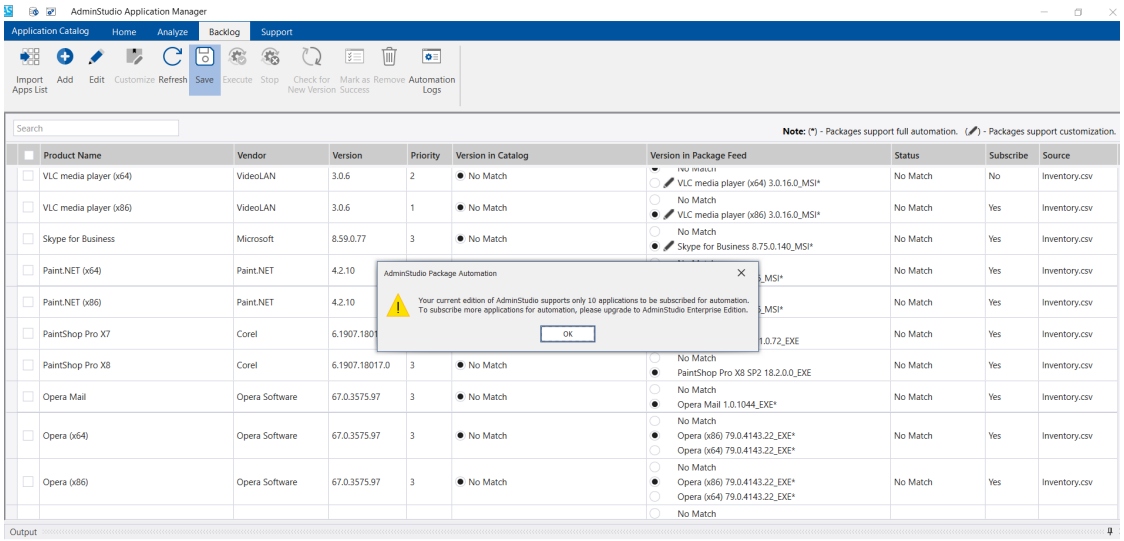
Column	Description
Status	After the apps list is imported into Application Catalog, AdminStudio assigns **No Match** status to all the applications by default. Select the matching entry in the **Version in Catalog** and **Version in Package Feed** columns and click on **Save**. Upon clicking on **Save**, AdminStudio will automatically change the status of the applications based on the selections made in **Version in Catalog** and **Version in Package Feed** columns. Here is the of all the possible status of an application: • **No Match**—Indicates **No Match** is selected, for an application, in both the **Version in Catalog** and **Version in Package Feed** columns. • **No Match in Catalog**—Indicates an appropriate matching package entry is found and selected in the **Version in Package Feed** column (Package Feed Modules packages) but there is no matching package entry found and No Match is selected in the Version in Catalog column. • **No Match in Package Feed**—Indicates an appropriate matching package entry in found and selected in the **Version in Catalog** column but there is no matching package entry found and **No Match** is selected in the **Version in Package Feed** column. • **New Version Available**—A matching package for an application is found and selected in the **Version in Catalog** (in Application Catalog) and **Version in Package Feed** (in Package Feed Module) columns. When the selected package is **Version in Package Feed** has the version newer (or greater) than package entry selected in the **Version in Catalog**, then the status is set to New Version Available. This status indicates that a new version of an application is available in Package Feed Module as compared to the version in catalog. • **Success**—After When a new version of an application is available in Package Feed Module, the next step will be to get this new version in the Application Catalog through various pre-configured actions like import, test etc. Once all the pre-configured actions are executed successfully on the newly available version, then the status of this application will be set to **Success**. If any of the pre-configured actions fail, then the status would be set appropriately. Example if Import action fails, then the status will be Import Failed. ***Note*** *• An application's status, which has matching package entry in* ***Version in Package Feed****, will be reset to New Version Available from its current status, whenever a new version of this application is available in Package Feed Module.*
Subscribe	Package Automation executes a set of pre-configured actions when a new version of a package is available in Package Feed Module. If you want to subscribe any application for automation, then set Subscribe to **Yes**. If **No Match** is selected in the **Version in Package Feed** column, then subscribing an application for the Automation is not allowed.
Source	Lists the name of the imported CSV file that contained this application.

3. Click **Execute** in the ribbon. Importing is initiated and messages are displayed in the **Output** window. When importing is finished, **package automation completed** is displayed in the **Output** window.

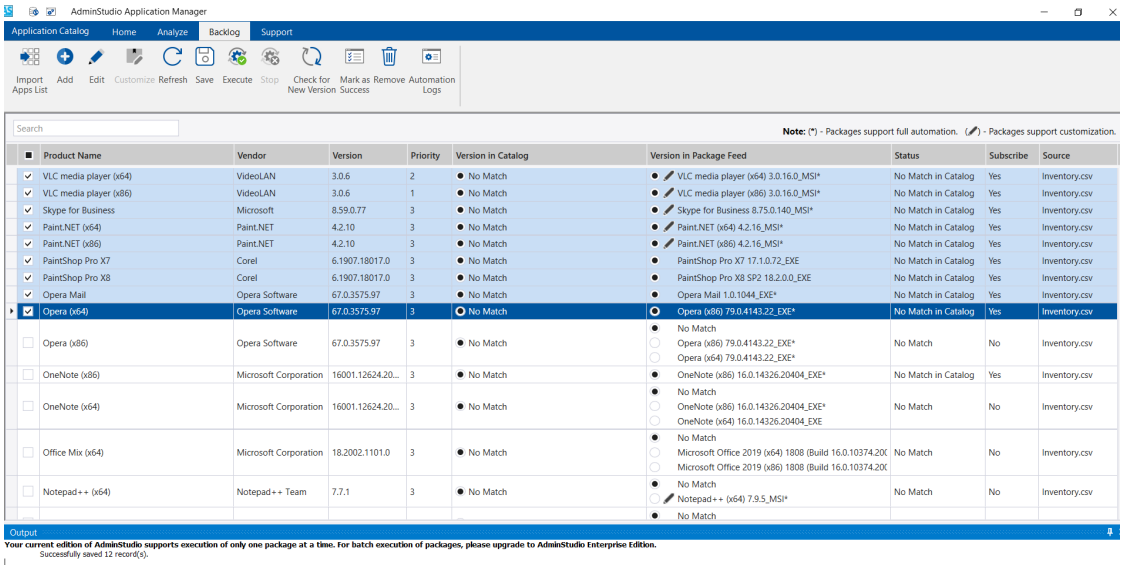
4. Clicking on the **Execute** button, if the downloaded package MD5 checksum value doesn't match with Package Feed Module checksum value then the downloaded package will be deleted from the downloaded location.



Note • *For Professional Licensing, Bulk executions are allowed only for the scheduled automation. Make sure that you can select maximum 10 applications at a time, if not below error message appears.*



Note • *For Professional Licensing, Bulk executions are not allowed for automation.*



Configuring Actions for Automation

Actions are a set of configurable tasks that will be performed on a package during automation. The actions that are available for Package Automation are **Import**, **Test**, **Wrap**, **Convert** and **Publish**. Each of these actions should be pre-configured to achieve automation. You could add only those actions for automation which are needed for your process of creating packages for deployment. For example: You may choose to add **Import**, **Test** and **Publish** actions only for automation. Whenever automation is triggered on the scheduled time, all the actions added for automation will be performed on all those applications which have one of these statuses: **New Version Available**, **No Match in Catalog** and all failed statuses. The same set of pre-configured actions which are added for automation will be executed for a package which is selected for execution in the Backlog tab by clicking on **Execute** button. When all the actions are added for automation, the actions are executed in the following order: **Import**, **Test**, **Wrap**, **Convert** and **Publish**.

- Configure Import Action for Automation
- Configure Test Action for Automation
- Configure Wrap Action for Automation
- Configure Convert Action for Automation
- Configure Publish Action for Automation

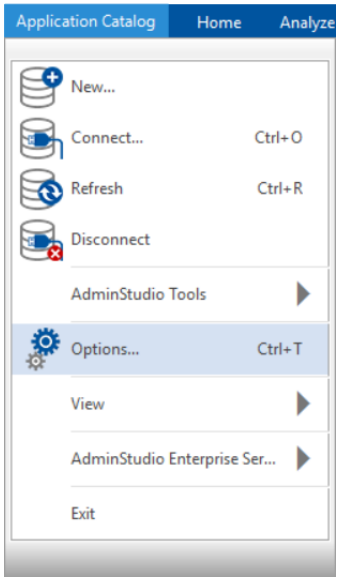
Note • *Import is prerequisite for all other actions therefore it is a mandatory action. This action will be executed as a first step during automation execution.*

Configure Import Action for Automation

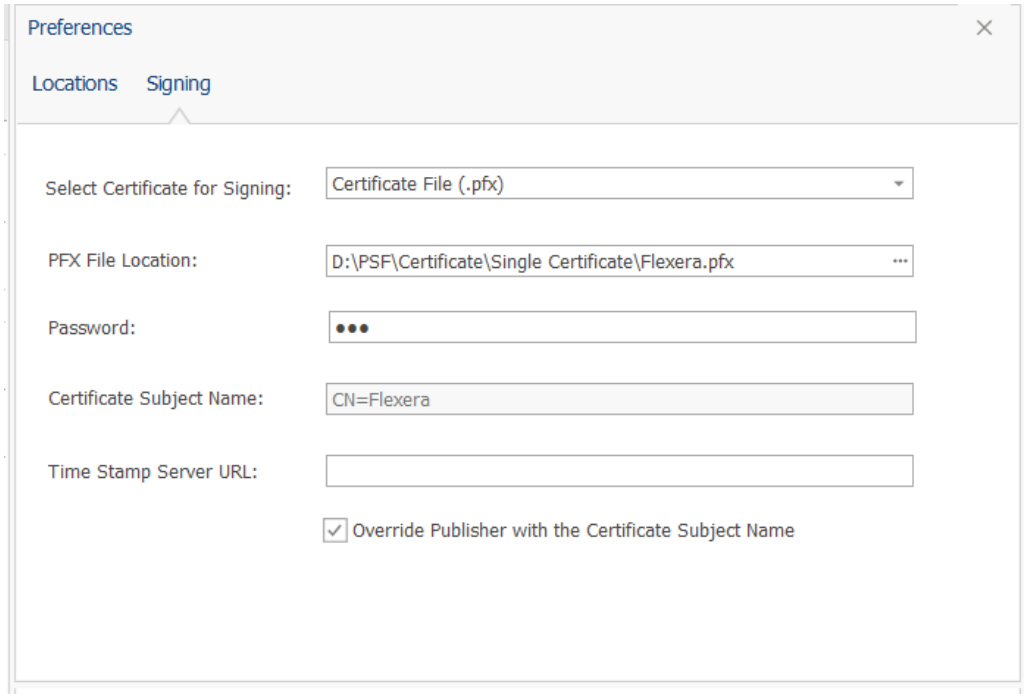
To configure the import actions, perform the following steps:

Task ***To configure import actions:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Configure Actions**. The **Configure Actions** tab opens.



3. In the **Select actions to configure** drop down, select **Import**.

4. Based on the selection, the package will go to the destination group in the Application Manager Catalog. In the **Destination Group** field the possible placeholders are: `[ProductName]\[Version]\[Vendor]\[Platform]\[SetupType]\[Language]`. This order can be rearranged.

For example:

- Applications/Chrome/60.0/Google/x86/MSI/EN.

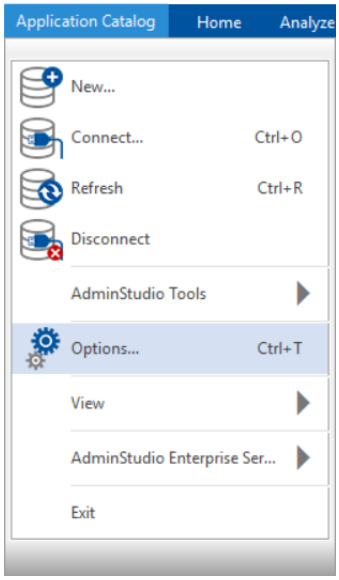
5. Select **Automatically create a custom transform file** check box if you want a transform (.mst) file to be create based on the selections made in the package customization wizard.
6. Select **Include Monitored Directory for Scheduled Automation** check box, if you want the packages in the Monitored Directory to be executed through Scheduled Package Automation.
7. After selecting the above fields, click **OK**.

Configure Test Action for Automation

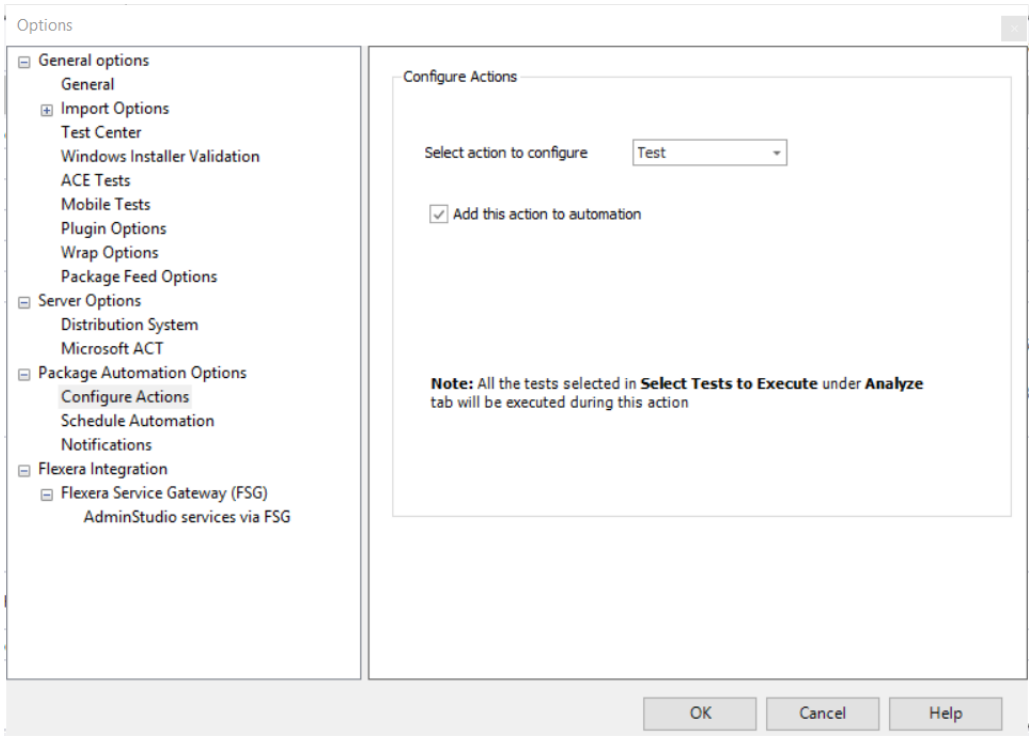
To configure the **Test** action for Automation, perform the following steps.

Task ***To configure the Test action:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Configure Actions**. The **Configure Actions** tab opens.
3. In the **Select actions to configure** drop down, select **Test**.
4. If you want this action to be automated, then select **Add this action to automation** check box.



5. After selecting the above fields, click **OK** to configure the selected **Test** action.

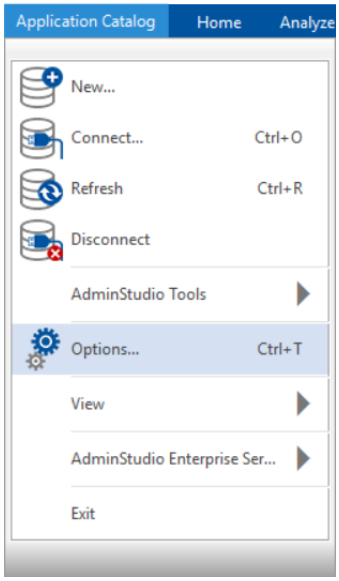
Configure Wrap Action for Automation

You can configure the **Wrap** action for Automation by selecting it and specifying the wrapper type as either **PowerShell** or **EXE** wrapper.

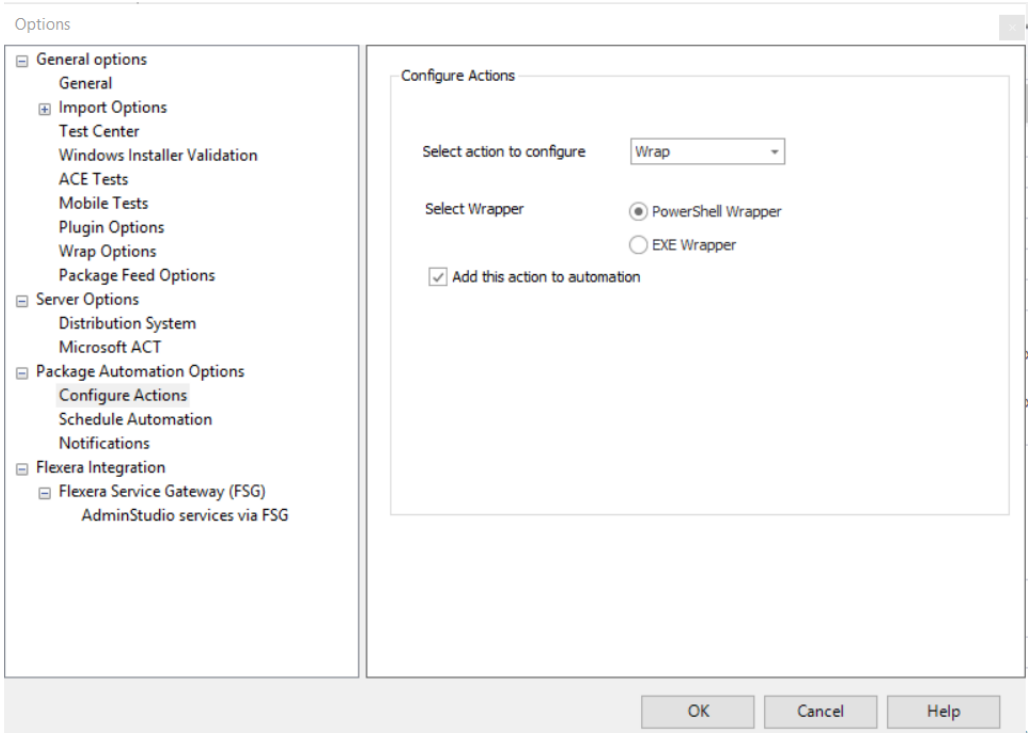
To configure the **Wrap** action, perform the following steps:

Task ***To configure the Wrap action:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Configure Actions**. The **Configure Actions** tab opens.
3. In the **Select actions to configure** drop down, select **Wrap**.



4. If you want to wrap the package in PowerShell Script (.ps1), then select the **PowerShell Wrapper**.

5. If you want to wrap the package in WiseScript (.exe), then select **EXE Wrapper**.
6. If you want this action to be automated, then select **Add this action to automation** check box.
7. After selecting the above fields, click **OK** to configure the selected **Wrap** action.

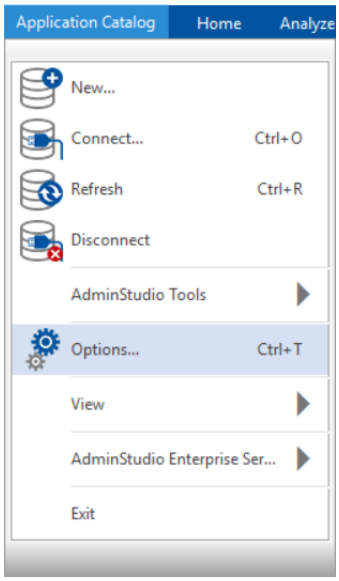
Configure Convert Action for Automation

You can configure the **Convert** action for Automation by selecting it and specifying a target package format., such as MSIX, MSI, App-V.

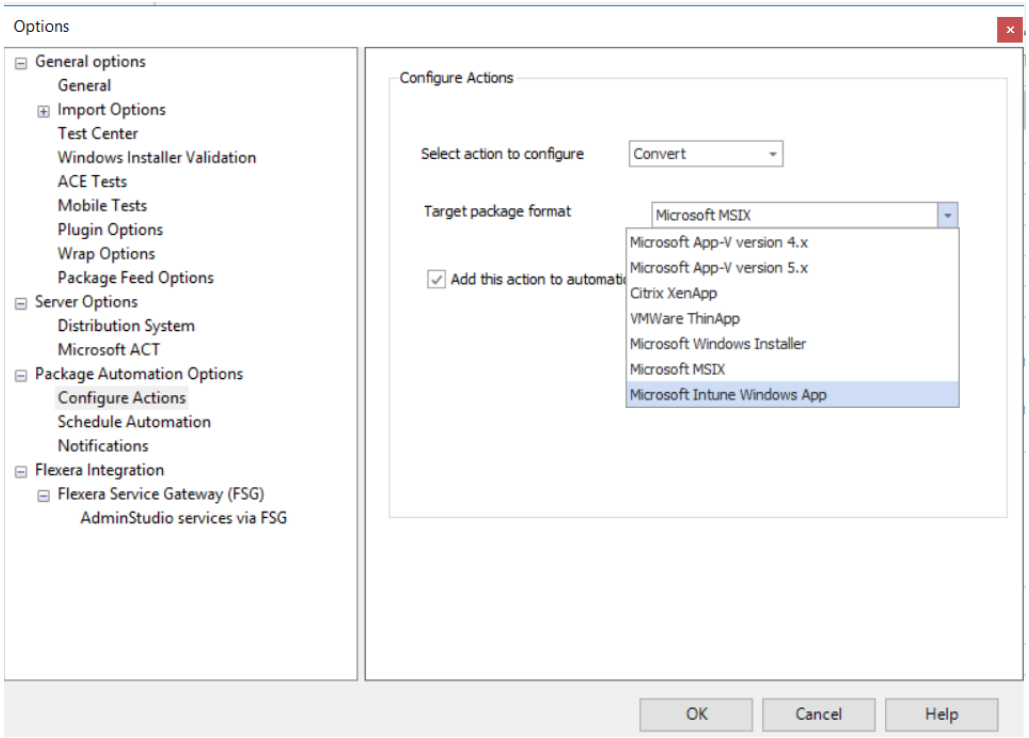
To configure the **Convert** action, perform the following steps:

Task ***To configure the Convert action:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation** Options, select **Configure Actions**. The **Configure Actions** tab opens.
3. In the **Select actions to configure** drop down, select **Convert**.



4. In the **Target Package format** drop down, select the required format.

Note • *The selected target package format will be imported to the Application Catalog after conversion.*

Note • *For Conversions to work you need **AACX** file has to be added in Plugin Options. If you select **Microsoft Intune Windows App** as a Target Package format then this conversion will work with out AACX files in the Plugin Options.*

Note • *For conversions to work silently, the packages should have install common lines.*

5. If you want this action to be automated, then select **Add this action to automation** check box.
6. After selecting the above fields, click **OK** to configure the selected **Convert** action.

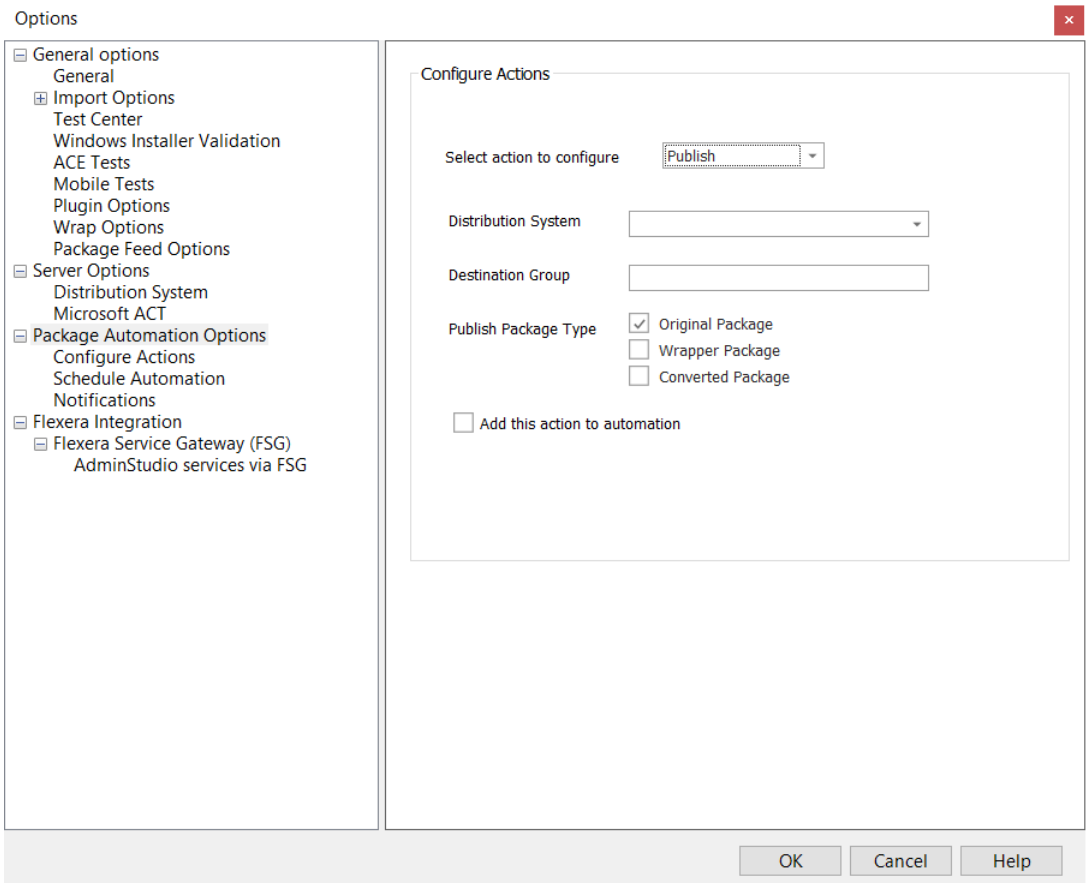
Configure Publish Action for Automation

Publish action distributes the package to the desired Distribution System. To configure Publish action, perform the following steps:

Task ***To configure the Publish action:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Package Automation Options**, select **Configure Actions**. The **Configure Actions** tab opens.

3. In the **Select actions to configure** drop down, select **Publish**.



4. Distribution systems configured in **Server Options** > **Distribution System** will appear in the **Distribution System** drop down, select the a required option from the list.

5. Enter the destination group in the **Destination Group** text box.

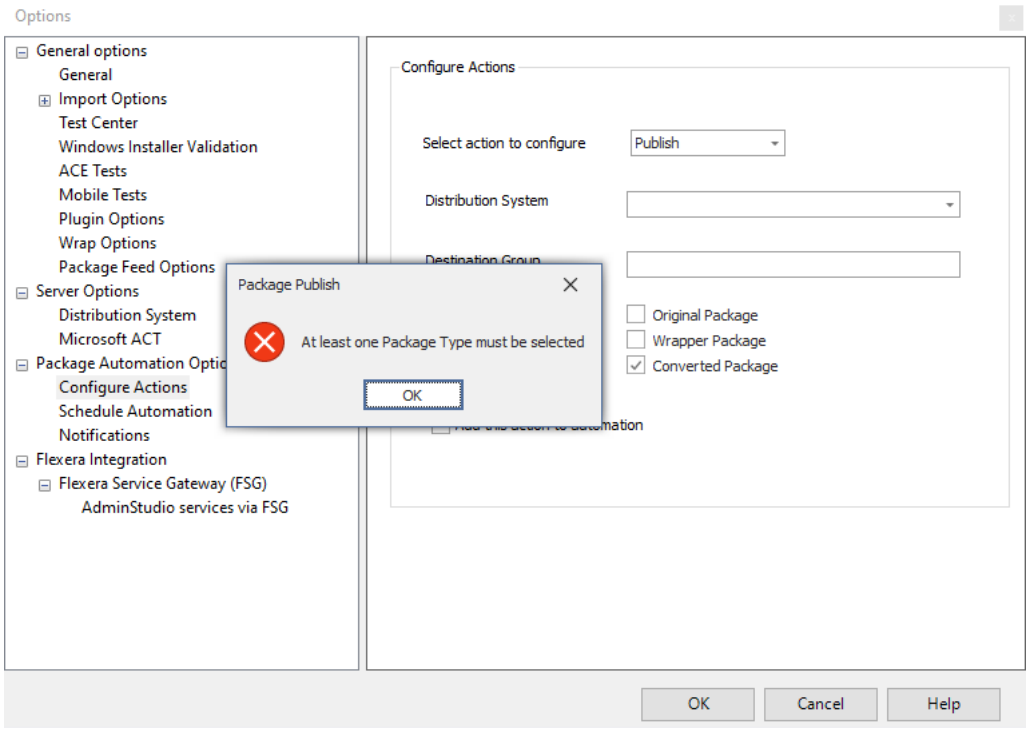
   Example: ConfigMgr Destination Group - Applications/Distribute.

   Where Applications is the primary node in ConfigMgr and Distribute is a folder under that.

6. You can publish the package in the following package types:

   - Original Package - Publishes the original package. It will be selected by default.
   - Wrapper Package - Publishes the wrapper package
   - Converted Package - Publishes the converted package

   At least one **Publish Package Type** should be selected for publish type. Otherwise error pop up will be shown to select at least single publish package type. Previous value selection will be retained.



7. If you want this action to be automated, then select Add this action to automation check box.
8. After selecting the above fields, click **OK** to configure the selected **Publish** action.

Note • *If PowerShell wrapping is selected for automation and MSIX conversion format is selected under conversion then conversion will fail from PowerShell (ps1) to MSIX format as it is not supported.*

Scheduling Automation

In Application Manager, the **Schedule Automation** options enables you to schedule the automation for the following frequency:

- Scheduling Automation for Once
- Scheduling Daily Automation
- Scheduling Weekly Automation
- Scheduling Monthly Automation

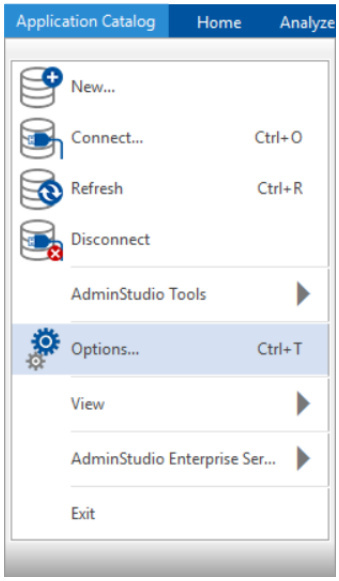
Note • *Applications scheduled for the Automation will be processed based on the **Priority** column value.*

Scheduling Automation for Once

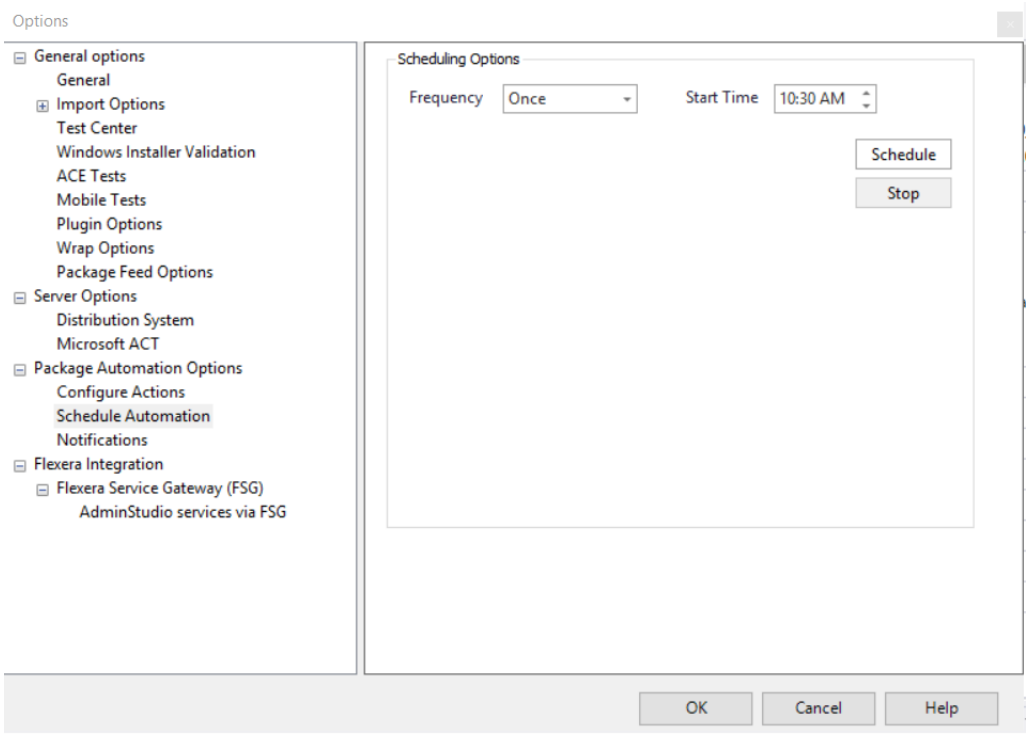
The automation will be triggered only once at the pre-configured time. To schedule **Once** frequency, perform the following steps:

Task ***To configure Schedule Once frequency:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Schedule Automation**. The **Scheduling Options** tab opens.



3. In the **Frequency** drop down, select **Once**.
4. Select the required time in the **Start Time** field.

5. After selecting, click **Schedule** for the automation.

Note • *When Package Feed is in eval mode, the user can schedule the Automation options from the Tool. The Schedule button will be enabled for automation.*

6. To stop the scheduled automation, click **Stop**.

Note • *In the event of Application Manager been closed (owing to any reason), then reconfigure the scheduler.*

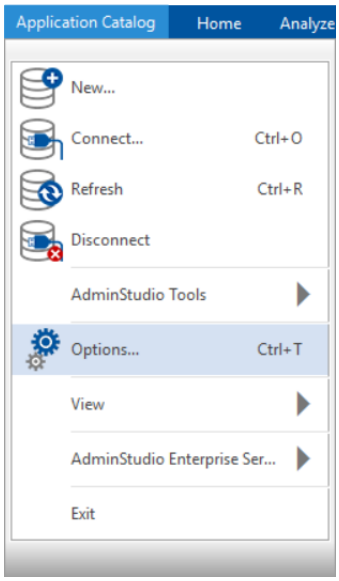
Scheduling Daily Automation

The automation will be triggered daily at the pre-configured time.

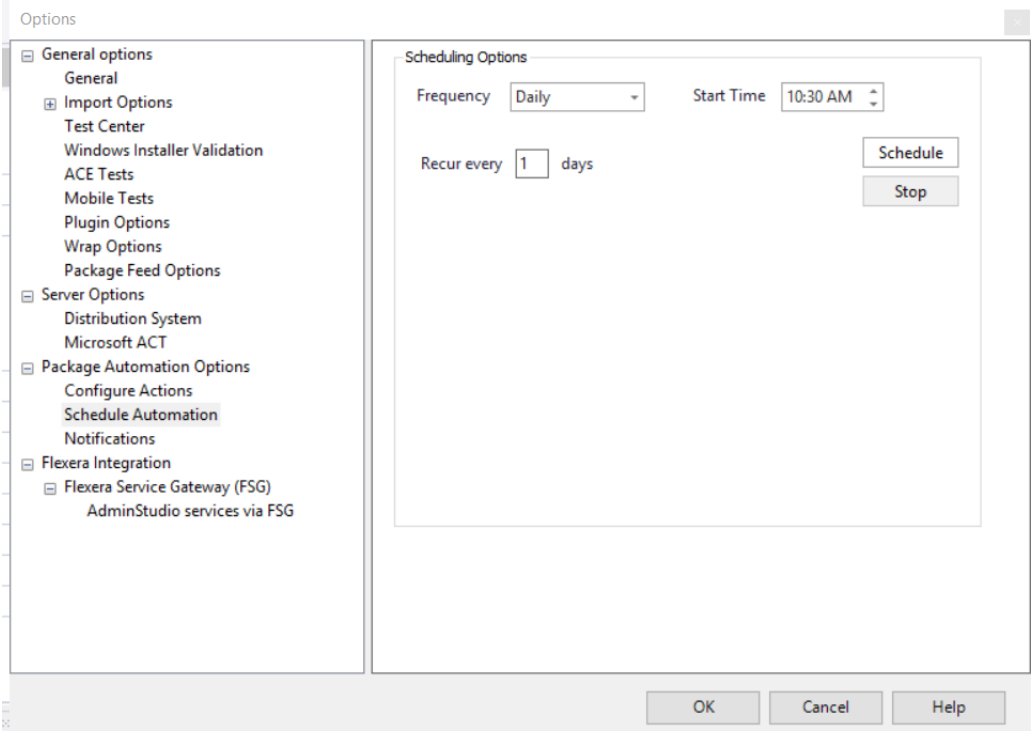
To schedule **Daily** frequency, perform the following steps:

Task ***To Schedule Daily Frequency:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Schedule Automation**. The **Scheduling Options** tab opens.



3. In the **Frequency** drop down, select **Daily**.
4. Select the required time in the **Start Time** field.
5. After selecting, click **Schedule** for the automation.
6. To stop the scheduled automation, click **Stop**.

Note • *In the event of Application Manager been closed (owing to any reason), then reconfigure the scheduler.*

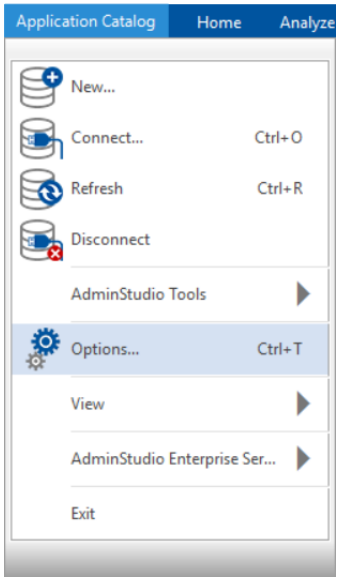
Scheduling Weekly Automation

When selecting **Weekly** frequency, the automation will be trigged weekly at the pre-configured time.

To schedule **Weekly** frequency, perform the following steps:

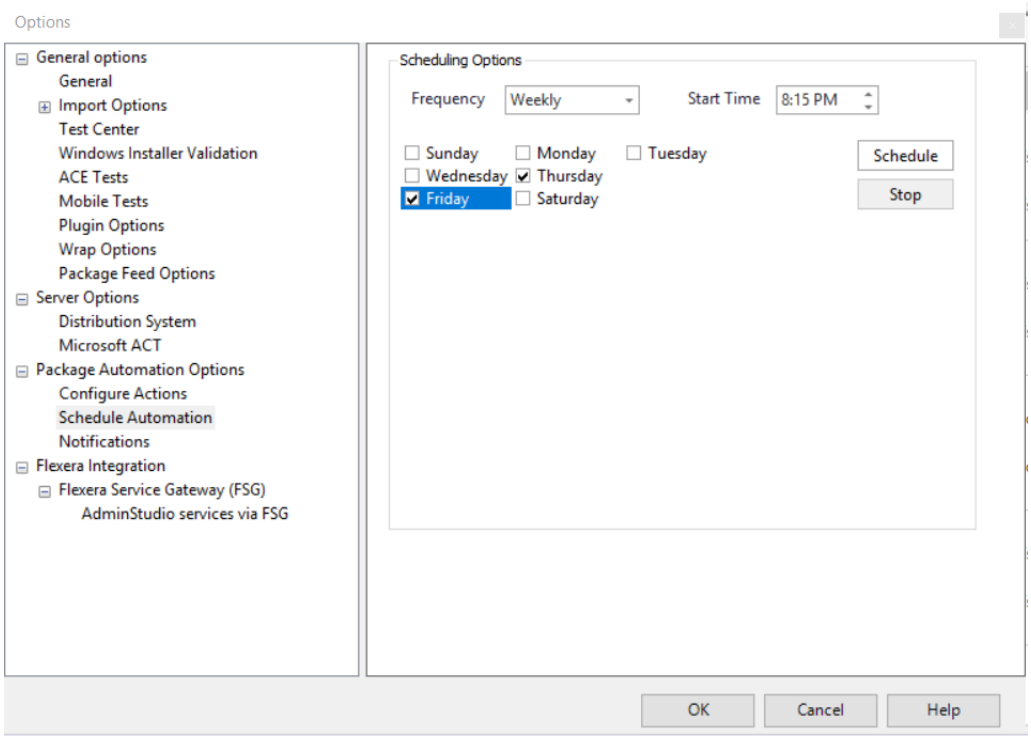
Task ***To Schedule Weekly Frequency:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Schedule Automation**. The **Scheduling Options** tab opens.
3. In the **Frequency** drop down, select **Weekly**. In the below options select the required day (by clicking the check box) of the week.

Note • *Multiple selections are allowed.*



4. Select the required time in the **Start Time** field.
5. After selecting, click **Schedule** for the automation.
6. To stop the scheduled automation, click **Stop**.

Note • *In the event of Application Manager been closed (owing to any reason), then reconfigure the scheduler.*

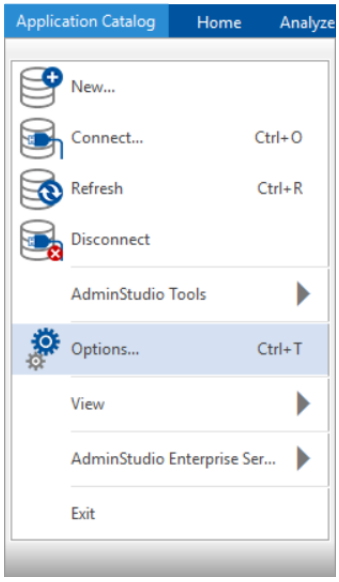
Scheduling Monthly Automation

When selecting monthly frequency, the automation will be trigged monthly at the pre-configured time.

To schedule **Monthly** frequency, perform the following steps:

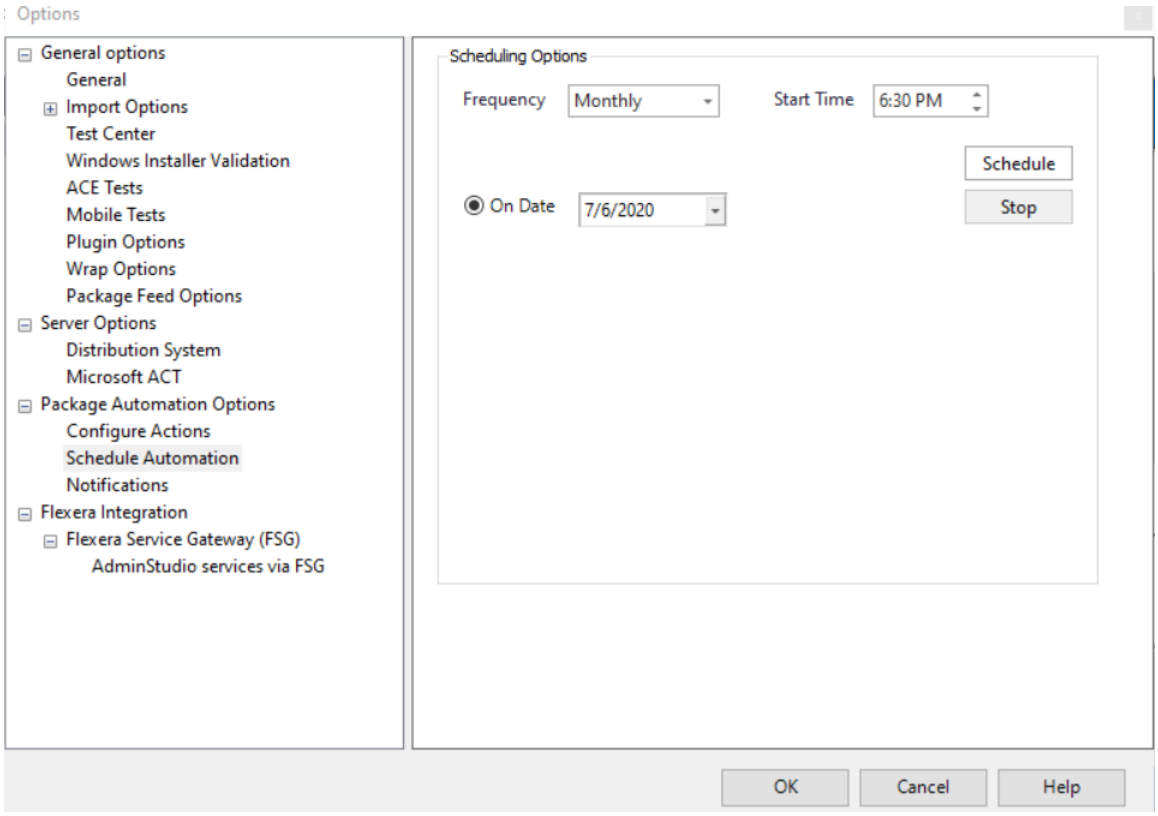
Task ***To schedule Monthly frequency:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Schedule Automation**. The **Scheduling Options** tab opens.
3. In the **Frequency** drop down, select **Monthly**.
4. Select required time in the **Start Time** field.
5. To select specific date of the month, select **On Date** radio button, and then select the required date from the calendar list (by clicking on the date drop down).

   For example: Every month 30th (as selected) on start time the automation will be triggered.



6. After selecting, click **Schedule** for the automation.
7. To stop the scheduled automation, click **Stop**.

Note • *In the event of Application Manager been closed (owing to any reason), then reconfigure the scheduler.*

Specifying Notification Settings for Automation

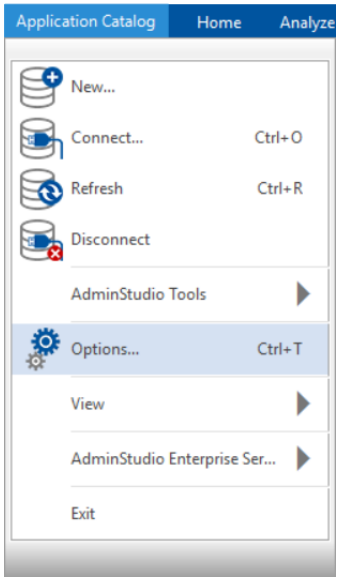
E-mail notifications will be sent to specific user(s) every time:

- A scheduled automation is completed
- A new version for a subscribed package is available in Package Feed

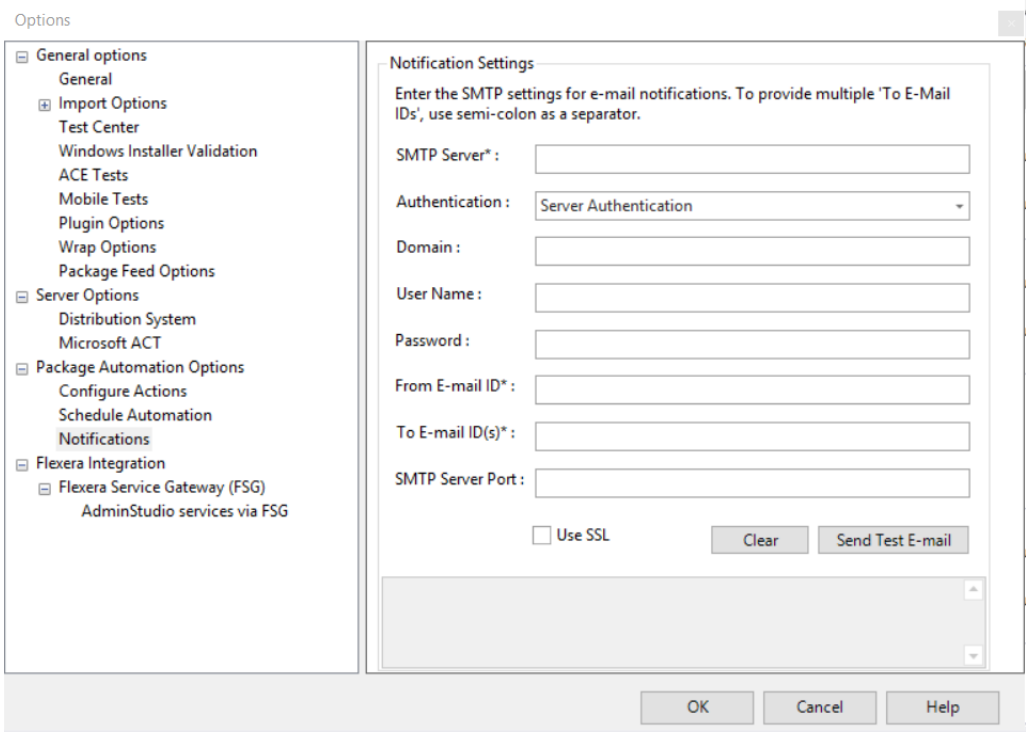
To set Notification options, perform the following steps:

Task ***To set Notifications options:***

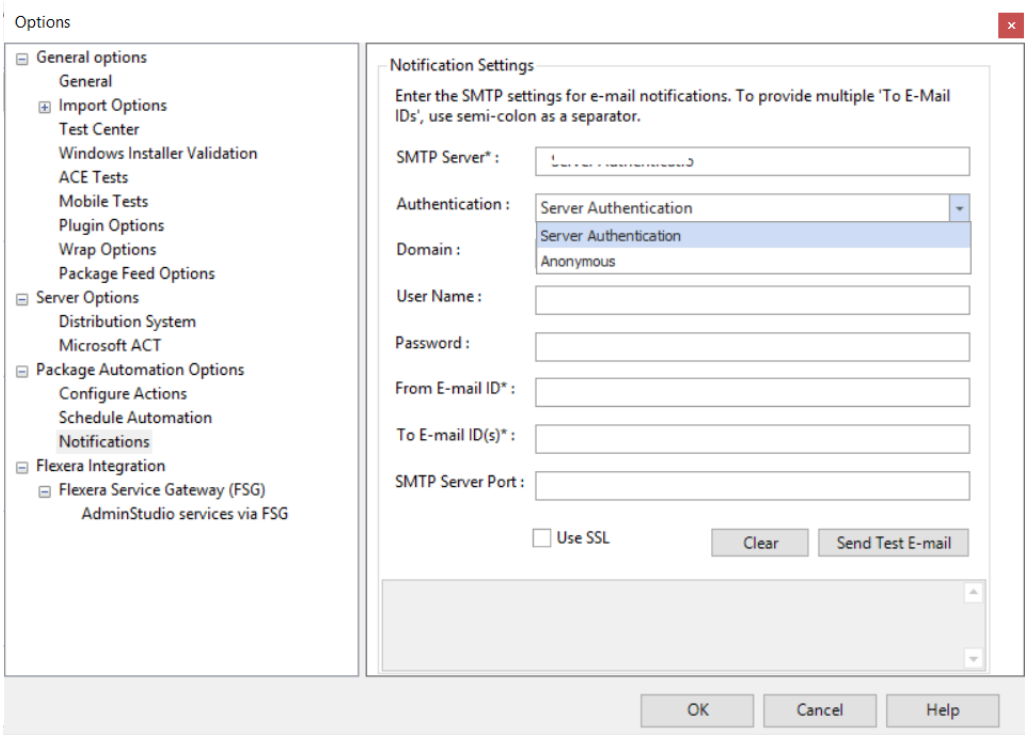
1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



2. Under **Package Automation Options**, select **Notifications**. The **Notification Settings** tab opens.



3. Enter the **SMTP Server** setting in the text box.
4. Click the **Authentication** drop down and set to either **Server Authentication** or **Anonymous**.



5. Enter the **Domain**, **User Name**, and **Password** in the respective text fields.
6. Enter the email address of the sender.
7. Enter email address of the recipients.

Note • *Multiple email address can be added separated by semi-colons.*

Note • *Notifications will be sent to the users whenever there is a new version available for the packages in the Application catalog.*

8. Enter **SMTP Server Port** in the text field.
9. After entering all the details, click **Send Test E-mail** to send a test e-mail to **To E-mail ID(s)** to make sure that the email addresses are valid, and then click **OK**.

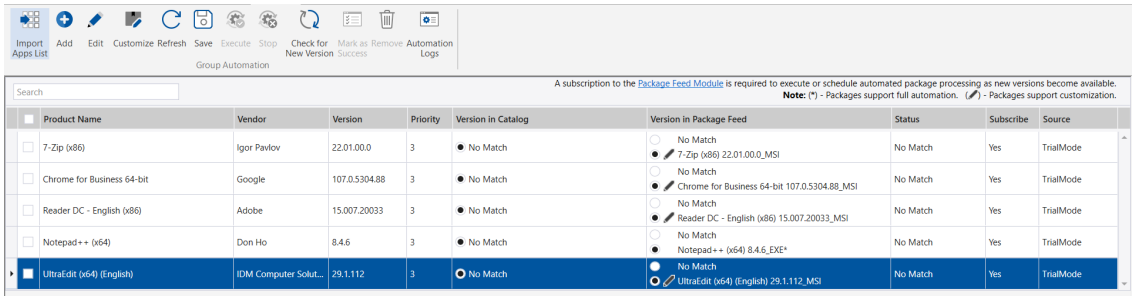
Automate Package Feed Module Trial Packages

The Backlog tab by default is populated with the Package Feed Module trial packages. You can subscribe these packages for automation by configuring actions and to run on a schedule to automatically download and process a package when a new version of the application is available in Package Feed Module.

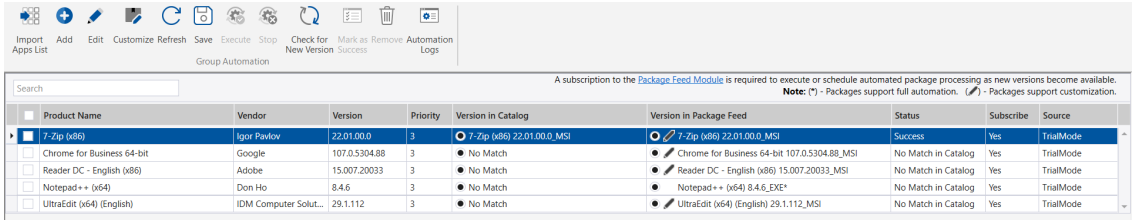
Task ***To automate Package Feed Module trial packages:***

1. Launch AdminStudio.
2. Navigate to **Backlog** tab. Package Feed Module trial packages will be auto-populated.
3. For each package entry in the Backlog tab, select a matching package from the **Version in Package Feed** column and **Version in Catalog** column, if the package is already imported into AdminStudio catalog.

Note • *All the Package Feed Module trial packages are by default subscribed for automation by setting the* ***Subscribe*** *column to* ***Yes****. If you wish to change, set the* ***Subscribe*** *column to* ***No****.*



4. Click **Save** button in the ribbon, to retain the changes.
5. If you wish to process the trial packages right away, select the packages and click the **Execute** button in the ribbon.
6. Upon successful execution, the status of the package will change to **Success**, which indicates the package version in your catalog now matches with the latest package version in the Package Feed Module.



Note • *The user can perform Scheduled Automation for the subscribed packages.*

Support for Publishing to Microsoft Intune and Converting to Intune Package Format

AdminStudio supports Intune for Conversion Wizard and Distribution System.

- Publishing to a Microsoft Intune Distribution System
- Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard

Publishing to a Microsoft Intune Distribution System

AdminStudio supports publishing of MSI and MSIX package to the **Microsoft Intune**. To publish MSI and MSIX format packages to Microsoft Intune, perform the following processes:

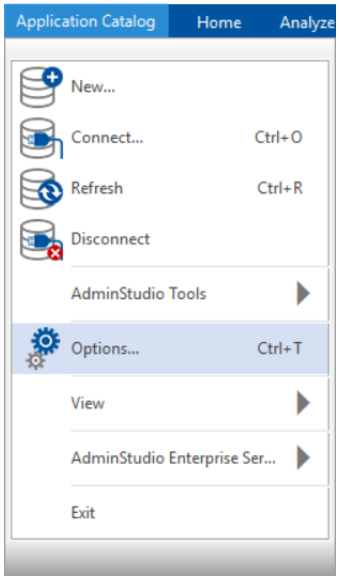
- Setting up Intune Connection for the Distribution System
- Setting Up the Distribution Properties
- Publishing the Applications

Setting up Intune Connection for the Distribution System

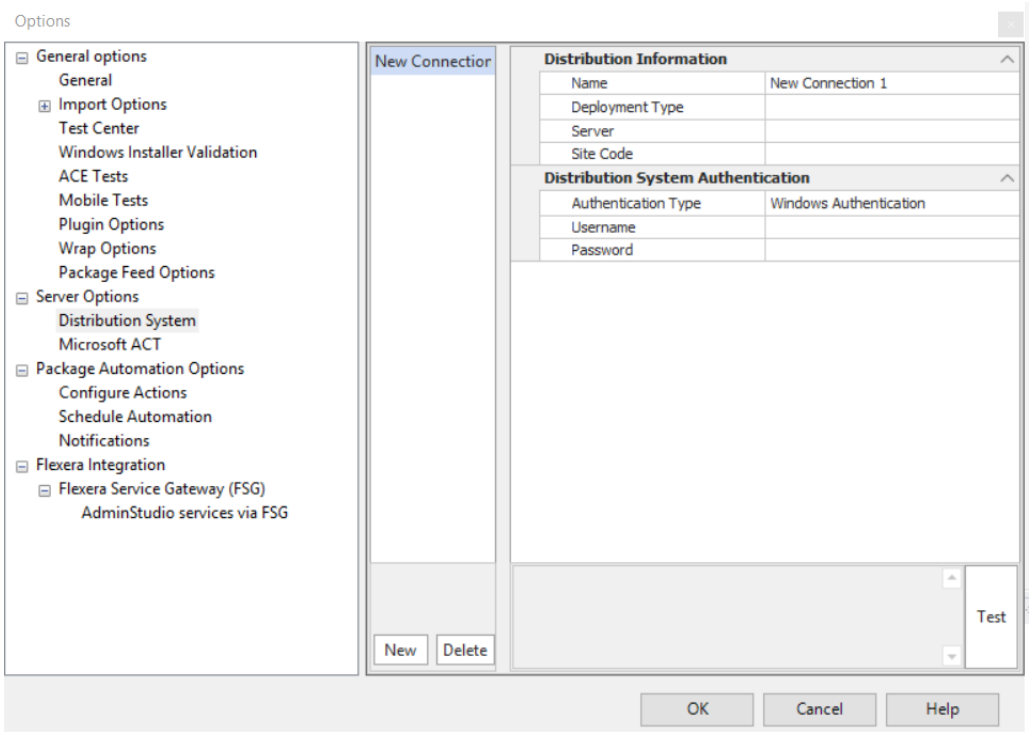
To set up Microsoft Intune connection in the Distribution Systems, perform the following steps:

Task ***To set up the Intune connection:***

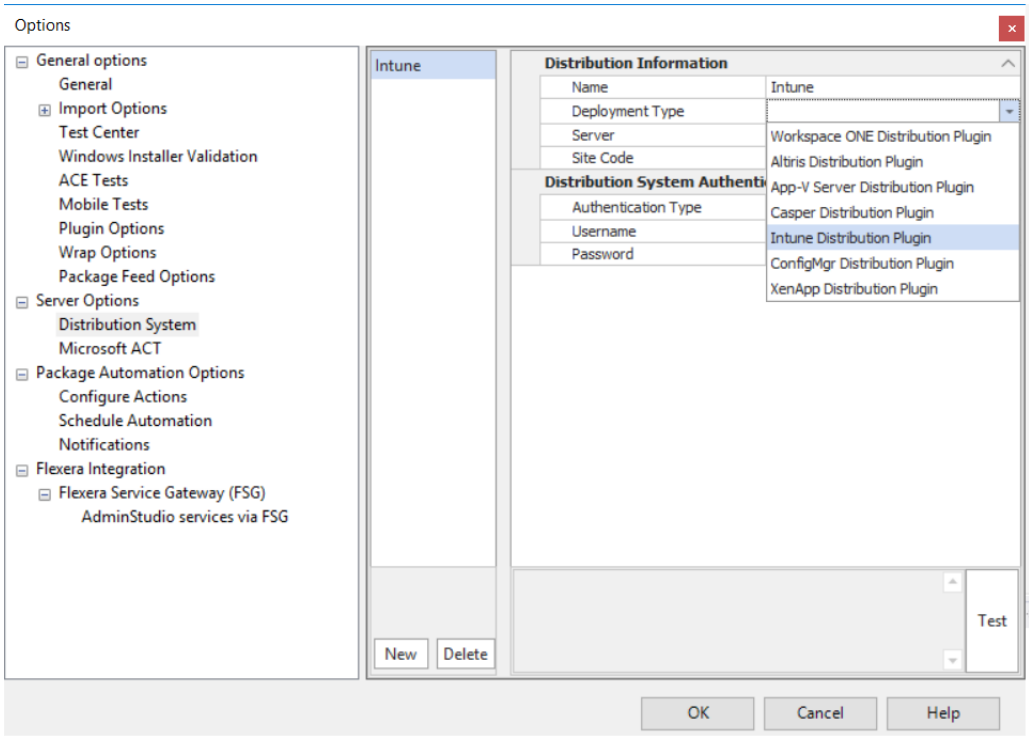
1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.



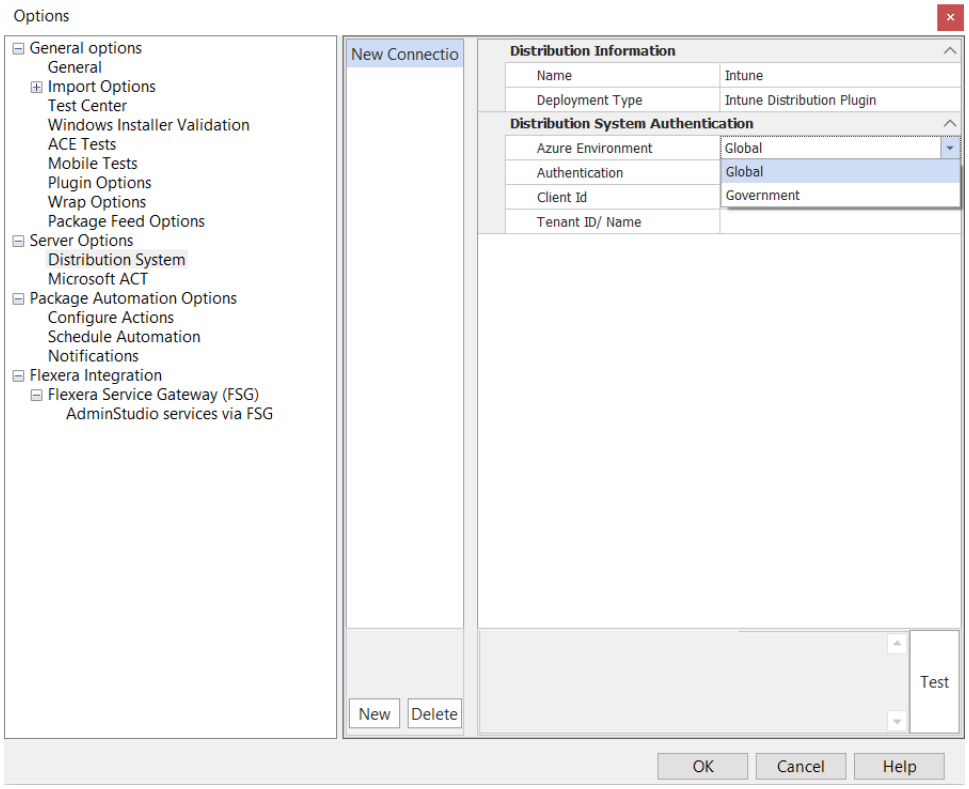
2. Under **Servers Options**, select **Distribution System**. The **Distribution System** tab opens and lists all defined connections. Click **New**, an empty connection setting field appears.



3. In the **Name** field, enter a name to identify this new named connection to a distribution system.
4. From the **Deployment Type** list, select **Intune Distribution Plugin**.

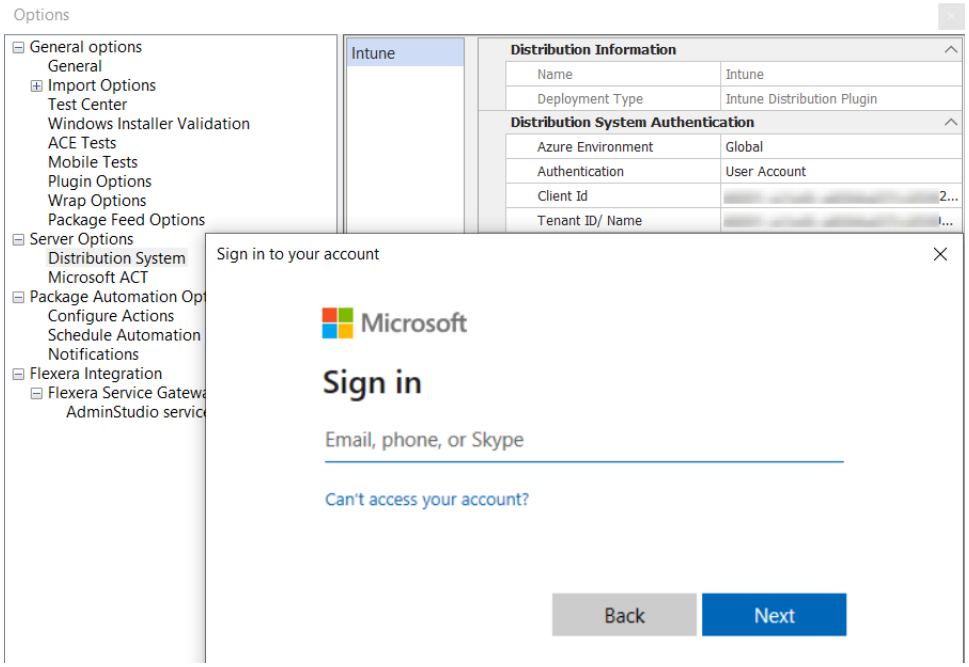


5. Under **Distribution System Authentication**, set the **Azure Environment** field to either **Global** or **Government**.



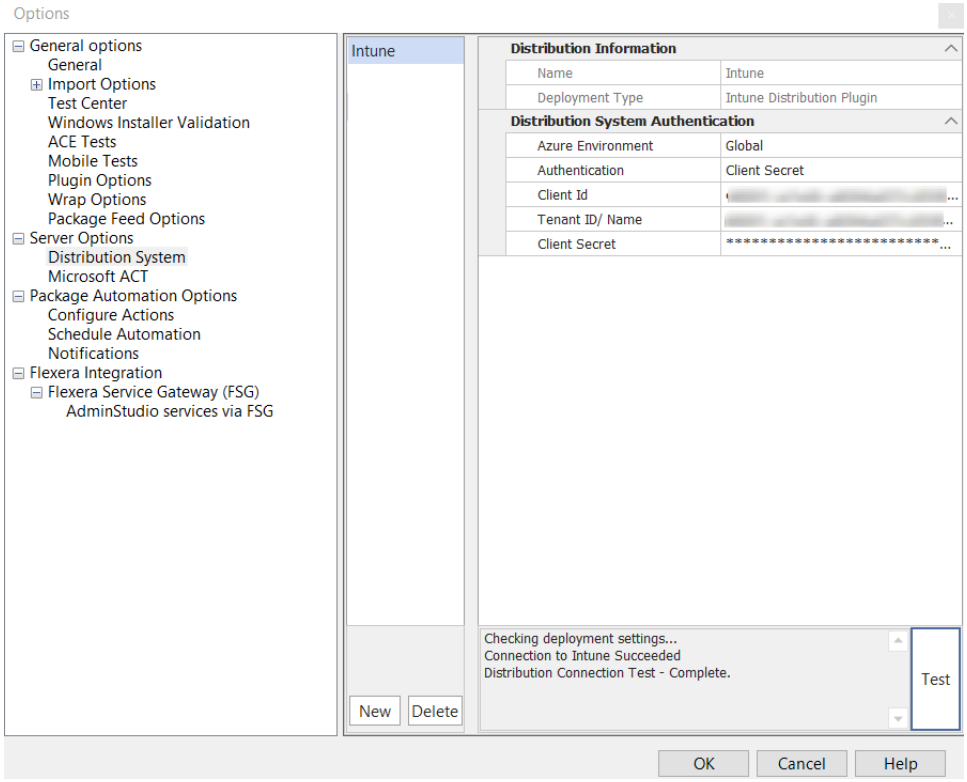
6. In the **Authentication Type** field set to either **Client Secret** or **User Account**.

7. If you select authentication as **User Account**, then enter **Client Id** and **Tenant ID/Name** in the respective text field. To know more details on configure Intune credentials, see Setting Up a Distribution Connection for Microsoft Intune. After entering the details, click **Test** to make sure that the authentications are correct. A Sign in window opens, enter the valid credentials and then click **Sign in**.



Note • *Sign in window popup every 24 hours.*

8. After successful testing of the distribution connection, click **OK** for the deployment of the connection.
9. If you select authentication as **Client Secret**, enter **Client Id**, **Tenant ID/Name**, and **Client Secret** in the respective text field. To know more details on configure Intune credentials, see Setting Up a Distribution Connection for Microsoft Intune. Click **Test** to make sure that the authentications are correct.



10. After successful testing of the distribution connection, click **Ok** for the deployment of the connection.

Setting Up the Distribution Properties

Once the application is imported, setup the application properties in the dynamic tab. These properties must be set up before publishing.

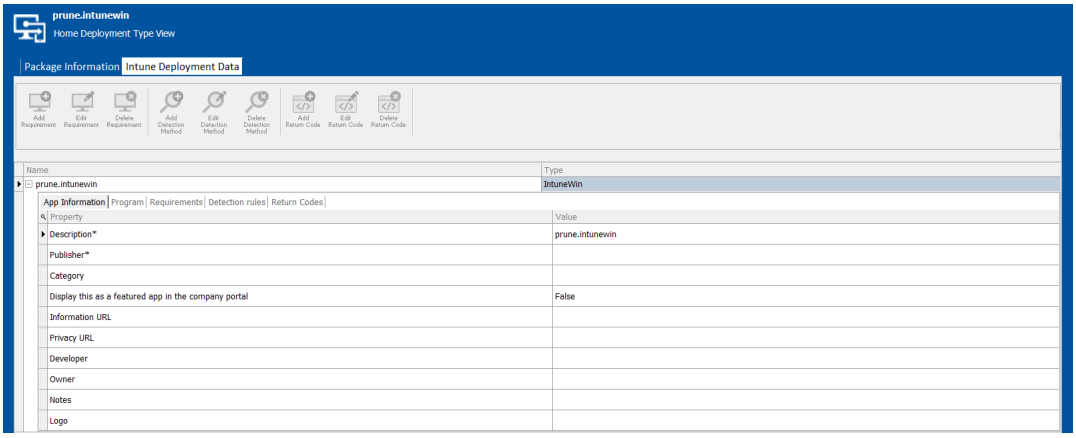


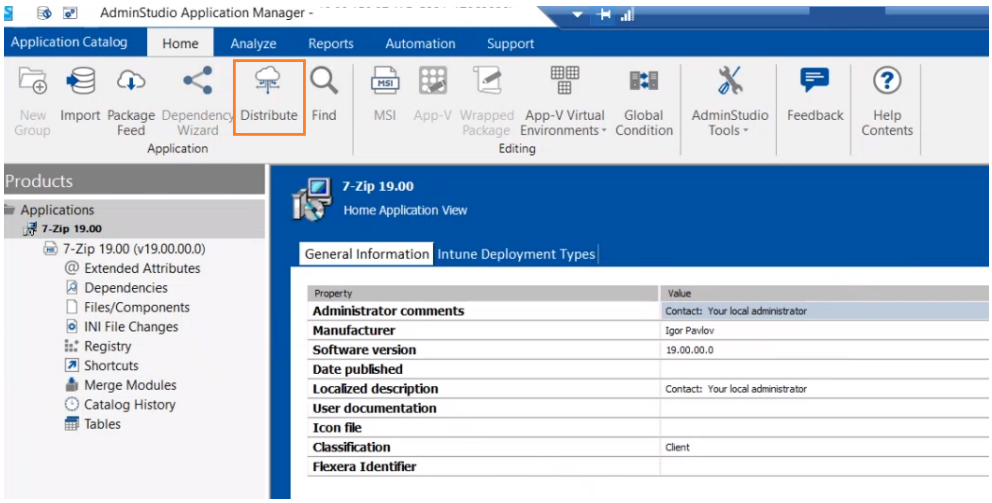
Figure 7-16: Set up the Distribution Properties

Publishing the Applications

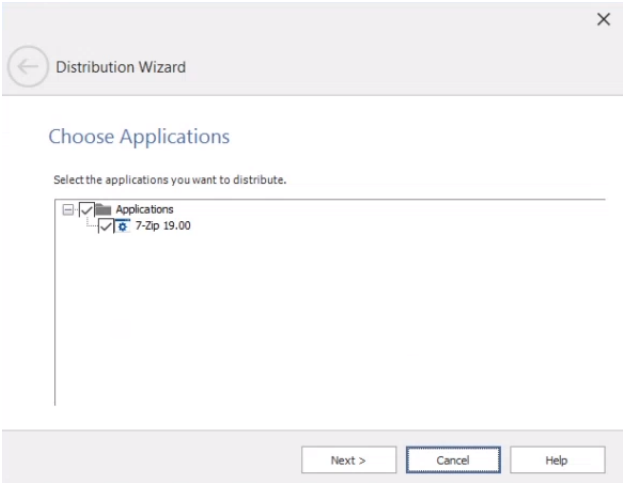
To publish or distribute applications, perform the following steps:

Task ***To publish applications:***

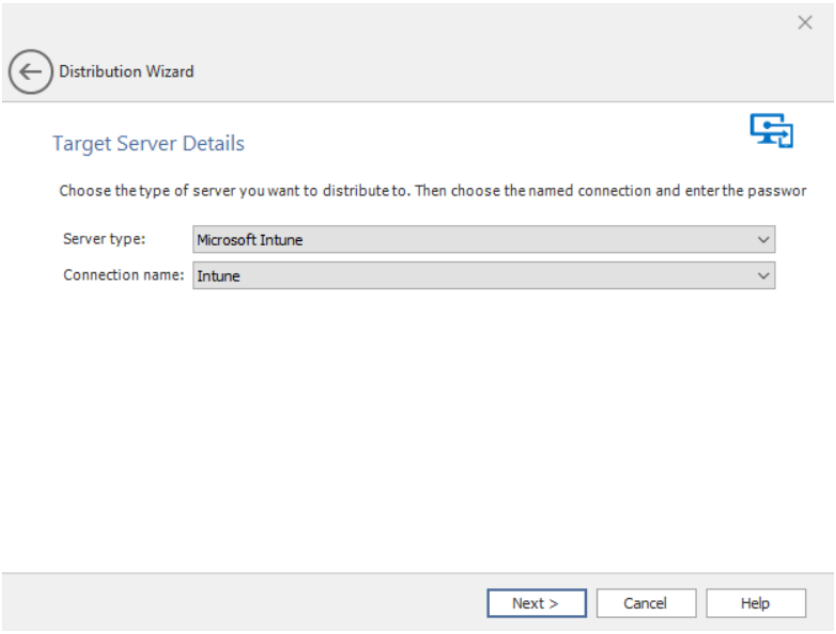
1. In the **Application Catalog** tree, select the application or group of applications, and click the **Distribute** button in the ribbon.



2. The **Choose Applications** panel under **Distribution Wizard** opens. Select the applications, and then click **Next**.



3. On the **Target Server Details** panel, select the required **Connection name** from the drop down, and then click **Next**.



4. The **Summary panel** appears with the details of the distribution. Click **Next**.
5. The **Distribution** panel opens with the distribution process. After completing the process, click **Finish**.

   Upon successful action, selected applications are published or distributed.

Note • *Converted Intunewin packages are published to the Intune server.*

Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard

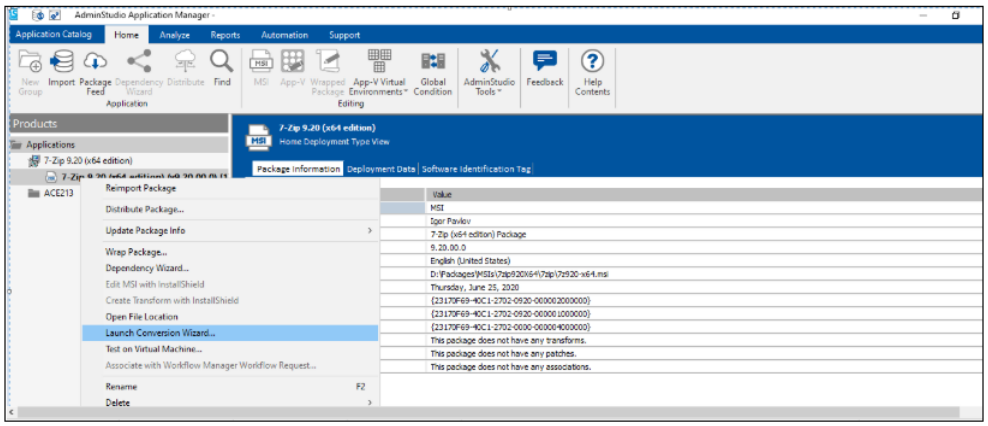
You can convert MSI/EXE packages to **Microsoft Intunewin** using the Conversion Wizard.

Important • *Conversion of MSI/EXE to Intune is supported only through the package nodes.*

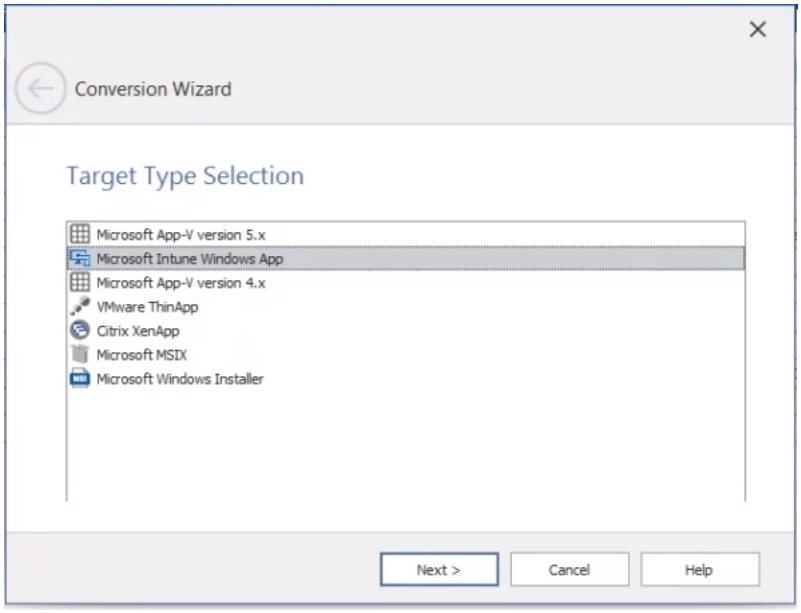
Perform the following steps to convert MSI/EXE packages to Intune using the Conversion Wizard.

Task ***To convert MSI/EXE packages to Intune:***

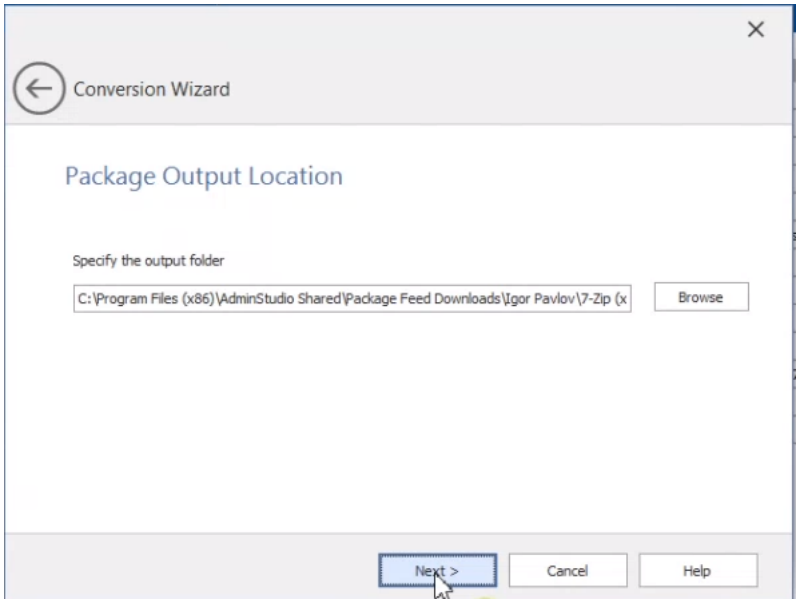
1. Perform the steps listed in Setting Conversion Wizard Options, including Setting Microsoft Intune Conversion Options.
2. On the **Applications** tab, right-click on the selected package(s), and then select **Launch Conversion Wizard** from the shortcut menu.



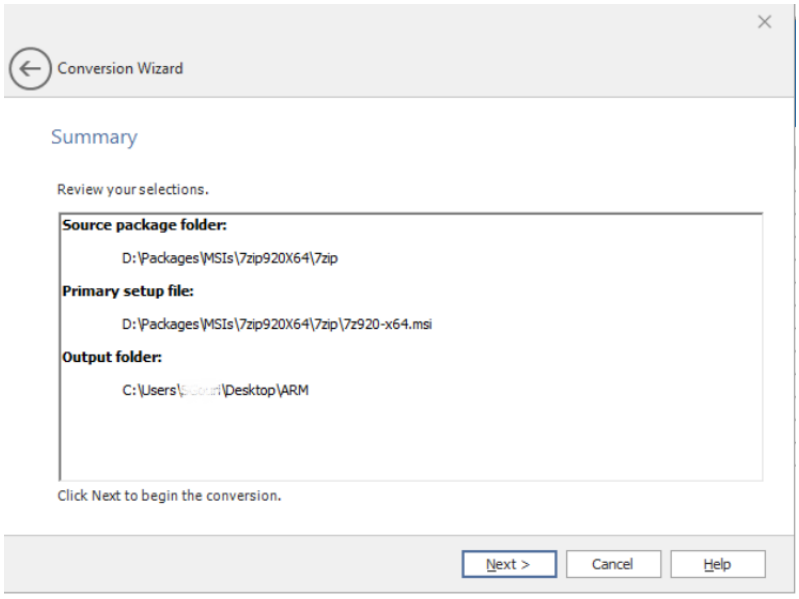
3. On the **Target Type Selection** panel of the **Conversion Wizard**, select **Microsoft Intune Windows App**, and then click **Next**.



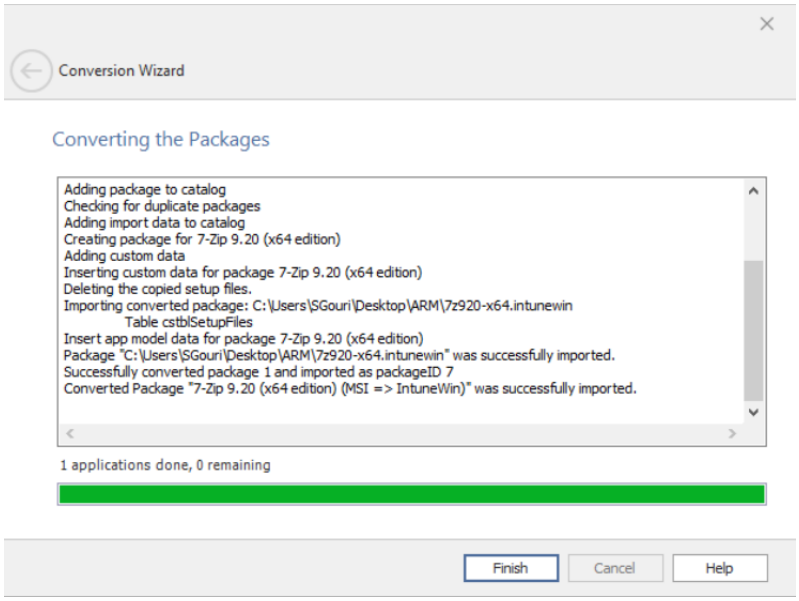
4. The **Package Output Location** panel opens. If you want to select different output location, click **Browse** and select preferred location (if required), and then click **Next**.



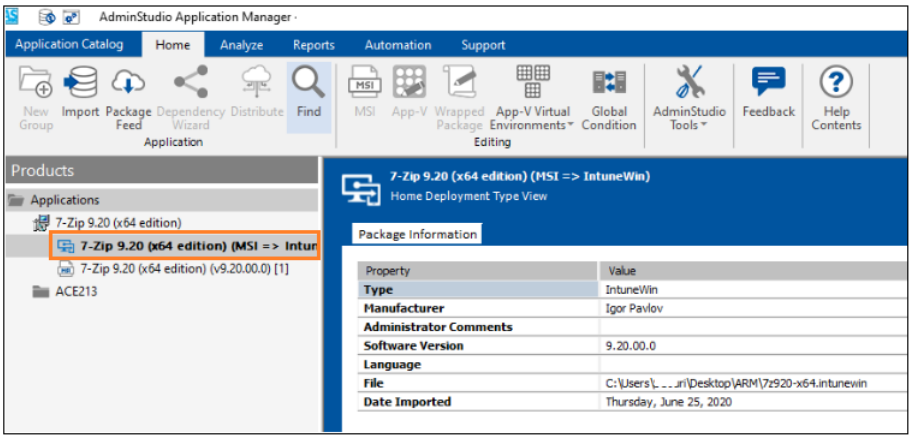
5. The **Summary panel** opens. Click **Next**.



6. On the **Converting the Packages** panel, click **Next** for the conversion process, and then click **Finish** once conversion process is complete.



Upon successful action, the converted package (converted to Intune) will import under that Application tree.



Note • *Converted package will be imported in .intunewin format.*

Monitored Directory for Package Automation

The Monitored Directory is a single directory were the packages will be executed on-demand and through scheduled automation. To configure the Monitored Directory, perform the following steps.

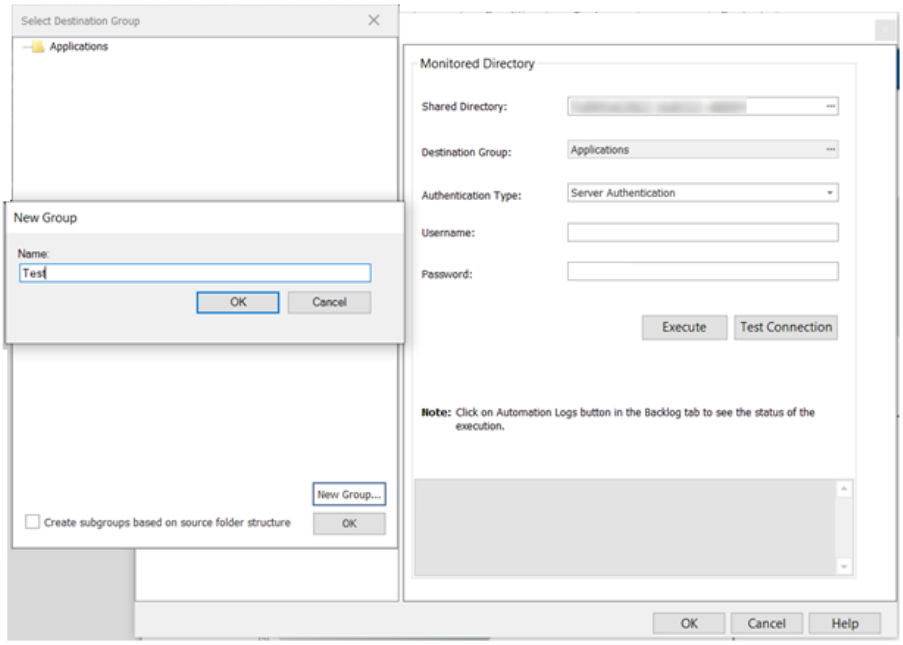
Task ***To configure the Monitored Directory:***

1. On the **Application Catalog** tab menu, select **Options**. The **Options** dialog box opens.
2. Under **Package Automation Options**, select **Monitored Directory**. The **Monitored Directory** tab opens.

3. In the **Shared Directory** field, enter the path or click the browse button (**...**) to browse to the path.
4. In the **Destination Group** field, click the browse button (**...**). The **Select Destination Group** dialog box opens.
5. Click the **New Group** button, enter name of the group, and then click **Ok**.

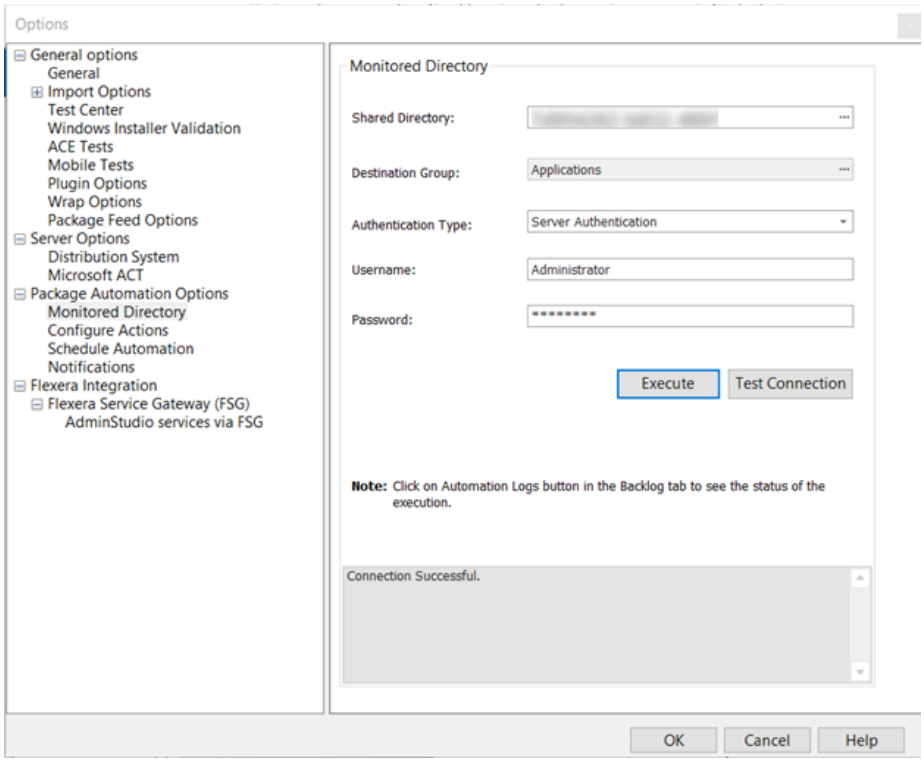
Note • *Multiple groups can be created.*

6. Click the **Create subgroup based on source folder structure** check box, if you want to create subgroups in the folder structure.



7. Click **Ok** button to exit the dialog box.
8. In the **Authentication Type** field, select one of the following type:
   - **Server Authentication**— Choose to use server login identification for authentication.
   - **Windows Authentication**— Choose to use Windows network authentication (your network login ID) to log into this Application Catalog.
   - **Username** and **Password**— If you chose **Server Authentication**, enter the appropriate Username and Password.
9. Click **Test Connection** to validate the Monitored Directory connection information. Upon successful validation, the **Connection Successful** message will appear at the bottom.

Note • *If validation is failed,* ***Connection Failed*** *message will be appeared.*



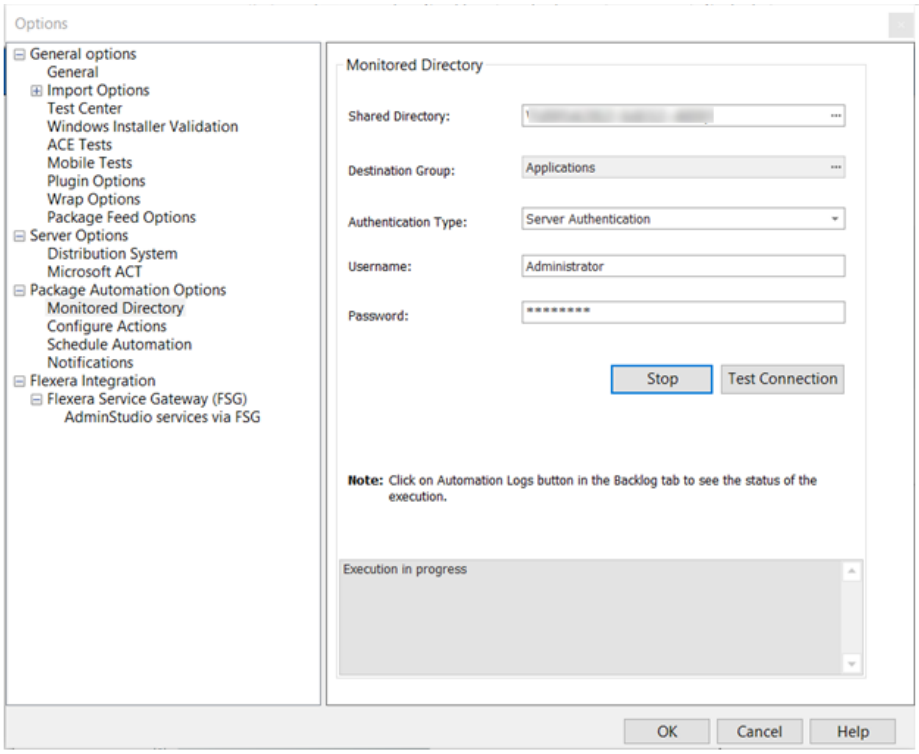
10. Click **Execute** button and confirm the popup for the execution. Upon successful execution, you can see the status of the execution under Automation Logs in the Backlog tab.

11. Packages in the Monitored Directory will be executed based on the package types, as shown in the below table.

Package Type	Import	Test	Wrap	Convert	Publish
ThinApp	Yes				
SFT	Yes	Yes		Yes (only App V)	Yes
PS1	Yes	Yes		Yes	Yes
PKG	Yes	Yes			
MSP	Yes				
MSIX	Yes				Yes
MSI	Yes	Yes	Yes	Yes	Yes
IPA	Yes	Yes			Yes
Intunewin	Yes				Yes (only Intune)
EXE	Yes	Yes	Yes	Yes	
DMG	Yes	Yes			
AppX	Yes	Yes			Yes
App-V	Yes	Yes			Yes

Package Type	Import	Test	Wrap	Convert	Publish
APP	Yes				
APK	Yes	Yes			Yes

12. During execution, (if required) you can stop by clicking **Stop** button.



13. After selecting the above fields, click OK.

Identifying Vendor Command Line Arguments

AdminStudio provides command line arguments for silent installation and uninstallation while importing any exe package into the Application Catalog. You can also modify the suggested command line arguments.

While importing, the default command line arguments are fetched for the supported exe packages. The Install Command Line and Uninstall Command Line switches are updated in the **Deployment Data** > **Programs** tab. When a package is updated from a previous version of AdminStudio, there is a menu option, “Command line update” to update the command line arguments.

You will be notified whether the default command line arguments are fetched or not through the import logs.

Some of the exe package types require the uninstaller for uninstallation. There are different types that require uninstallers. They are given below:

1. InstallAnywhere
2. WiseScript

3. NullSoft
4. InnoSetup

There are supported Exe Types and Default Arguments available that are listed below:

Installer Type	InstallCommandLine	UninstallCommandLine
InstallScript	installer.exe /s	installer.exe /x
InstallShield	installer.exe /s /v" /qb"	installer.exe /s /x /v" /qb"
SuiteInstallShield	installer.exe /silent	installer.exe /silent -remove
InstallAnywhere	installer.exe -i silent	-i silent
WiseScript	installer.exe /s	/s
AdvancedInstaller	installer.exe /i /quiet /norestart	installer.exe /x /quiet /norestart
Nullsoft	installer.exe /S	/S
InnoSetup	installer.exe /verysilent /norestart	/verysilent /norestart
WixBurn	installer.exe -s -norestart	installer.exe -uninstall -s -norestart

In AdminStudio, you can now wrap msi/exe packages to Wise script wrapped exes. You can wrap msi/exe packages using the **Wrap Package Wizard**. You can use the **Wrap Options** to select the wrap types.

For more information, see Wrapping MSI/EXE Packages to Wise Script Wrapped EXEs.

***Note** • If due to the nature of a software package, suggestions from AdminStudio are unavailable; it provides a customized **Search String** to ease the search of the correct command line arguments online.*

Viewing Application Testing and Analysis Reports on the Reports Tab

***Edition** • Application Manager's Reports tab is included with AdminStudio Enterprise Edition.*

On the Application Manager **Reports** tab, AdminStudio provides a wide array of reports containing Application Manager summary information on the applications in your Application Catalog, giving you insight into the readiness of those packages for distribution.

These reports include test results from operating system compatibility, MSIX Conversion Compatibility, virtualization compatibility, remote application publishing compatibility, best practices testing, and application conflict testing. For Mac OS, iOS and Android apps, reports on feature use, risk assessment, device compatibility, and policy compatibility are available. Reports are also included on App-V packages in your Application Catalog, as well as ConfigMgr (Formerly called as System Center Configuration Manager) deployment information.

Because these reports are created using Microsoft SQL Reporting Services, you can create your own custom reports by adding client report definition (`.rdlc`) files to the AdminStudio installation directory.

To learn more about the **Reports** tab, review the following topics:

- Available Reports
- Viewing a Report
- Exporting a Report in PDF, Excel, or Word Format
- Creating Custom Reports

Available Reports

Edition • *Application Manager's Reports tab is included with AdminStudio Enterprise Edition.*

The available reports on the **Reports** tab include test results from operating system compatibility, MSIX Conversion Compatibility, virtualization compatibility, remote application publishing compatibility, best practices testing, and application conflict testing. For macOS, iOS and Android apps, reports on feature use, risk assessment, device compatibility, and/or policy compatibility are available. Reports are also included on App-V packages in your Application Catalog, as well as ConfigMgr (Formerly called as System Center Configuration Manager) deployment information.

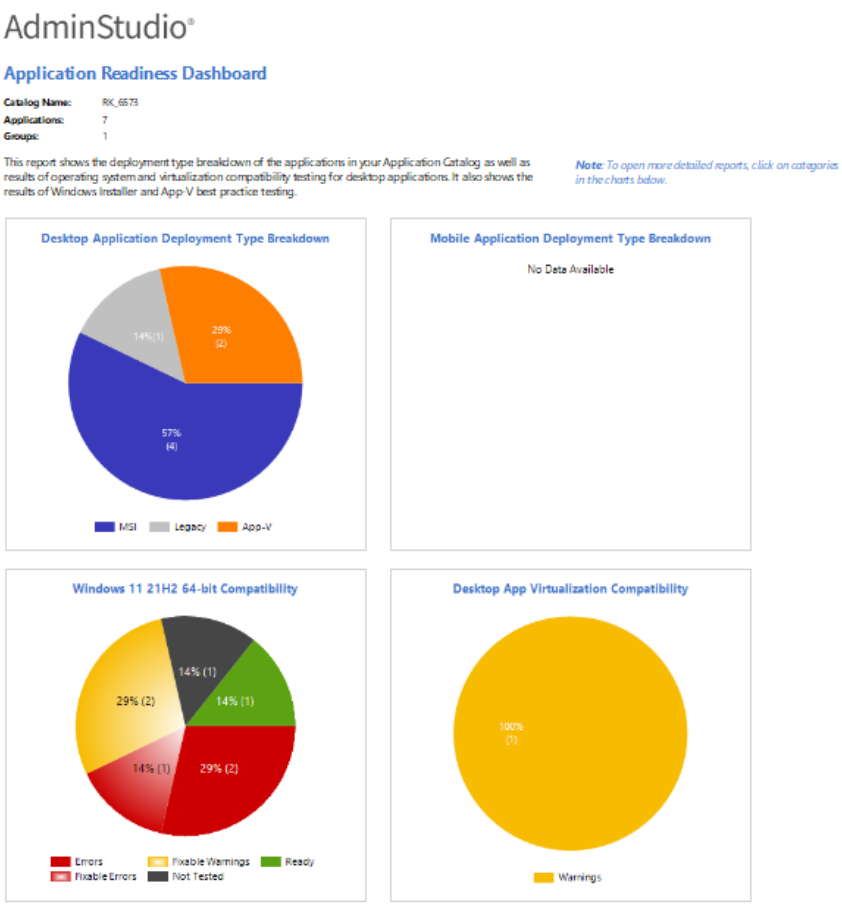


Figure 7-17: Application Readiness Dashboard Report

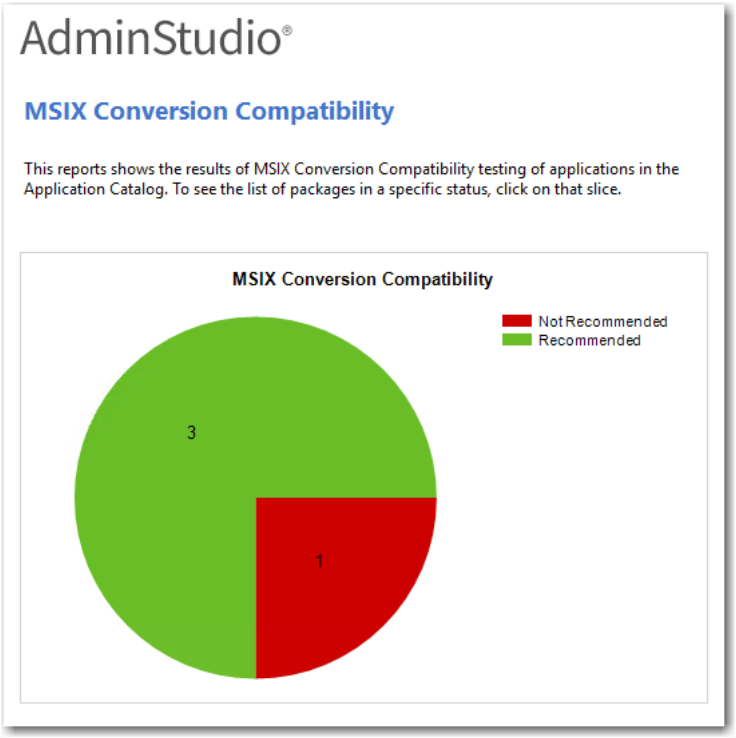


Figure 7-18: MSIX Conversion Compatibility Report

The **Reports** tab includes the following reports:

Table 7-13 • Reports Available on Reports Tab

Report Category	Reports
Application Catalog Dashboards	This category includes the following reports: • Application Readiness Dashboard • Operating System Compatibility Group Dashboard • Virtualization Compatibility Group Dashboard • Microsoft Application Compatibility Toolkit • Overall Package Testing Results
Desktop OS Compatibility	This category includes the following reports: • Windows Desktop OS Compatibility Dashboard • Mac Desktop OS Compatibility Dashboard • Test Impact Report • Test Suppression Report • Snapshot Analysis
MSIX	• MSIX Conversion Compatibility

Table 7-13 • Reports Available on Reports Tab

Report Category	Reports
Mobile	This category includes the following reports: • iOS/Android/Windows Phone Mobile Dashboard • iOS/Android/Windows Phone App Details • iOS/Android/Windows Phone App Feature Use • iOS/Android/Windows Phone App - Device Compatibility • iOS/Android/Windows Phone App - OS Compatibility • iOS App - Policy Compatibility • iOS Best Practices and Risk Assessment
Virtualization Compatibility	This category includes the following reports: • Application Virtualization Compatibility Dashboard

Installer Best Practices	This category includes the following reports:
	• Windows
	  ◦ Installer Best Practices
	  ◦ Windows Installer Conflicts
	  ◦ Shared Extensions
	  ◦ Error Category Breakdown
	• Mac
	  ◦ Mac Installer Best Practices and Risk Assessment

App-V Best Practices	This category includes the following reports:
	• App-V Best Practices
	• App-V Conflicts
	• App-V 4 Dynamic Suiting
	  ◦ Dependencies
	  ◦ Dependents
	  ◦ Recommended Dependencies
	• App-V 5 Virtual Environments
	  ◦ SCCM Server Environment
	  ◦ App-V Server Environment

Deployment Reports	This category includes the following reports: • Configuration Manager Deployments • Intune Deployments

Table 7-13 • Reports Available on Reports Tab

Report Category	Reports
Dependencies	This category includes the following reports: • Java Runtime Dependency

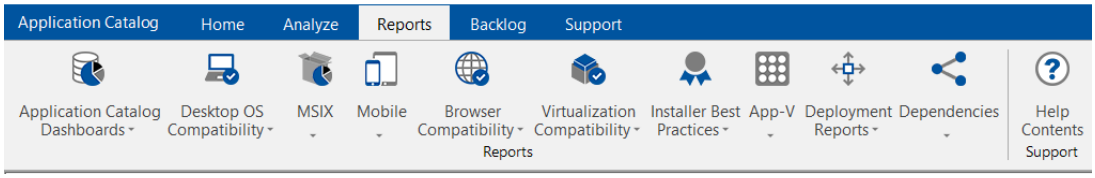
Viewing a Report

Edition • *Application Manager's Reports tab is included with AdminStudio Enterprise Edition.*

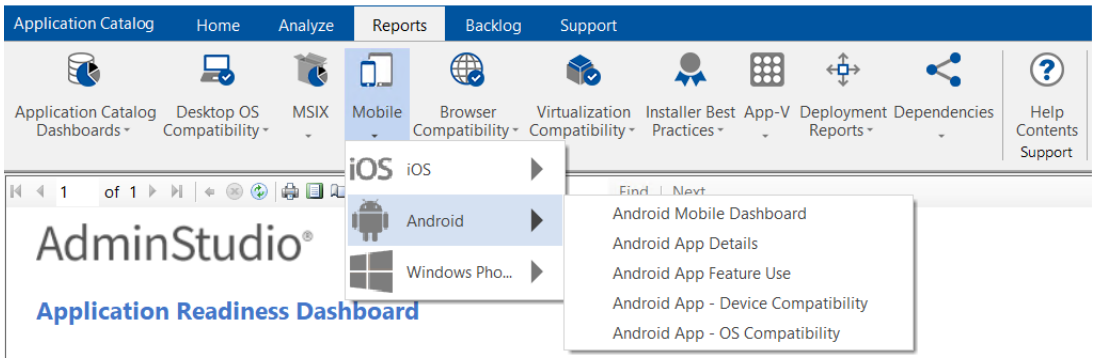
To view a report on the Reports tab, perform the following steps.

Task ***To view a report on the Reports tab:***

1. Open Application Manager and select the **Reports** tab. The Reports ribbon lists the available report groups:



2. Click on one of the report group icons and select a report from the drop-down list.



   The report opens.

3. For most reports, detailed sub-reports are available by clicking on one of the categories of the pie bar chart, on one of the numbers in an issue count column, one of the icons, or on a package name. Click on the available hyperlinks until you have explored all of the levels of the report.

Note • *For more information, see Viewing Mobile App Reports.*

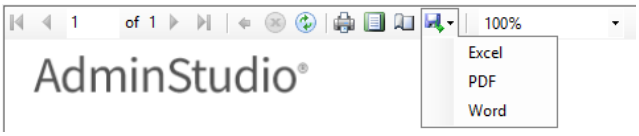
Exporting a Report in PDF, Excel, or Word Format

Edition • *Application Manager's Reports tab is included with AdminStudio Enterprise Edition.*

You can save any of the Application Catalog summary reports or any of their drill-through reports in PDF, Microsoft Excel, or Microsoft Word format.

Task ***Saving a report:***

1. Open the report that you want to save.
2. In the toolbar, click the **Save** icon.



3. From the menu, select either **Excel**, **PDF**, or **Word**. The report is exported and you are prompted for a location to store the report.
4. Specify a location and click **Save**.

Note • *You can also print the currently viewed report by clicking the* ***Print*** *icon in the toolbar.*

Creating Custom Reports

Edition • *Application Manager's Reports tab is included with AdminStudio Enterprise Edition.*

You can create your own custom reports that are generated using Microsoft SQL Reporting Services. Creating a custom report involves the following three steps:

- **Write a stored procedure** to obtain data from the AdminStudio Application Catalog database.
- **Create an .rdlc file** to format the display of the data in the desired manner. After you create an `.rdlc` file, using either Microsoft SQL Server Business Intelligence Development Studio or Visual Studio 2012, you need to copy that file to the AdminStudio installation directory.
- **Edit the AdminStudio.Reports.xml file** to add a reference to your custom report. The AdminStudio reporting framework uses the information in the `AdminStudio.Reports.xml` file to present the reporting view. This file includes the location, name, icons, and ribbon location of the report files. It also identifies the SQL queries or stored procedures that need to be run to populate the data for the report.

Reports Tab Report Groups

When you add a reference to your report to the AdminStudio Report Definition file, you need to specify where you want the report to be displayed on the **Reports** tab ribbon. By default, AdminStudio reports are grouped into eight groups, and each group has a drop-down list of reports:

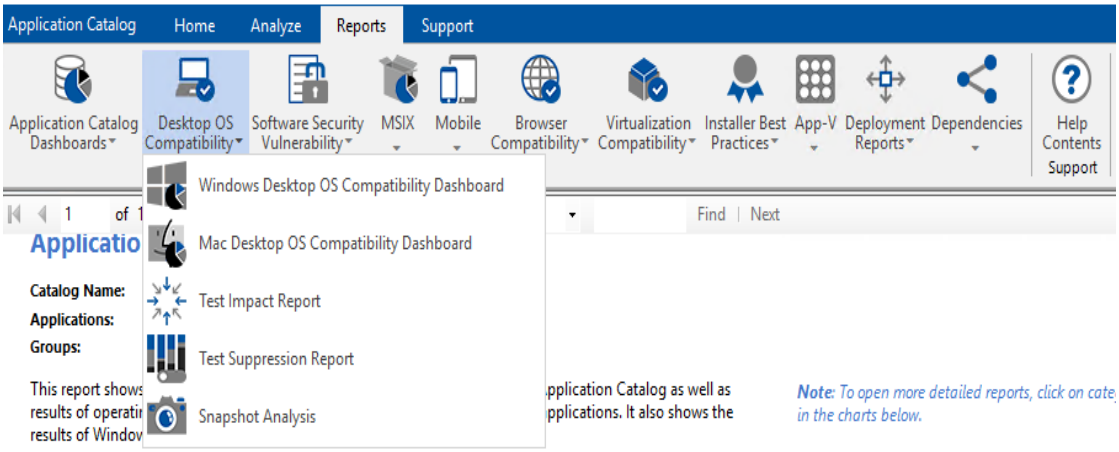


Figure 7-19: AdminStudio Report Groups

Before you begin to edit the AdminStudio Report Definition file, as described in Creating a Custom Report, you need to decide in which of these groups you want your new custom report to be listed. or whether you want your report to be listed in a new group. The parameters used to define a report's location in the ribbon are the `Group`, `GroupIndex`, and `OrderIndex` parameters of the `<Report>` element.

Creating a Custom Report

To create a custom AdminStudio report, perform the following steps:

Task ***To create a custom report and add it to the Reports tab ribbon:***

1. Write an SQL stored procedure to obtain the data for the report from the AdminStudio Application Catalog database.

   For reference on writing stored procedures, you can open the following file and look at the stored procedures for the existing AdminStudio reports:

   `[AdminStudio_Installation_Directory]\Support\SQL_Scripts\Reporting.StoredProcedures.sql`

   For example, if you first find the name of an existing report in the `AdminStudio.Reports.xml` file (such as `Test Impact Report`), and then identify the stored procedure used to generate that report (such as `sp_asrpt_GetOsCompatTestImpactTopLevel`), you can then open the `Reporting.StoredProcedures.sql` file and look at that stored procedure.

2. Design a custom report in an `.rdlc` file using either Microsoft SQL Server Business Intelligence Development Studio or Visual Studio 2012, along with `.rdlc` files for each drill-through sub-report that you want to include.

   Tip • *An easy way to make sure that your custom report matches the layout of existing reports is to copy one of the* `.rdlc` *files in the* `[AdminStudioInstallDirectory]\Common\ReportDefinition\RDLC` *directory and use it as your starting point.*

3. Copy your new `.rdlc` file(s) to the following directory on the machine where AdminStudio is installed:

   `[AdminStudioInstallDirectory]\Common\ReportDefinition\RDLC`

4. Open the `AdminStudio.Reports.xml` file, found in the following location, in a text editor:

   `[AdminStudioInstallDirectory]\Common\ReportDefinition`

5. In the `AdminStudio.Reports.xml` file, locate and copy the code for an existing report that is listed in the group on the **Reports** tab ribbon that you want your report to be listed in. You need to copy all of the code between the `<Report>` and `</Report>` elements, including the code for drill-through sub-reports.

   For example, if you want your report to appear in the **Windows Installer Best Practices** group, you could copy the code for the **Shared Extensions** report (and its drill-through report):

```
01  <Report Name="Shared Extensions" RdlcPathType="Relative"
            RdlcPath="RDLC\SharedExtensions.rdlc" Group="Windows Installer Best Practices"
            GroupIndex="5" OrderIndex="3" Icon="33120" GroupIcon="33138" >
02      <DataSources>
03          <DataSource DataSourceName="ds_asrpt_SharedExtensions"
                    SqlCommandType="StoredProcedure"
                    SqlCommandString="sp_asrpt_SharedExtensions">
04              <SqlParameters>
05                  <SqlParameter/>
06              </SqlParameters>
07          </DataSource>
08      </DataSources>
09      <Report Name="Shared Extensions By Product" RdlcPathType="Relative"
                RdlcPath="RDLC\SharedExtensionsByProduct.rdlc">
10          <DataSources>
11              <DataSource DataSourceName="ds_asrpt_SharedExtensionsByProduct"
                        SqlCommandType="StoredProcedure"
                        SqlCommandString="sp_asrpt_SharedExtensionsByProduct">
12                  <SqlParameters>
13                      <SqlParameter DrillThroughParameterName="Extension"
                            SqlParameterName="@extension" SqlParameterValue=""/>
14                  </SqlParameters>
15              </DataSource>
16          </DataSources>
17      </Report>
18  </Report>
```

Important • *Line numbers have been added to this example.*

6. Use the information on the following table to replace the highlighted text above with the correct information for your new custom report, including information for all of your custom report's drill through reports:

Line	Element/Parameter	Values
01	`<Report`	Opens the definition of the main report.
01	`Name`	Enter the name of the custom report. This name will be listed in the drop-down menu that opens when you click on the report group icon in the ribbon.
01	`RdlcPathType`	Valid values are `Relative` or `Absolute`.

Line	Element/Parameter	Values
01	RdlcPath	Enter the name of the custom report `.rdlc` file.
01	Group	Enter the name of the report group that this report will be listed in. In this example, the Group parameter is set to `Windows Installer Best Practices`. ***Note*** • *All of the reports in this group will have the same value for this parameter.* ***Note*** • *This parameter is not necessary for drill-through reports.*
01	GroupIndex	Set this parameter to a number to specify the order that this group will appear in the **Reports** ribbon. In this example, the GroupIndex parameter is set to 5, meaning that it will be the fifth group listed in the Reports ribbon, counting from left to right. ***Note*** • *All of the reports in this group will have the same value for this parameter.* ***Note*** • *Not necessary for drill-through reports.*
01	OrderIndex	Set this parameter to a number to specify the order that this report will appear in drop-down list that opens when you click on this report's group icon in the **Reports** ribbon. In this example, the OrderIndex parameter is set to **3**, meaning that it will be the third report in the list. ***Note*** • *Not necessary for drill-through reports.*
01	Icon	This icon specifies the icon that is displayed to the left of the report name in the drop-down list. For the reports shipped with AdminStudio, this parameter is set to a five-digit code to specify an image resource in the AdminStudio binary file. For your custom report, set this parameter to an absolute path to the icon that you want to use for this report. ***Note*** • *All of the reports in this group will have the same value for this parameter.* ***Note*** • *Not necessary for drill-through reports.*

Line	Element/Parameter	Values
01	GroupIcon	This icon specifies the icon that is displayed to the Reports ribbon for this group. For the groups defined in the XML file shipped with AdminStudio, this parameter is set to a five-digit code to specify an image resource in the AdminStudio binary file. If you are creating a custom group, set this parameter to an absolute path to the icon that you want to use for this group.
02	<DataSources>	Element that opens the list of data sources for this report.
03	<DataSource	Element that opens a data source for this report.
03	DataSourceName	This name should match the data source specified in the RDLC file.
03	SqlCommandType	Valid values are StoredProcedure or SQL.
03	SqlCommandString	Enter the stored procedure name or an SQL string.
04	<SqlParameters>	Element that opens the list of SQL parameters.
05	<SqlParameter>	A parameter to pass to the stored procedure. This is usually used for drill-through reports to convey the context of what was clicked.
09	<Report	Because this element is nested within the first <Report> element, it defines a drill-through report for the first report. It will not be listed in the drop-down list; it is opened by clicking on a link in the original report. Define this report using the same parameters you used to define the parent report. However, it is not necessary to define the following parameters: Group, GroupIndex, GroupIcon, Icon, OrderIndex.

7. After you save this file, open the **Reports** tab and this locate this report, which should be listed in the location you specified.

Managing ConfigMgr (Formerly called as System Center Configuration Manager) Application Model Data

When the **Home** tab is selected in the Application Manager ribbon and an application is selected in the tree, the **Application View** opens, which provides summary information about the application, deployment data for each of its deployment types, dependencies/supersedences information, and ConfigMgr (Formerly called as System Center Configuration Manager) deployment status. Much of this information is used during deployment to ConfigMgr (Formerly called as System Center Configuration Manager).

- Specifying General Application Information
- Specifying Deployment Data for an Application's Packages
- Specifying Intune Deployment Data for an Application's Packages
- Viewing Reference Data: Dependencies and Supersedences

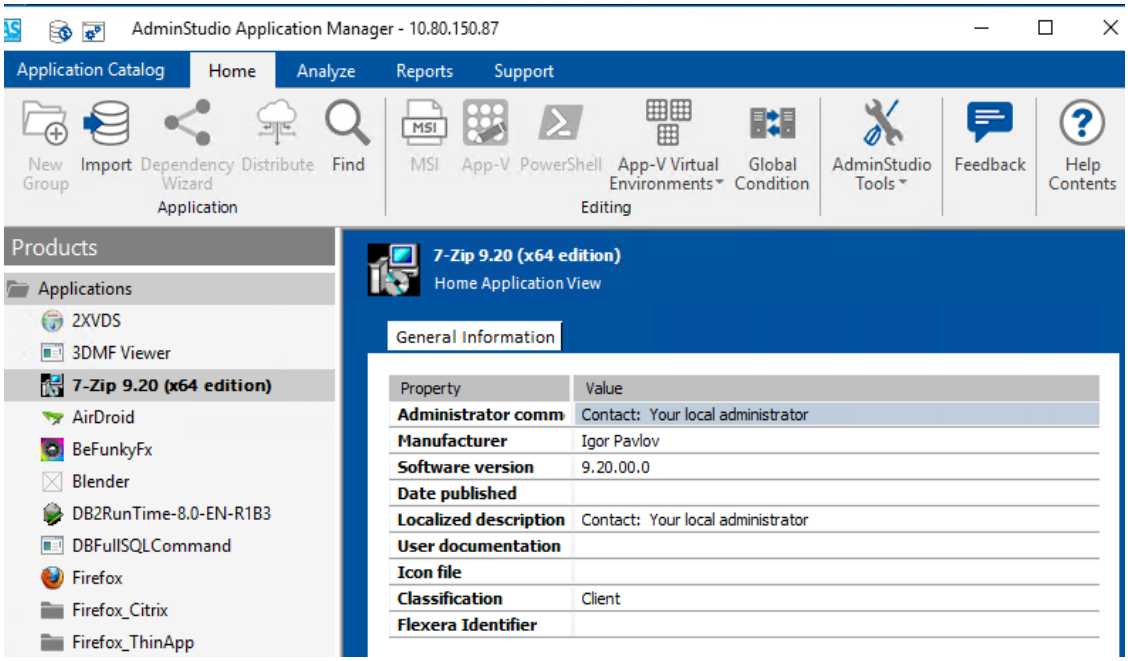
- Viewing Microsoft Configuration Manager Deployment Information
- Viewing Microsoft Intune Deployments Report
- Enabling Application Extended Attributes
- Retiring or Reinstating an Application in System Center 2012 Configuration Manager

Specifying General Application Information

You can view summary information about the application that AdminStudio gathered during package import on the **General Information** tab of the **Application View**.

Task ***To view general application information:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an application in the tree. The **Home Application View** opens.
3. Click the **General Information** tab. The **General Information** tab opens.



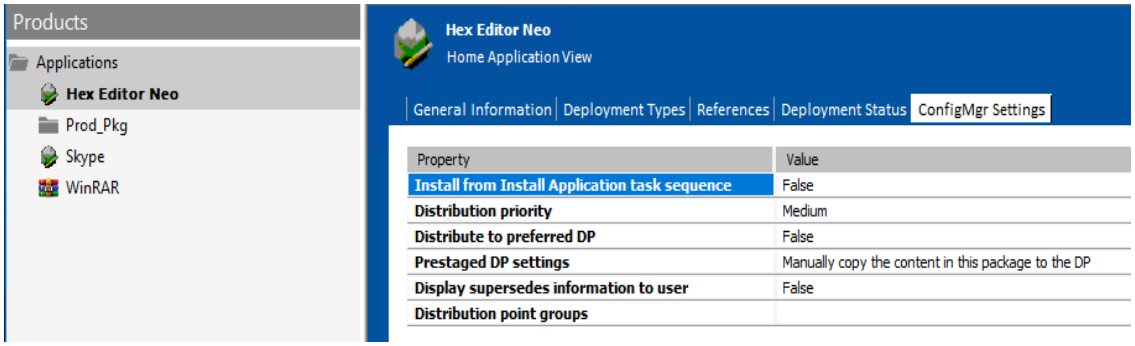
4. Review and edit the listed data, as described in General Information Tab.

Specifying ConfigMgr (Formerly called as System Center Configuration Manager) Information

You can view summary information about the application that AdminStudio gathered during package import on the **SCCM Settings** tab of the **Application View**.

Task ***To view general application information:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an application in the tree. The **Home Application View** opens.
3. Click the **SCCM Settings** tab. The **SCCM Settings** tab opens.



4. Review and edit the listed data, as described in ConfigMgr Settings Tab.

Specifying Deployment Data for an Application's Packages

The **Deployment Types** tab of the **Application View** lists data for all of the application's deployment types. It contains the same information that is displayed on the Deployment Data Tab for each of its associated deployment types (packages).

When you click the plus sign next to a package name on the **Deployment Types** tab, it expands to list the same deployment information that is displayed on the **Deployment Data** of the **Home Deployment Type View** for the selected package. Much of this information is used during deployment to ConfigMgr (Formerly called as System Center Configuration Manager).

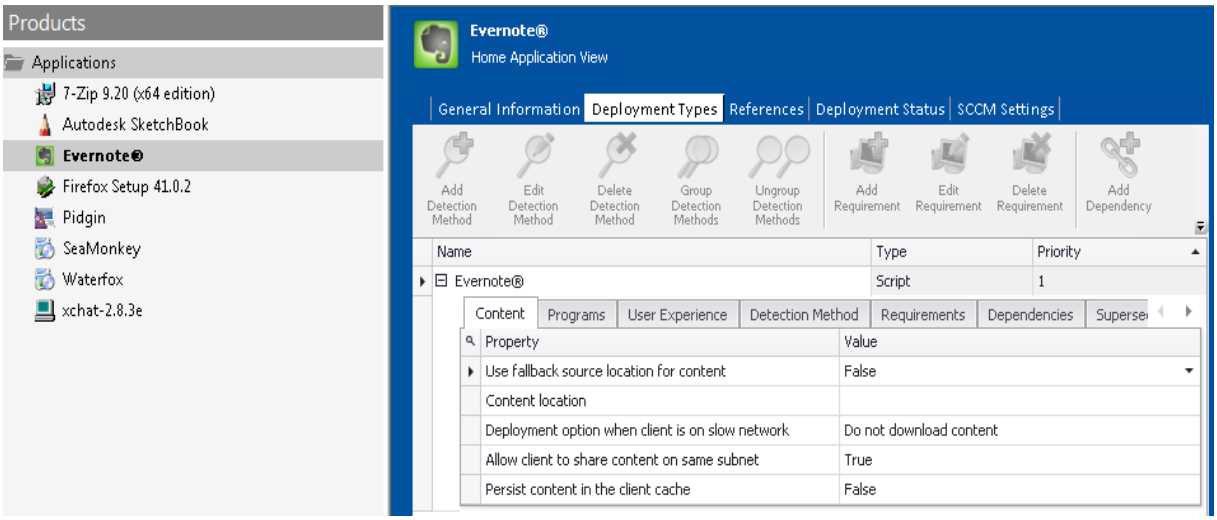


Figure 7-20: Application View / Deployment Types Tab (Expanded)

For information on the metadata displayed and defined on the subtabs of the **Deployment Types** tab, see the following sections:

- Specifying Package Content Deployment Data
- Specifying Package Programs Deployment Data
- Specifying Package User Experience Deployment Data
- Specifying Package Detection Methods Deployment Data
- Specifying Package Requirements Deployment Data
- Specifying Package Dependencies Deployment Data
- Specifying Package Supersedences Deployment Data
- Viewing and Editing Return Codes

Specifying Intune Deployment Data for an Application's Packages

The **Intune Deployment Types** tab of the **Home Application View** lists data for all of the application's deployment types. It contains the same information that is displayed on the Microsoft Intune Deployment Data Tab for each of its associated deployment types (packages).

When you click the plus sign next to a package name on the **Intune Deployment Types** tab, it expands to list the same deployment information that is displayed on the **Intune Deployment Data** of the **Home Deployment Type View** for the selected package.

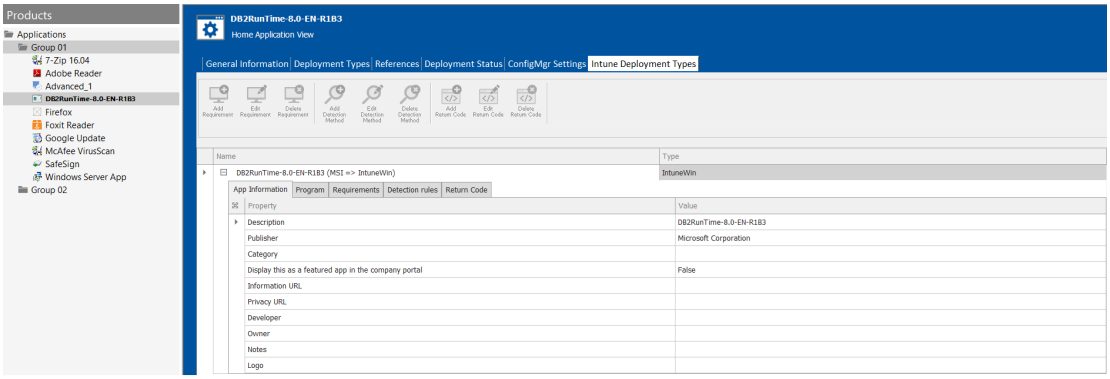


Figure 7-21: Application View / Intune Deployment Types Tab

For information on the subtabs of the Intune Deployment Types tab, see the following sections:

- Specifying App Information
- Specifying Package Programs
- Specifying Package Requirements
- Specifying Package Detection Rules
- Specifying Return Codes

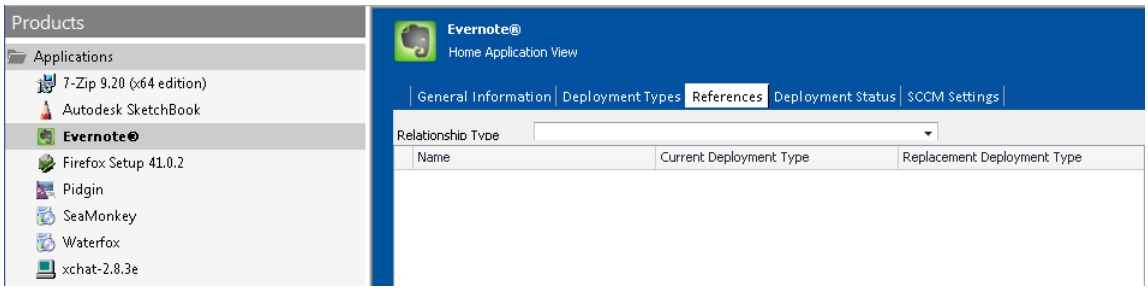
Viewing Reference Data: Dependencies and Supersedences

On the **References** tab of the **Home Application View**, you can view a list of packages that are dependent upon this application or that supersede this application. These dependencies are defined on the **Dependencies** and **Supersedence** subtabs of the **Deployment Data** of the **Home Deployment Type View** for a selected package. If another package has specified that it is dependent upon or supersedent to this package, that package will be listed here.

For more information, see Specifying Package Dependencies Deployment Data and Specifying Package Supersedences Deployment Data.

Task ***To view reference data:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an application in the tree. The **Home Application View** opens.
3. Click the **Reference** tab. The **References** tab opens.



4. Review and edit the listed data, as described in References Tab.

Viewing Microsoft Configuration Manager Deployment Information

If you have specified your ConfigMgr (Formerly called as System Center Configuration Manager) server connection information on the **Distribution System** tab of the Application Manager **Options** dialog box, Application Manager will display an application's Configuration Manager deployment status both on the **Deployment Status** tab of the **Home Application View** and on the **Microsoft Configuration Manager Deployments Report** on the **Reports** tab.

- Viewing an Application's Configuration Manager Deployment Status
- Viewing the Microsoft Configuration Manager Deployments Report

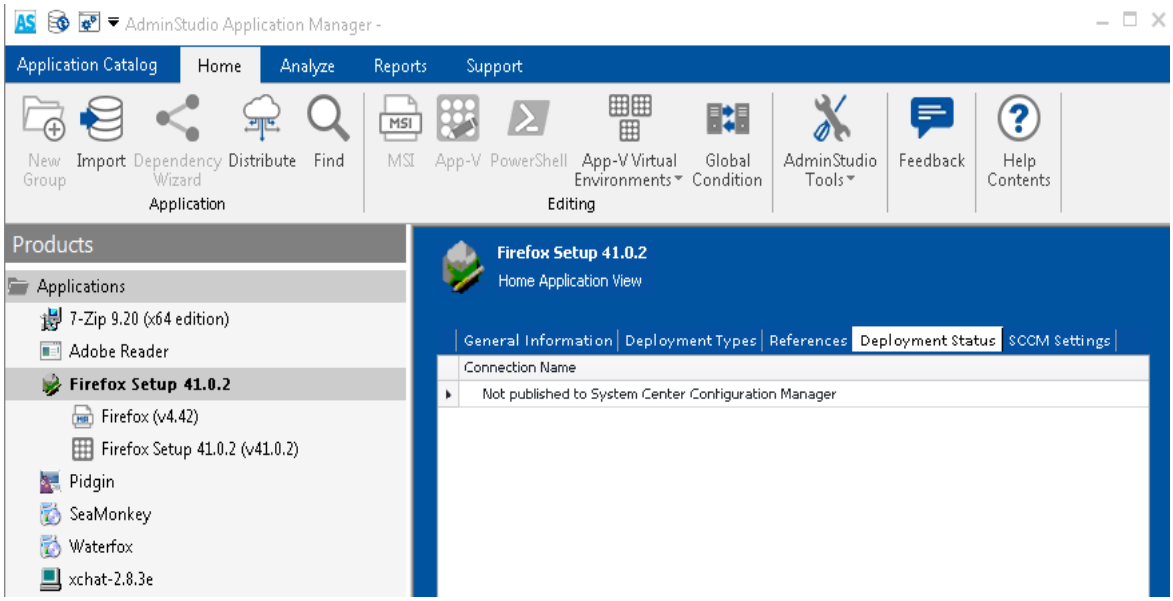
Viewing an Application's Configuration Manager Deployment Status

The **Deployment Status** tab of the **Home Application View** lists data from ConfigMgr (Formerly called as System Center Configuration Manager) that is specific to this application, not to its deployment types. The data is read from the active ConfigMgr (Formerly called as System Center Configuration Manager) server that has been specified on the **Server Options > Distribution System** tab of the Application Manager **Options** dialog box.

If Application Manager is unable to establish an active link to the ConfigMgr (Formerly called as System Center Configuration Manager) server, then a message indicating that there is no active connection will be displayed.

Task ***To view reference data:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an application in the tree. The **Home Application View** opens.
3. Click the **Deployment Status** tab. The **Deployment Status** tab opens.



4. Review the listed data, as described in Deployment Status Tab.

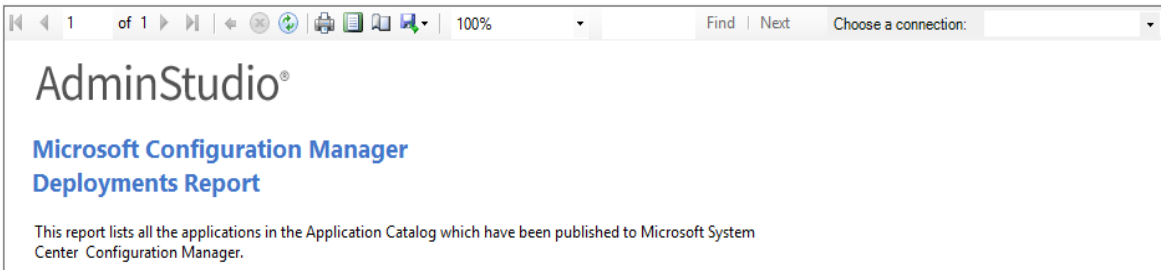
Viewing the Microsoft Configuration Manager Deployments Report

The **Microsoft Configuration Manager Deployments Report** on the **Reports** tab lists the applications in the Application Catalog which have been published to ConfigMgr (Formerly called as System Center Configuration Manager).

Note • *In order for this report to contain information, you need to have entered your ConfigMgr (Formerly called as System Center Configuration Manager) server information on the* ***Distribution System*** *tab of the Application Manager* ***Options*** *dialog box, as described in Creating Multiple Named Connections to Distribution Systems.*

Task ***To view the Microsoft Configuration Manager Deployments Report:***

1. Open Application Manager and select the **Reports** tab in the ribbon.
2. In the ribbon, click **Deployment Reports** and select **Configuration Manager Deployments**.
3. If you have multiple connections to ConfigMgr (Formerly called as System Center Configuration Manager) servers, select the connection that you want to view a report on from the **Choose a connection** list.



The **Microsoft Configuration Manager Deployments Report** includes the following information:

- **Name**—Name of deployed application.
- **# of Deployments**—Number of machines this application has been deployed to by the connected ConfigMgr (Formerly called as System Center Configuration Manager) server.
- **Status**—Whether this application's status is **Active** (ready for deployment) or **Inactive** (not ready for deployment).
- **Is Deployed?**—Value is **True** if application has been deployed; set to **False** if the application has not been deployed.

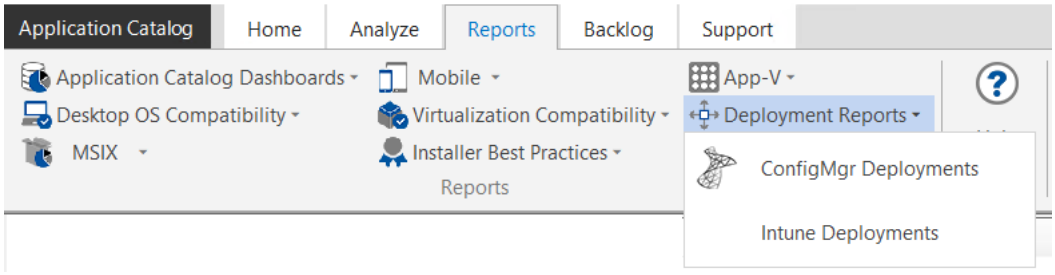
Viewing Microsoft Intune Deployments Report

The **Intune Deployments Report** on the **Reports** tab lists the packages in the Application Catalog which have been published to Intune.

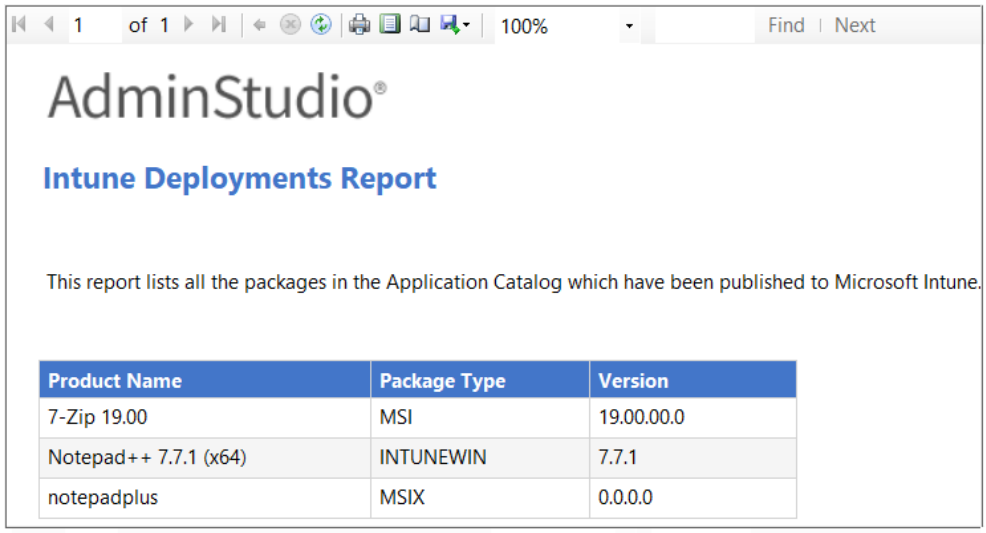
Note • *In order for this report to contain information, you need to have entered your Intune server information on the* ***Distribution System*** *tab of the Application Manager* ***Options*** *dialog box, as described in Creating Multiple Named Connections to Distribution Systems.*

Task ***To view the Microsoft Intune Deployments Report:***

1. Open Application Manager and select the **Reports** tab in the ribbon.
2. In the ribbon, click **Deployment Reports** and then select **Intune Deployments** from the list.



3. If you have multiple connections to Intune servers, select the connection that you want to view a report on from the **Choose a connection** list.



The **Intune Deployments Report** includes the following information:

- **Product Name**—Name of deployed package.
- **Package Type**—Specifies the type of package.
- **Version**—Specifies the version of package.

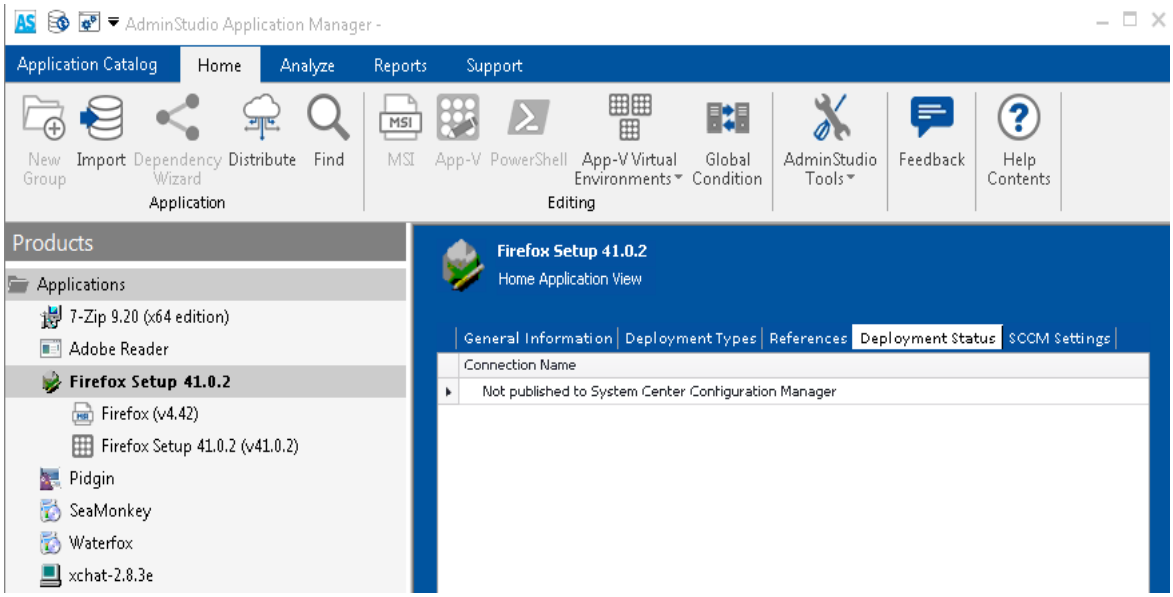
Retiring or Reinstating an Application in System Center 2012 Configuration Manager

You can retire or reinstate an application in System Center 2012 Configuration Manager by changing its **Status** property on the **Deployment Status** tab of the **Application View**, without even being required to republish the application.

When you *retire* an application, it is no longer available for deployment but the application and any deployments of the application are not deleted. Existing copies of this application that have been installed on client computers will not be removed. If an application that has no deployments is retired, it will be deleted from the Configuration Manager console after 60 days. However, any installed copies of the application are not removed.

Task ***To retire or reinstate an application in System Center 2012 Configuration Manager:***

1. Open the **Home** tab of Application Manager.
2. Select application node of an application that has been published to System Center 2012 Configuration Manager. The **General Information** tab of the **Application View** opens.
3. Open the **Deployment Status** tab.



4. Set the **Status** field to one of the following options:

   - **Retire**—Select this option to make this application unavailable for distribution by System Center 2012 Configuration Manager.
   - **Active**—Select this option to reinstate this application, making a formerly retired application once again available for distribution by System Center 2012 Configuration Manager.

Managing macOS Desktop Application Metadata

To enable you to successfully manage macOS desktop applications in your enterprise—both local package files and those from a public store—AdminStudio can extract and analyze metadata from those applications.

All of this metadata is needed for operating system compatibility and best practices testing.

Information about macOS desktop application metadata is presented in the following sections:

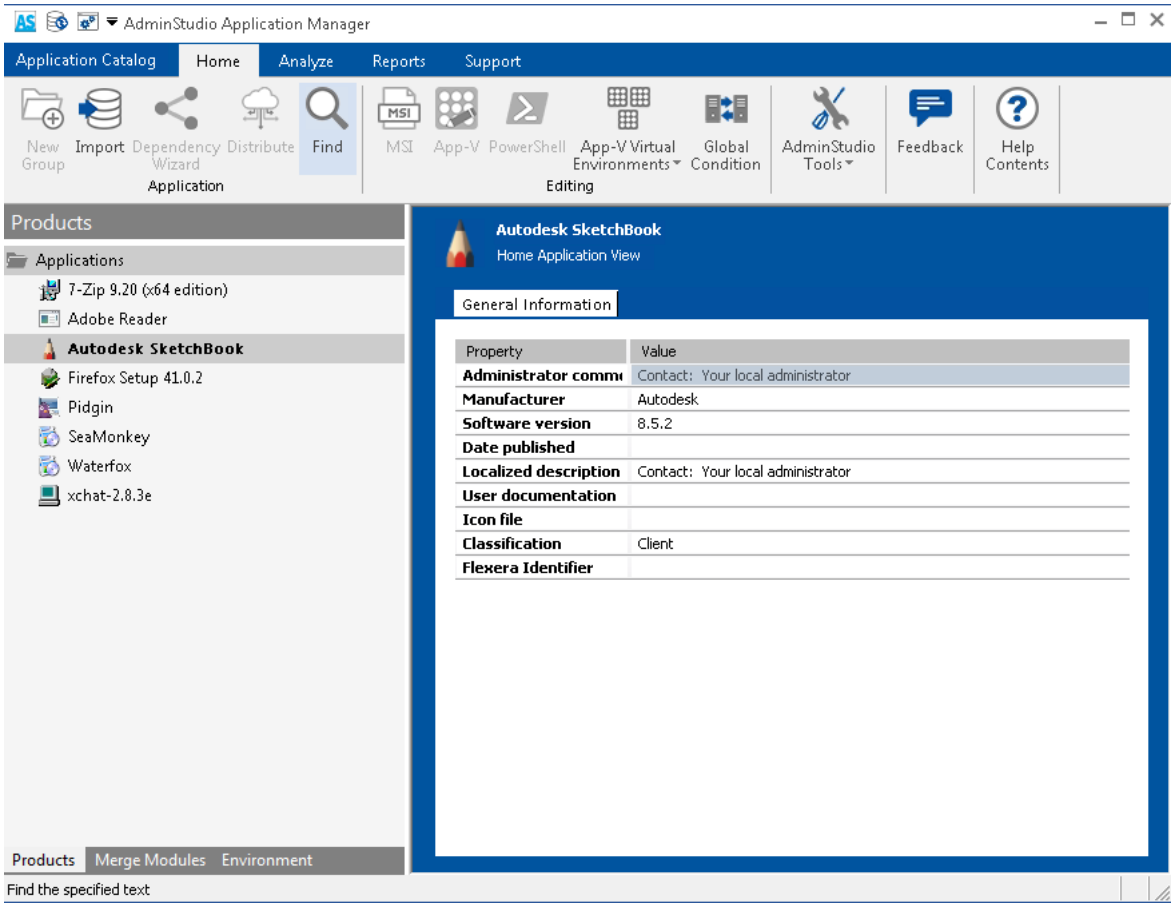
- Viewing Imported macOS Desktop Application Metadata
- Customizing Apple Installer Package PKG Installer Settings
- Viewing Bundled Packages of macOS PKG and DMG Files

Viewing Imported macOS Desktop Application Metadata

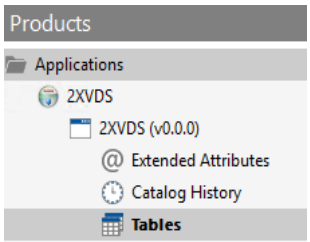
AdminStudio automatically extracts metadata when importing a macOS desktop application (.pkg file, .dmg file, or link to Mac App Store app). To view the metadata of a macOS desktop application, perform the following steps:

Task ***To view macOS desktop application metadata:***

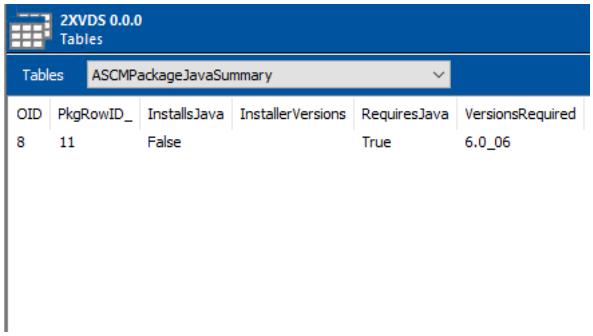
1. In Application Manager, open the **Home** tab.
2. In the tree, select the application node of a macOS desktop application. The **General Information** tab of the **Application View** opens.



3. Review the properties, as described in General Information Tab.
4. In the tree, expand the mobile app's application node and its deployment type node to display the subnode icons, and select the **Tables** subnode.



The **Tables View** opens, listing metadata from the selected table.



5. From the **Tables** list in the toolbar, select one of the following tables:

macOS Desktop Application Type	Table Name
All types	`cstblPackage`
Apple disk image package (.dmg)	`csDmgCustomTable` `cstblSetupFiles`
Apple installer package (.pkg)	`csPkgCustomTable` `cstblSetupFiles`
Apple Mac App Store application	`csAppDeepLinkExtraInfo` `ASCMApplicationFeature`

These tables list the properties and values that AdminStudio has extracted during import. This metadata is used when these applications are tested for OS compatibility and best practices.

6. To view the results of the testing of macOS desktop applications, see Performing Compatibility, Best Practices, and Risk Assessment Testing and Viewing and Filtering Test Results.

7. This metadata along with test results are used to generate the reports displayed on the **Reports** tab. For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab.

Customizing Apple Installer Package PKG Installer Settings

Just as a Windows Installer package can be customized by adding a transform file, an Apple installer package (`.pkg`) can be customized by editing an XML file that is embedded within it. In AdminStudio 2016, the settings defined in a `.pkg` file's embedded XML file are displayed on the new **PKG Installer Choices** tab of the package's **Home Deployment Type** view.

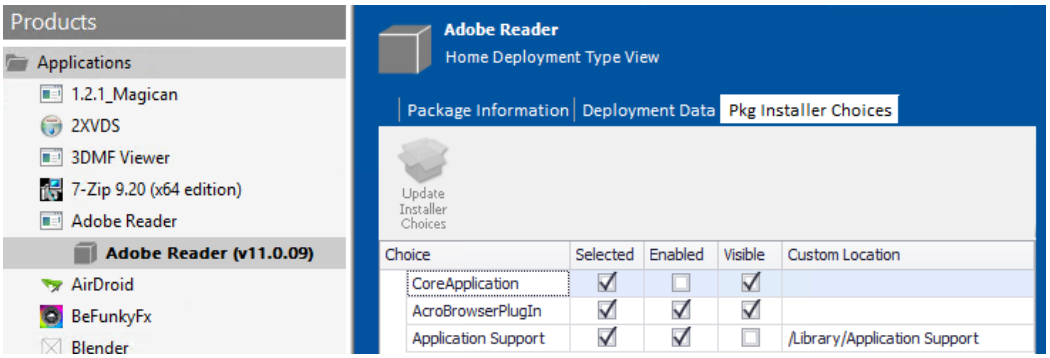


Figure 8: New PKG Installer Choices Tab of Home Deployment Type View for Mac PKG Installer

The **PKG Installer Choices** tab lists all settings that have been defined in that `.pkg` file's embedded XML settings file by the application manufacturer. To customize this installer (such as to prepare it for silent installation by Casper), you can make changes to the settings on this tab and then click **Update Installer Choices**. AdminStudio will then save your changes in the package's embedded settings file.

For each installer **Choice** listed on the **PKG Installer Choices** tab, the following options are available:

Option	Description
Visible	This option can be either selected or not selected: • **Selected**—This choice setting will be displayed in the installer. • **Not selected**—This choice setting will not be displayed in the installer.
Selected	This option can be either selected or not selected: • **Selected**—If this choice setting is displayed in the installer, its check box will be checked. • **Not selected**—If this choice setting is displayed in the installer, its check box will not be checked.
Enabled	This option can be either selected or not selected: • **Selected**—If this choice setting is displayed in the installer, it will be enabled. • **Not selected**—If this choice setting is displayed in the installer, it will be disabled.
Custom Location	If this choice setting explicitly permits the user to specify a user-defined installation path, the path entered in this field would populate the user-defined installation path when it is displayed in the installer.

Note • *Modifying the installer choices of an Apple installer package does not affect the digital signature of the package.*

Viewing Bundled Packages of macOS PKG and DMG Files

If an Apple installer package (.pkg) or disk image (.dmg) contains child packages bundled within it, those child packages will be listed on the **Bundled Packages** tab of the **Home Deployment Type** view.

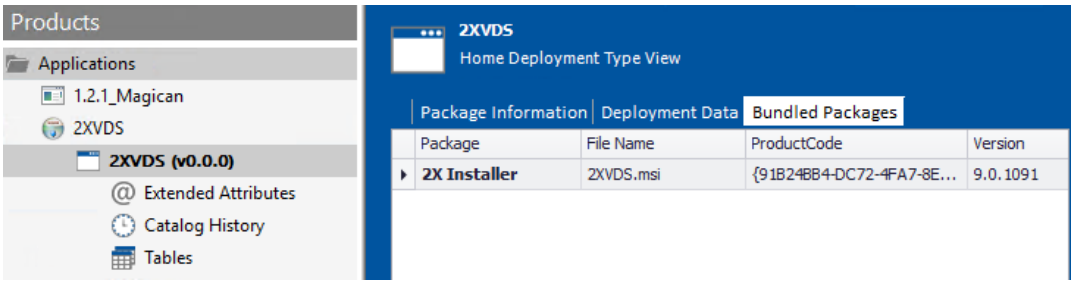


Figure 9: Bundled Packages Tab of Home Deployment Type View for Apple Installer Package (.pkg)

When the imported .dmg or .pkg file is tested, each of these child packages is also tested, and the combined test results are listed in Analyze, as described in Viewing Combined Test Results of Child Applications of PKG and DMG Installers.

Note • *AdminStudio will only inspect macOS package files one level deep. If a .dmg or .pkg package contains another .dmg or .pkg package bundled within it, that child package will not be inspected.*

Managing Mobile App Metadata

To enable you to successfully manage mobile apps in your enterprise—both local mobile apps and those from a public store—AdminStudio can extract and analyze metadata from those apps.

All of this metadata is needed for operating system compatibility, device compatibility, best practices, and risk and assessment testing, and for feature use reporting. By understanding a mobile app's configuration and property settings, AdminStudio can identify which apps might pose a security risk.

For Apple iOS mobile apps, you can import and view *Enterprise Policy Configuration* files so that you can determine the impact of enforcing those policies.

Information about mobile app metadata is presented in the following sections:

- About Mobile App Metadata
- Viewing Imported Mobile App Metadata
- Specifying the Path to Local iOS and Android Public Store Apps
- Managing iOS Enterprise Policy Configuration Files
- iOS Property Files (Info.plist) and iOS Enterprise Policy Files (*.plist)
- Viewing Mobile App Reports

About Mobile App Metadata

In enable you to successfully manage mobile apps in your enterprise—both local mobile apps and those from a public store—AdminStudio can extract and analyze metadata from those apps.

Some of this metadata is included in mobile app manifest files or within the application binaries and associated files. AdminStudio automatically extracts this information during mobile app import:

- **Apple iOS mobile apps (local)**—Many Apple iOS mobile apps have associated `Info.plist` (or property list) files which contain mobile app properties. AdminStudio captures the information in an iOS mobile app's associated `.plist` file.
- **Google Android mobile apps (local)**—Google Android apps include internal XML-based manifest files (`AndroidManifest.xml`) that contain additional mobile app metadata. AdminStudio captures this information during the import of an Android mobile app.
- **Binary scans**—AdminStudio scans the actual mobile app binary files during import to gather additional information.
- **Public store web sites**—For public store apps, AdminStudio captures all of the information that is displayed in the store about a mobile app.
- **Downloaded public store apps**—For iOS and Android public store mobile apps, you can choose to download the binaries of these apps to store locally (such as in a local iTunes Library or Google Android file share). This enables AdminStudio to scan those binary files, extract metadata, and store it in the Application Catalog along with the public store mobile app.

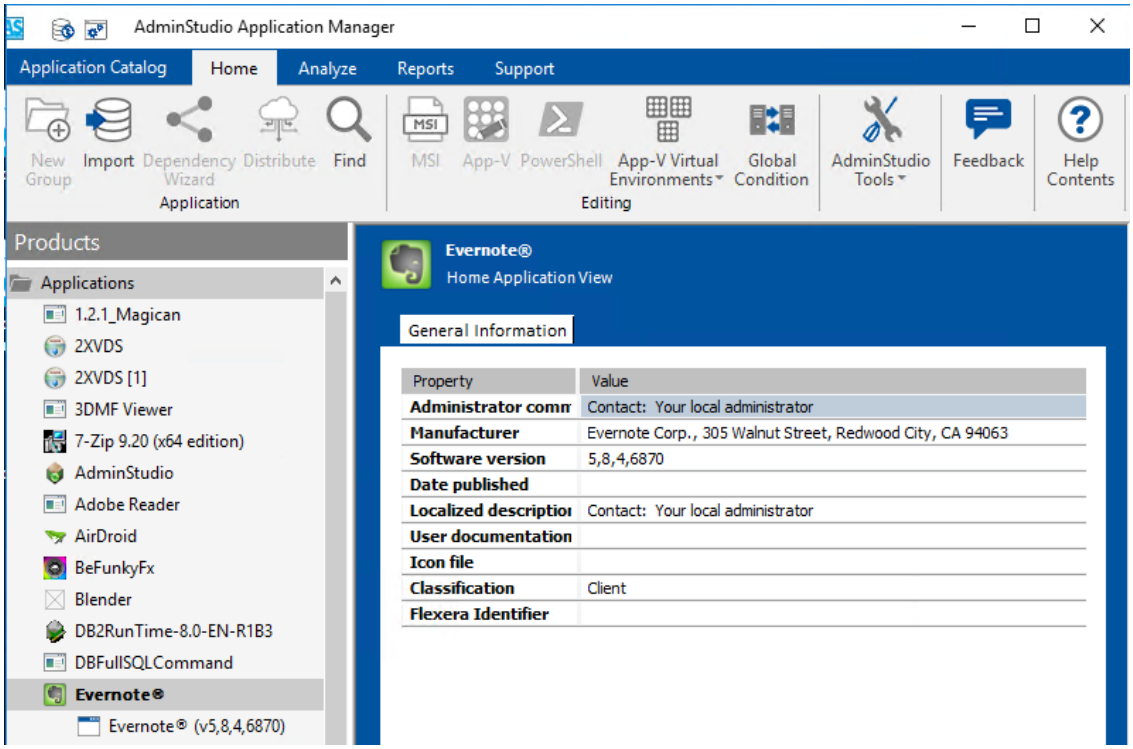
Viewing Imported Mobile App Metadata

AdminStudio automatically extracts mobile app metadata from multiple locations during mobile app import, as described in About Mobile App Metadata.

To view the metadata of a mobile app, perform the following steps:

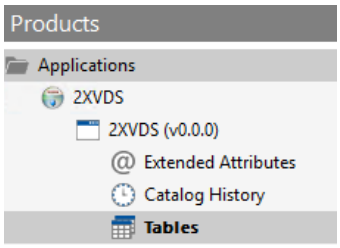
Task ***To view mobile app metadata:***

1. In Application Manager, open the **Home** tab.
2. In the tree, select the application node of a mobile app. The **General Information** tab of the **Application View** opens.

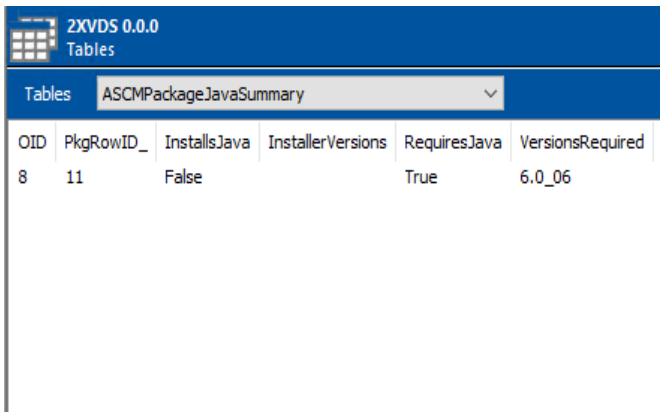


3. Review the properties, as described in General Information Tab.

4. In the tree, expand the mobile app's application node and its deployment type node to display the subnode icons, and select the **Tables** subnode.



The **Tables View** opens, listing metadata from the selected table.



5. From the Tables list in the toolbar, select one of the following tables:

Mobile App Type	Table Name
All types	ASCMApplicationFeature
Apple iOS (local)	csIpaCustomTable
Apple iOS (public store)	csIpaDeepLinkExtraInfo
Google Android (local)	csApkCustomTable
Google Android (public store)	csApkDeepLinkExtraInfo
Microsoft UWP app or Windows 8 app (local)	csAppxCustomTable
Microsoft Windows Store (public store)	csAppxDeepLinkExtraInfo

   These tables list the properties and values that AdminStudio has extracted during mobile app import. This metadata is used when mobile apps are tested for OS compatibility, best practices, and risk assessment.

6. To view the results of mobile app testing, see Performing Compatibility, Best Practices, and Risk Assessment Testing and Viewing and Filtering Test Results.

7. This metadata along with test results are used to generate the reports displayed on the **Reports** tab. For more information, see Viewing Mobile App Reports.

Specifying the Path to Local iOS and Android Public Store Apps

When you import a mobile app from a public store, AdminStudio extracts available metadata from the public store web site. However, to enable AdminStudio to examine the actual binary file of the mobile app so that it can extract even more metadata, you can specify the network directory where downloaded public store mobile app binary files are stored: a local iOS iTunes Library or Google Android file share.

If you specify this location, and then you import a public store mobile app that has also already been downloaded locally, AdminStudio will analyze the downloaded binary's data to discover more details about the features used by the app, which will result in more detailed test results.

On the **Plugin Options** tab of the Application Manager **Options** dialog box, you can specify the location in your network of mobile apps that you have already downloaded from a public store:

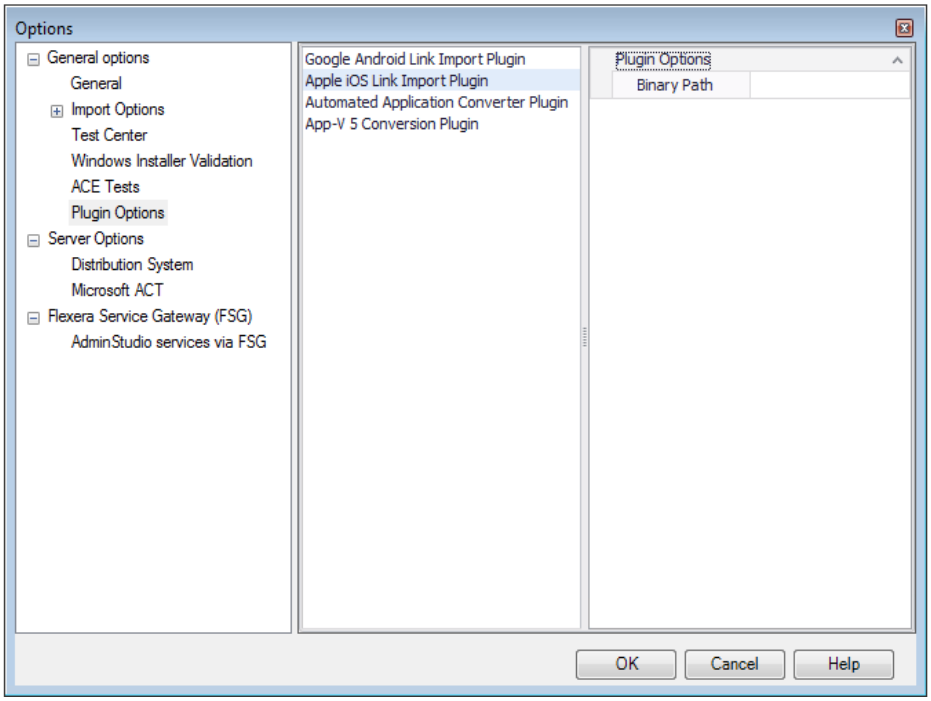
- **iTunes Library**—Apple iOS apps downloaded from the Apple iTunes Store
- **Google Android file share**—Google Android apps downloaded from the Google Play Store

Task ***To specify the path to Local iOS and Android public store apps:***

1. On the Application Catalog tab menu, click **Options**. The **Options** dialog box opens.

2. Open the **General options > Plugin Options** tab.

3. Select either **Apple iOS Link Import Plugin** or **Google Android Link Import Plugin**.



4. In the **Binary Path** field, enter the local path to the location of your iTunes Library or Google Android file share.
5. Click **OK**.

iOS Property Files (Info.plist) and iOS Enterprise Policy Files (*.plist)

There are two different types of files that can be associated with an iOS mobile app that have the `.plist` file extension:

- Information Property-List Files (Info.plist)
- Enterprise Policy Property List Files (*.plist or *.mobileconfig)

Information Property-List Files (Info.plist)

The `Info.plist` file contains critical information about the configuration of an iOS mobile app—such as iOS versions that are supported and device compatibility—which the operating system uses to interact with the app. This file is automatically created when the mobile app is compiled. This information is used by the App Store and by iOS to determine the app's capabilities and to locate key resources. Every app must include this file and it must be named `Info.plist`. AdminStudio uses the information in this file to perform testing. Also, this file is required by ConfigMgr (Formerly called as System Center Configuration Manager) to deploy a mobile app.

Enterprise Policy Property List Files (*.plist or *.mobileconfig)

An enterprise policy file is a device-level configuration profile that defines policies that implement security standards for a specific device. Enterprise policy files contain the configurations for iPhones, iPads, and iPod Touch devices, and policy settings (or rules) that specify what features of a mobile app or device an enterprise user is permitted to use. You can also enterprise policy files to distribute configuration information to a large number of devices throughout your enterprise, such as Wi-Fi or email settings. Enterprise policy files contain the following types of settings:

- Restrictions on device features, such as location services, GPS, or camera
- Wi-Fi settings
- VPN settings
- Email server settings
- Exchange settings
- LDAP directory service settings
- CalDAV calendar service settings
- Web clips
- Credentials and keys

For more information, see Managing iOS Enterprise Policy Configuration Files.

Managing iOS Enterprise Policy Configuration Files

Mobile apps are capable of accessing and exposing critical and sensitive corporate data, presenting challenges to enterprise security. To address this concern, AdminStudio is able to identify an organization's mobile apps that display behaviors that may introduce risk to corporate security and data privacy.

One method to implement these enterprise security standards for iOS mobile apps is through using Enterprise Policy Configuration files (`.mobileconfig` or `.plist`) By associating these policy files with iOS mobile devices, you are able to enforce and manage mobile policies.

AdminStudio enables you to import iOS policy configuration files, view their settings, and determine the policy compatibility of the iOS mobile apps in your Application Catalog.

- About Enterprise Policy Configuration Files
- Importing Enterprise Policy Configuration Files
- Viewing Enterprise Policy Configuration File Settings

About Enterprise Policy Configuration Files

Mobile apps are capable of accessing and exposing critical and sensitive corporate data, presenting challenges to enterprise security. One method of implementing enterprise security standards for iOS mobile apps is through using Enterprise Policy Configuration files (`.mobileconfig` or `.plist`). By associating these policy files with iOS mobile apps, you are able to enforce and manage mobile policies.

Settings in an Enterprise Policy Configuration File

Enterprise Policy Configuration files contain the configurations for iPhones, iPads, and iPod Touch devices and policy settings (or rules) that specify what features of a mobile app or device an enterprise user is permitted to use. Policy files contain the following types of settings:

- Device security policies and restrictions
- Wi-Fi settings

- VPN settings
- Email server settings
- Calendar settings
- Exchange settings
- LDAP directory service settings
- Credentials and certificates that permit iPhone and iPad devices to work with your enterprise systems

Types of Enterprise Policy Configuration Files

Enterprise Policy Configuration files are created using Apple utilities:

- **iPhone Configuration Utility**—Used for iOS Mountain Lion (10.8).
- **Apple Configurator**—Used for iOS Mavericks (10.9).

These XML-based policy files have the extension of `.mobileconfig` or `.plist`. If one of these policy files are downloaded to an iOS device, it would be listed on the device's **Settings > General** screen.

Tip • *Not all files associated with an iOS app that have a `.plist` extension are policy files. See iOS Property Files (Info.plist) and iOS Enterprise Policy Files (*.plist).*

Importing Enterprise Policy Configuration Files

Edition • *Support for mobile apps is included when you purchase AdminStudio Professional or Enterprise Edition with Mobile.*

You can import an iOS Enterprise Policy configuration file using the Import Wizard. An Enterprise Policy Configuration file (`.mobileconfig` or `.plist`) is used to manage the use of a mobile app in an enterprise. It contains information such as VPN configuration information, Wi-Fi settings, and email settings. It can also restrict the use of certain mobile app features that may introduce risk to corporate security and data privacy such camera use, location services, or GPS.

To import an iOS Enterprise Policy configuration file into the Application Catalog using the Import Wizard, perform the following steps:

Task ***To import an iOS Enterprise Policy configuration file:***

1. Open Application Manager.
2. Click on the **Environment** tab. The tree lists the Security Patches, OS Snapshots, and Enterprise Policy Configuration files that have already been imported into the Application Catalog.
3. In the tree, right-click on the **Enterprise Policy Configurations** group and click **Import** on the Application Manager ribbon. The **Enterprise Policy File Selection** panel opens.
4. Click **Browse** and select the Enterprise Policy Configuration file (`.mobileconfig` or `.plist`) that you want to import.
5. Click **Next**. The **Summary** panel opens.

6. Click **Next** to begin the import. The **Running the Import Panel** opens and progress messages are displayed.
7. When the import is complete, click **Finish** to close the wizard.

Viewing Enterprise Policy Configuration File Settings

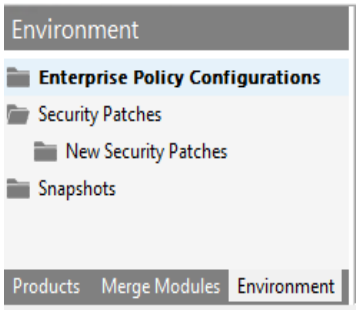
Edition • *Support for mobile apps is included when you purchase AdminStudio Professional or Enterprise Edition with Mobile.*

After you have imported an iOS Enterprise Policy Configuration file using the Import Wizard, you can view the settings in that configuration file on the **Enterprise Policy View** in Application Manager.

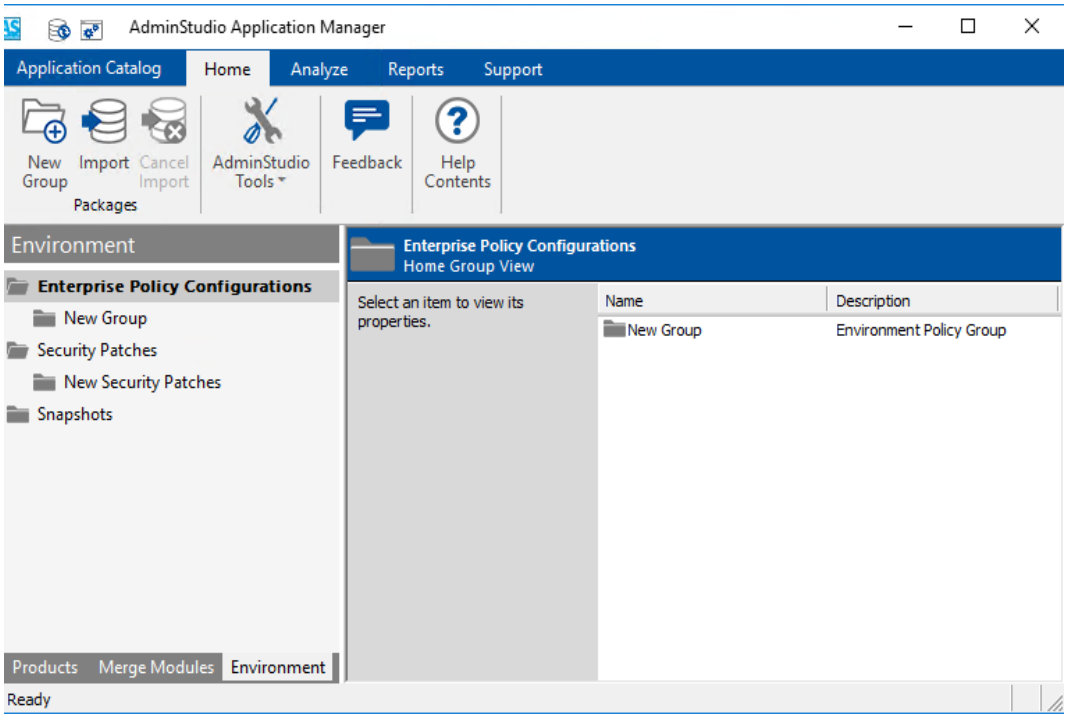
To view the settings of an iOS Enterprise Policy Configuration file, perform the following steps:

Task ***To view the settings of an iOS Enterprise Policy Configuration file:***

1. Click **Home** tab on the Application Manager ribbon.
2. Click on the **Environment** tab. The tree lists the Security Patches, OS Snapshots, and Enterprise Policy Configuration files that have already been imported into the Application Catalog.



3. In the tree, select an Enterprise Policy Configuration file. The **Enterprise Policy View** opens.



4. Scroll down the list to view all settings.

Viewing Mobile App Reports

AdminStudio offers extensive reporting on mobile apps. Using the metadata extracted during mobile app import, operating system compatibility, device compatibility, best practices, and risk and assessment testing is done, and the results of that testing can be viewed on the **Reports** tab. In addition, reports on mobile app feature use and on the impact of enforcing enterprise policies can also be viewed.

- Viewing Mobile App Analysis and Test Result Reports
- Viewing iOS Enterprise Policy Compatibility Reports

Viewing Mobile App Analysis and Test Result Reports

When a mobile app is imported, AdminStudio captures metadata for each app, as described in About Mobile App Metadata. When mobile apps are tested for operating system compatibility, best practices, and risk assessment—as described in Performing Compatibility, Best Practices, and Risk Assessment Testing, additional metadata is generated. Reports displaying this metadata for Apple iOS and Google Android mobile apps can be viewed in on the **Reports** tab.

On the **Reports** tab, the following summary reports are available:

Table 7-14 • Mobile App Reports on the Reports Tab

Report	Description
iOS/Android Mobile Dashboard	Provides summary charts of the major mobile app reports. Click to open more detailed reports.

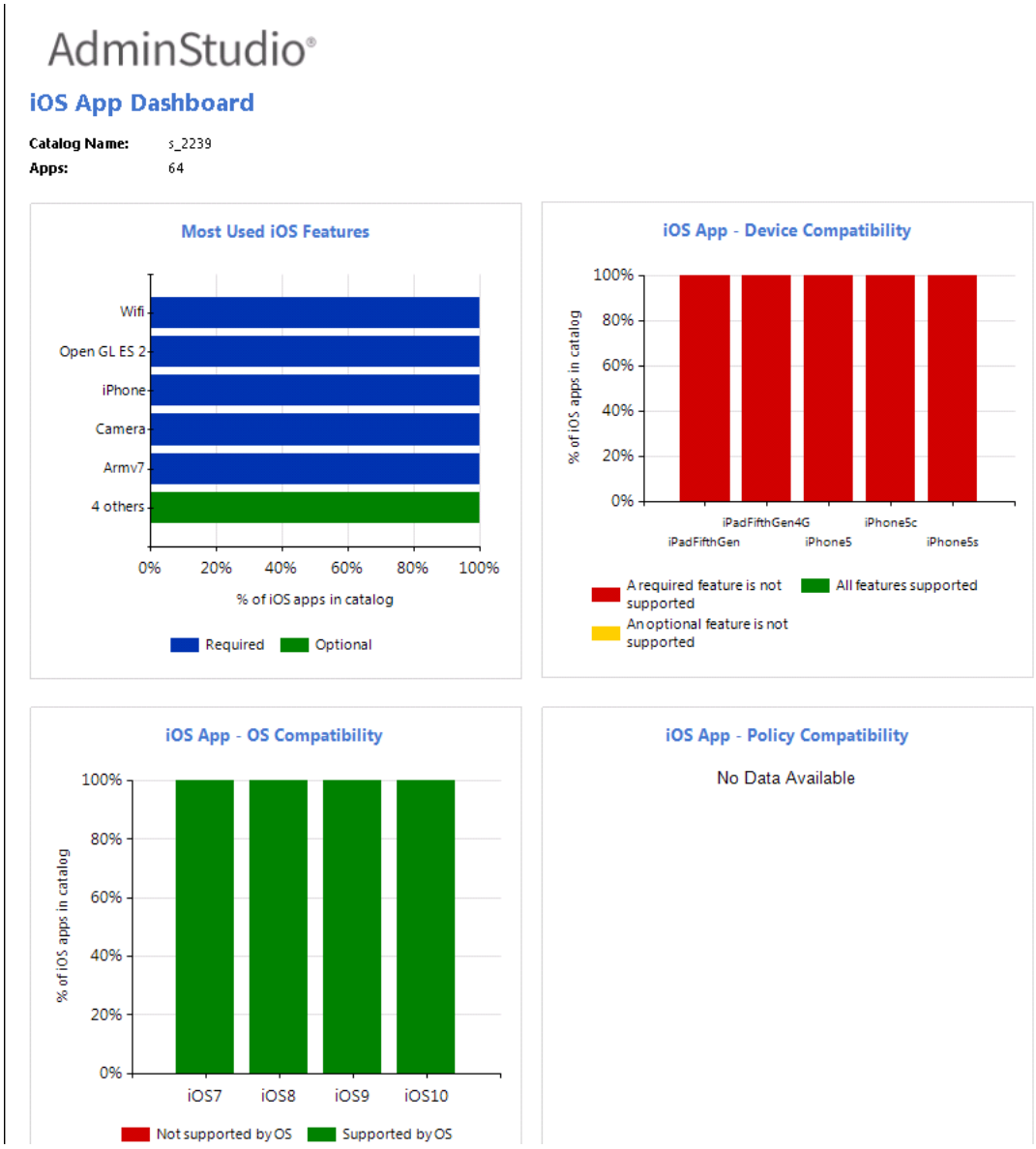
Table 7-14 • Mobile App Reports on the Reports Tab

Report	Description
iOS/Android App Details	Provides a combined view of the results of the analysis of feature use, feature compatibility with devices, and OS compatibility. Select the app to view by making a selection from the **Choose an app** list in the toolbar.
iOS/Android App Feature Use	The summary report lists the features that are being used by mobile apps in the Application Catalog and indicates the percentage of those apps that consider each feature to be either optional or required. Click on a bar segment to open more detailed reports. On **Feature Use** detail reports, you can select the feature that you want to view by clicking **Options** in the toolbar and making a selection from the list.
iOS/Android App - Device Compatibility	The summary report shows the compatibility of mobile apps on each of its operating system's devices. Each stacked bar indicates the percentage of mobile apps that fall into each of three categories: all features supported, an optional feature is not supported, or a required feature is not supported. Click on a bar segment to view **Device Compatibility** detail reports.
iOS/Android App - OS Compatibility	Shows the compatibility of mobile apps on each of its operating systems. Each stacked bar indicates the percentage of mobile apps that fall into one of two categories: supported by OS or not supported by OS. Click on a bar segment to view **OS Compatibility** detail reports.
iOS App - Policy Compatibility	The summary report shows the feature compatibility of iOS mobile apps on iOS devices for each Enterprise Policy in the Application Catalog. Each stacked bar indicates the percentage of iOS apps that fall into each of three categories: all features supported, an optional feature is not supported, or a required feature is not supported. Click on a bar segment to view **Policy Compatibility** detail reports. On the **Policy Compatibility** detail reports, you can choose to view additional policies by making a selection from the **Choose options** list in the toolbar.
iOS Best Practices and Risk Assessment	Summary reports shows the results of Apple iOS best practices and risk assessment testing to determine a mobile app's readiness for deployment. Mobile apps are assigned one of the following statuses: ready, warning, errors, not tested. Click on the pie chart segments to view detail reports.

To view mobile app analysis and test result reports, perform the following steps.

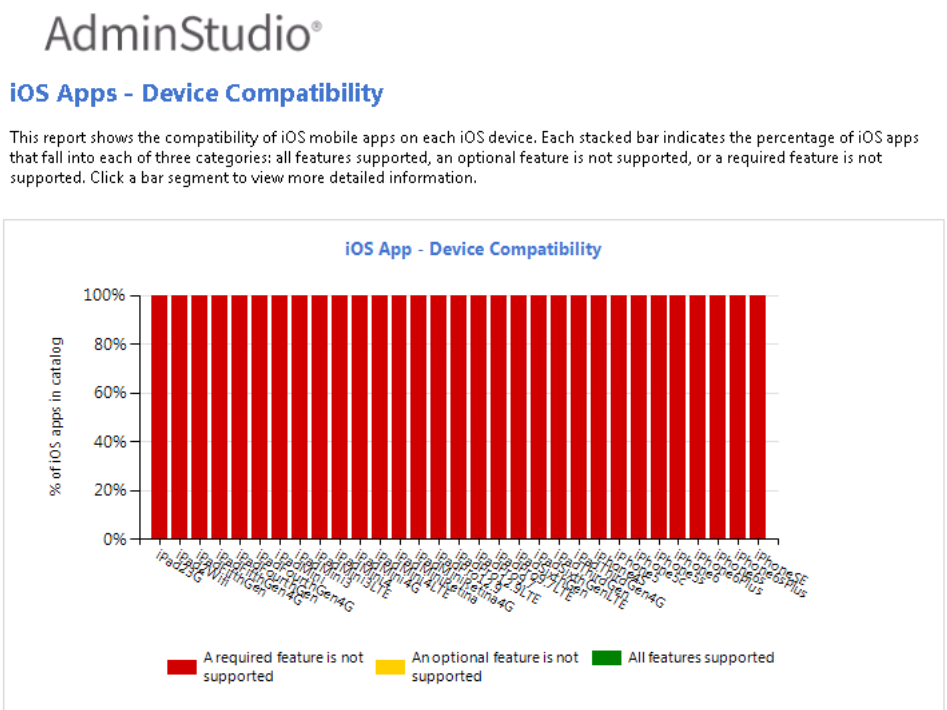
Task ***To view mobile app analysis and test results reports:***

1. Open the **Reports** tab.
2. From the toolbar, select **Mobile > iOS or Android > iOS or Android Mobile Dashboard**. The iOS/Android App Dashboard report opens, listing the most used features, device compatibility, OS compatibility, and policy compatibility summaries.



3. Click on any of the bar segments to view more detailed reports. For example, below is the **iOS Compatibility** detail report for iOS 6.

4. From the toolbar, select **Mobile > iOS or Android > iOS or Android Device Compatibility**. The iOS/Android **Device Compatibility** report opens, listing a summary of the compatibility of the mobile apps in the Application Catalog on each device.

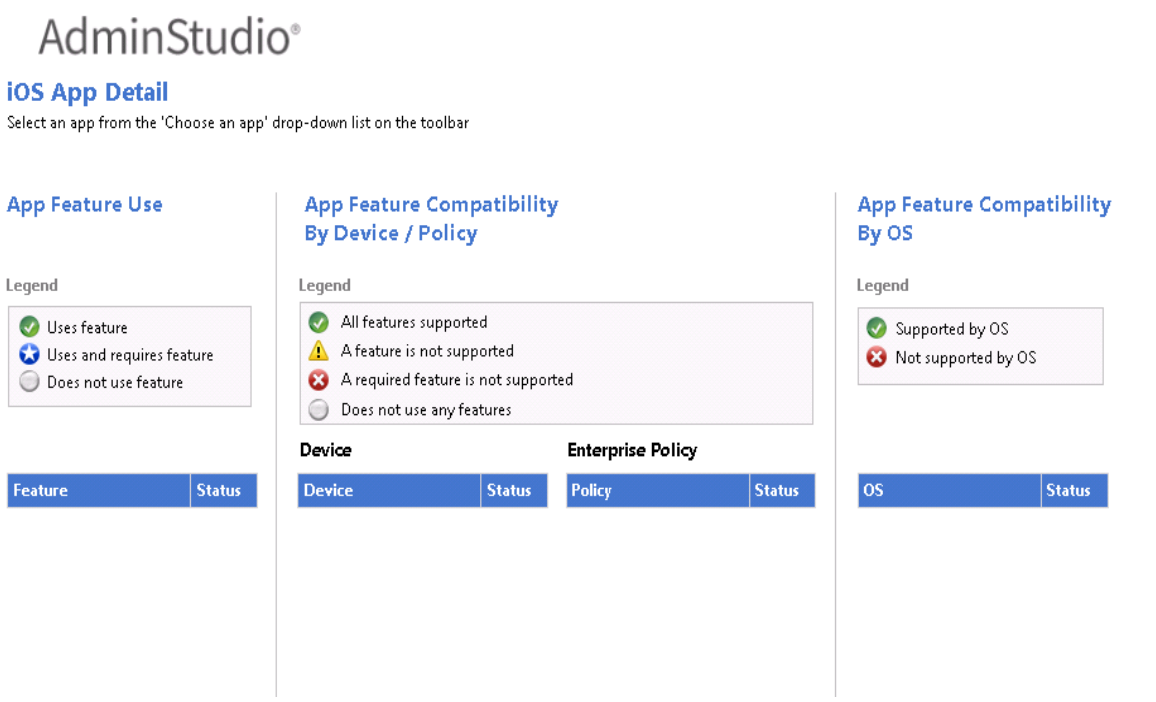


5. Click on any of the bar segments to view a **Device Compatibility** detail report. For example, below is a **Device Compatibility** detail report for an iOS device (iPadThirdGen). It displays an icon to indicate the compatibility of each iOS mobile app on this device.

Note • *On the **Device Compatibility** detail report, you can click on the icons in the device column to view more detailed reports.*

6. On some reports, you can make a selection in the toolbar to control the data that is being displayed in the report.

7. From the toolbar, select **Mobile > iOS or Android > iOS or Android App Details**. On the **iOS/Android App Detail** report, you can select the mobile app that you want to view by making a selection from the **Choose an app** list in the toolbar.

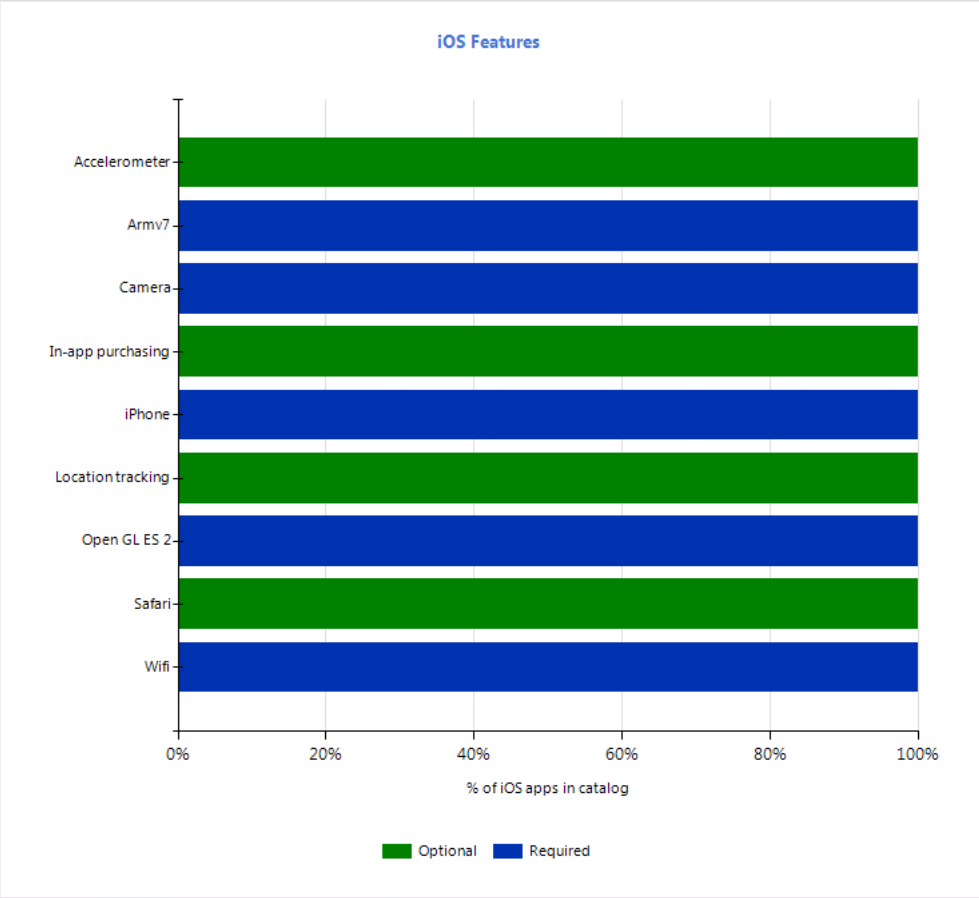


8. From the toolbar, select **Mobile > iOS or Android > iOS or Android Feature Use**. The iOS Feature Use report opens, listing the features that are being used by mobile apps in the Application Catalog, and indicates the percentage of those apps that consider each feature to be either optional or required.

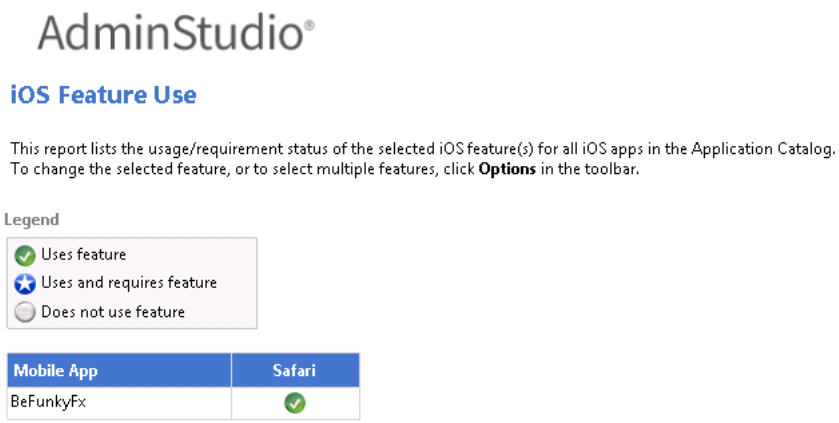
AdminStudio®

iOS Feature Use

This report lists the iOS features that are being used by iOS apps in the Application Catalog, and indicates the percentage of those apps that consider each feature to be either optional or required.



9. Click on any of the bar segments to view a **Feature Use** detail report. For example, below is a **Feature Use** detail report for a feature (in-app purchasing). There is an icon in the feature column to indicate the usage/requirement status of this feature on each mobile app in the Application Catalog.



10. To select the feature that you want to view on the **Feature Use** detail report, click **Options** in the toolbar and select a feature from the list.

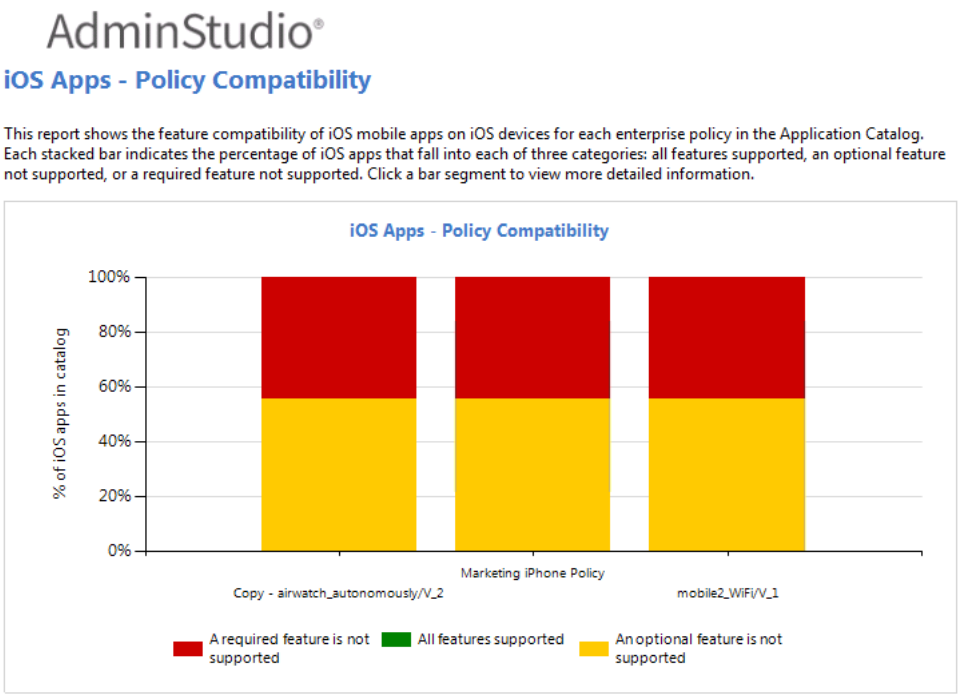
Viewing iOS Enterprise Policy Compatibility Reports

For iOS mobile apps, you can import Enterprise Policy Compatibility files and then view the impact of enforcing those policies on specific mobile apps, as described in Managing iOS Enterprise Policy Configuration Files.

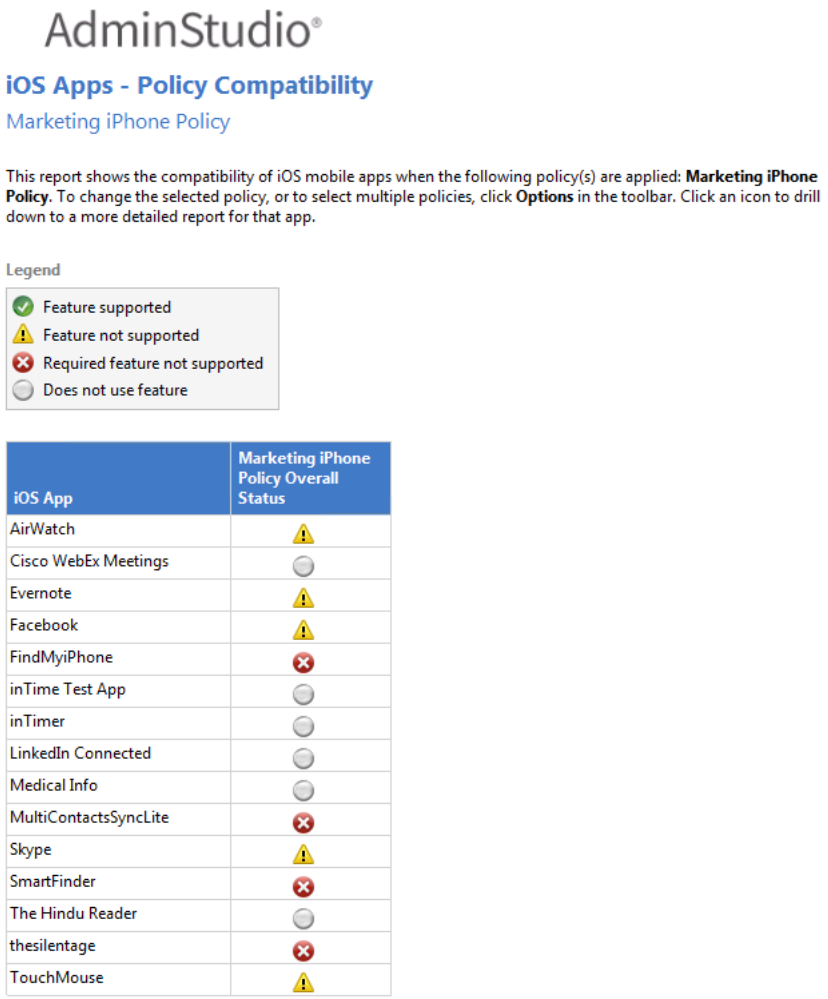
To view iOS Enterprise Policy Compatibility reports on the Reports tab, perform the following steps:

Task ***To view iOS Enterprise Policy Compatibility reports:***

1. Open the **Reports** tab.
2. From the toolbar, select **Mobile > iOS > iOS App - Policy Compatibility.** The **iOS Apps - Policy Compatibility** report opens, displaying the feature compatibility of iOS mobile apps on iOS devices for each enterprise policy that has been imported.



3. Click one of the bar segments to view a **Policy Compatibility** detail report. For example, below is a policy compatibility report for an iOS policy named Marketing iPhone Policy.



AdminStudio®

iOS Apps - Policy Compatibility

Marketing iPhone Policy

This report shows the compatibility of iOS mobile apps when the following policy(s) are applied: **Marketing iPhone Policy**. To change the selected policy, or to select multiple policies, click **Options** in the toolbar. Click an icon to drill down to a more detailed report for that app.

Legend

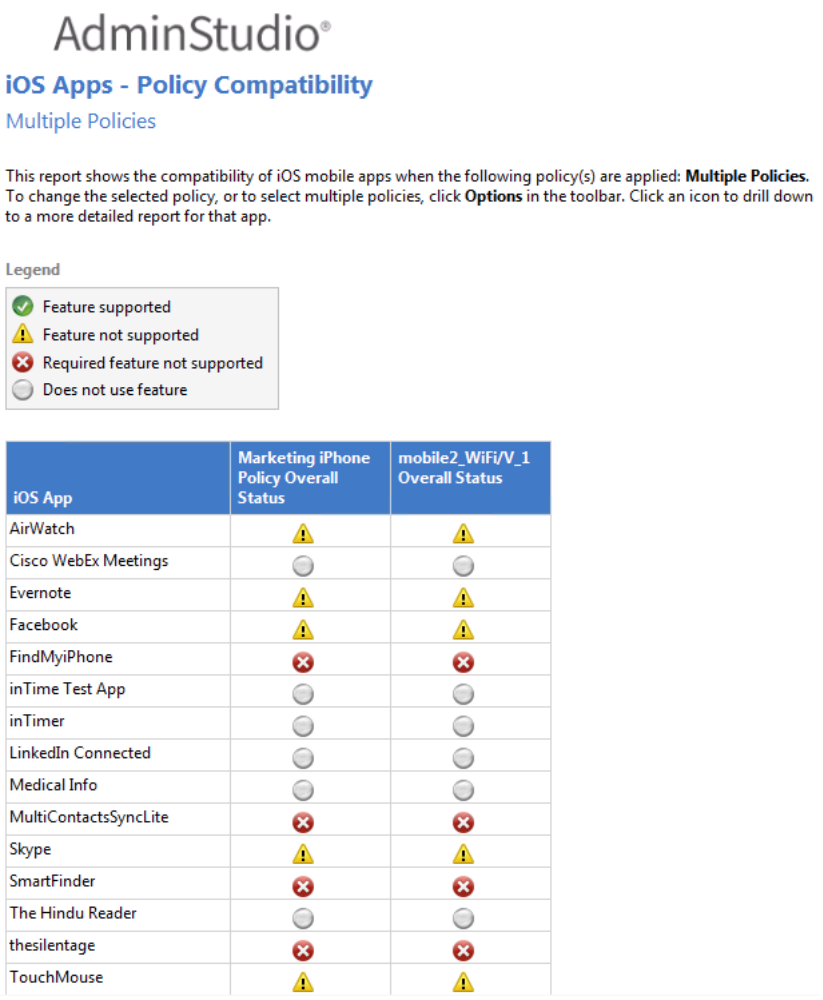
- Feature supported
- Feature not supported
- Required feature not supported
- Does not use feature

iOS App	Marketing iPhone Policy Overall Status
AirWatch	⚠
Cisco WebEx Meetings	○
Evernote	⚠
Facebook	⚠
FindMyiPhone	✖
inTime Test App	○
inTimer	○
LinkedIn Connected	○
Medical Info	○
MultiContactsSyncLite	✖
Skype	⚠
SmartFinder	✖
The Hindu Reader	○
thesilentage	✖
TouchMouse	⚠

This report displays an icon to indicate the compatibility of each iOS mobile app with the selected policy.

- Feature supported
- Feature not supported
- Required feature not supported
- Does not use feature

4. To display additional policies in this report, click **Options** in the toolbar to open the **Chose options** dialog box and select additional policies. Additional columns will be added to the table:



AdminStudio®

iOS Apps - Policy Compatibility

Multiple Policies

This report shows the compatibility of iOS mobile apps when the following policy(s) are applied: **Multiple Policies**. To change the selected policy, or to select multiple policies, click **Options** in the toolbar. Click an icon to drill down to a more detailed report for that app.

Legend

- Feature supported
- Feature not supported
- Required feature not supported
- Does not use feature

iOS App	Marketing iPhone Policy Overall Status	mobile2_WiFi/V_1 Overall Status
AirWatch	⚠	⚠
Cisco WebEx Meetings	○	○
Evernote	⚠	⚠
Facebook	⚠	⚠
FindMyiPhone	✖	✖
inTime Test App	○	○
inTimer	○	○
LinkedIn Connected	○	○
Medical Info	○	○
MultiContactsSyncLite	✖	✖
Skype	⚠	⚠
SmartFinder	✖	✖
The Hindu Reader	○	○
thesilentage	✖	✖
TouchMouse	⚠	⚠

Managing App Portal Application Information

AdminStudio can communicate with App Portal and the FlexNet Manager Suite via the Flexera Service Gateway, as described in Integrating with Other Flexera Applications via the Flexera Service Gateway.

When AdminStudio is integrated with App Portal, when you publish a supported application from AdminStudio to System Center 2012 Configuration Manager, Symantec Altiris Server, or Casper Suite Server, you can choose to have a catalog item for that application automatically created in App Portal. And, if you are integrated with FlexNet Manager Suite, automatic license management can also be performed.

On the **App Portal Information** tab of the **Application View**, you need to specify the following:

- **Whether to create a catalog item**—Specify whether you want a new App Portal catalog item to be created when the supported application is published to System Center 2012 Configuration Manager, Symantec Altiris Server, or Casper Suite Server. To indicate that you want to create an App Portal catalog item upon publish, you need to:
  - **Select Notify option**—Click on the browse button in the **Categories** field to open the **Categories** dialog box, and select the **Notify Flexera App Portal on publish of current Application** option.
  - **Specify categories**—On the **Categories** dialog box, select an App Portal category or categories for this new catalog item.

- **Descriptions**—Enter the **Brief Description** and **Long Description** that will describe the new catalog item in App Portal.
- **Template**—Optionally, select an App Portal template that you want to base the new catalog item on.
- **Keywords**—Optionally, enter keywords that you wan the new catalog item to be searchable by in the App Portal **Browse Home** tab.

If AdminStudio is integrated with FlexNet Manager Suite, you should also open the **General Information** tab of the **Application View** and determine whether a Flexera Identifier is associated with this application. If not, you need to search for an application's Flexera Identifier, which is used when integrating with FlexNet Manager Suite to perform automatic license management. For more information, see Managing an Application's Flexera Identifier.

Instructions for performing these tasks are included in this section:

- Enabling Automatic Creation of App Portal Catalog Item
- Setting Brief Description and Long Description
- Specifying Catalog Item Categories
- Selecting an App Portal Template
- Specifying Catalog Item Keywords
- Troubleshooting: App Portal Catalog Item Not Created Upon AdminStudio Publication

Enabling Automatic Creation of App Portal Catalog Item

You can automatically create a new catalog item in App Portal each time either an application is published to System Center 2012 Configuration Manager, Symantec Altiris Management Suite, or Casper Suite Server, or when a package is published to ConfigMgr (Formerly called as System Center Configuration Manager). The instructions for enabling this feature vary depending upon whether you are publishing a package or an application.

- Catalog Item Creation When Publishing an Application
- Catalog Item Creation When Publishing a Package to ConfigMgr (Formerly called as System Center Configuration Manager)

Catalog Item Creation When Publishing an Application

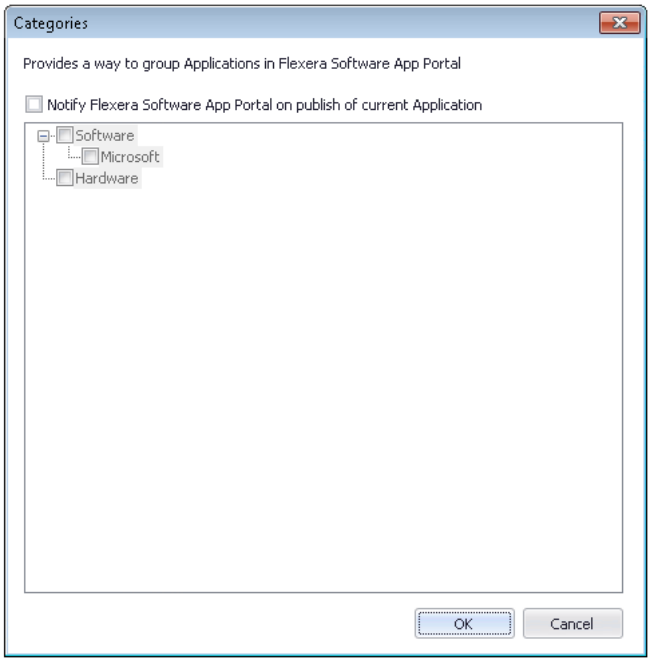
If you want a new catalog item to be created in App Portal when an application is published to System Center 2012 Configuration Manager, Symantec Altiris Management Suite, or Casper Suite Server, you need to select an option on the **Categories** dialog box, which is accessed from the **App Portal Information** tab of the **Application View**.

Note • *App Portal does not support the creation of catalog items for Mac App Store apps.*

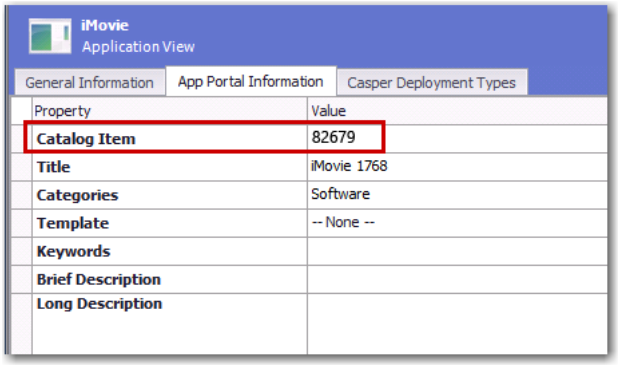
Task ***To enable automatic creation of App Portal Catalog item upon publication:***

1. Open the Application Manager **Home** tab.
2. Select an application in the tree. The **Application View** opens.

3. Select the **App Portal Information** tab. The **App Portal Information** tab opens.
4. Next to the **Categories** field, click the browse button. The **Categories** dialog box opens.



5. Select the **Notify Flexera App Portal on publish of current Application** option.
6. Select an App Portal category or categories, as described in Specifying Catalog Item Categories.
7. Click **OK**. When this application is published to System Center 2012 Configuration Manager, Symantec Altiris Management Suite, or Casper Suite Server, an App Portal catalog item will be automatically created, and the **Catalog Item** field on the **App Portal Information** view will display the App Portal Catalog ID for that catalog item:



Catalog Item Creation When Publishing a Package to ConfigMgr (Formerly called as System Center Configuration Manager)

If both AdminStudio and App Portal are connected via the Flexera Service Gateway, when you publish a *package* from AdminStudio to System Center 2007 or 2012 Configuration Manager, a catalog item for that package should automatically be created in App Portal (in the default catalog category).

However, a catalog item will be created for this package in App Portal *only* if the **Default Category** field in App Portal is set to a valid category in App Portal. If the **Default Category** field on the App Portal **Settings > Web Site > General** tab is set to `-Select-` instead of to a valid category, an App Portal catalog item will not be created.

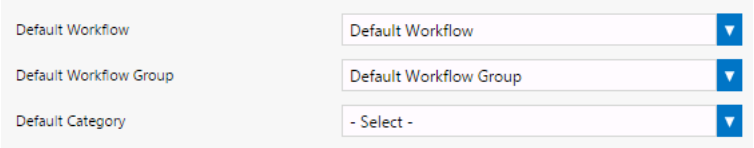


Figure 7-1: Default Category Field on Web Site > General Tab

If an App Portal catalog item fails to be created when you publish an AdminStudio package to ConfigMgr (Formerly called as System Center Configuration Manager), make sure that a category is selected in the **Default Category** list on the App Portal **Settings > Web Site > General** tab.

Setting Brief Description and Long Description

On the **App Portal Information** tab of the **Application View**, you can specify both a **Brief Description** and a **Long Description**, which will be used to describe the application on the App Portal's storefront.

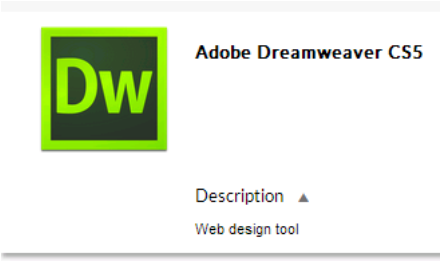


Figure 7-2: Catalog Item's Brief Description on the Browse Home tab

Figure 7-3: Catalog Item's Long Description Displayed in Detail View (Opened by Clicking on Catalog Item Name)

On the **App Portal Information** tab, you can also preview the **Title** that App Portal will assign to this application's catalog item.

Task ***To set the Brief Description and Long Description fields:***

1. Open the Application Manager **Home** tab.
2. Select an application in the tree. The **Application View** opens.
3. Select the **App Portal Information** tab. The **App Portal Information** tab opens.
4. Review the text in the read-only **Title** field. It is a concatenation of the text in the **Manufacturer** and **Version** fields on the **General Information** tab of the **Application View**, as well as the application name displayed in the Application Manager tree. This is a preview of the name that App Portal will assign to this catalog item.

   Note • *To edit this App Portal catalog item title, launch App Portal and edit the* ***Title*** *field on the* ***General > Title & Description*** *tab of the* ***Catalog Item Properties*** *dialog box for this catalog item.*

5. In the **Brief Description** field, enter the text that you want to display under this catalog item **Title** in the App Portal storefront. This text should briefly describe the purpose of the catalog item.
6. In the **Long Description** field, enter a more detailed description of this catalog item (optional).

   Note • *This field is referred to as the* ***Full Description*** *in App Portal.*

Specifying Catalog Item Categories

When an end user browses the App Portal catalog on the **Browse Home** tab, catalog items are organized into categories. In the image below, the **Software > Adobe** category contains two catalog items.

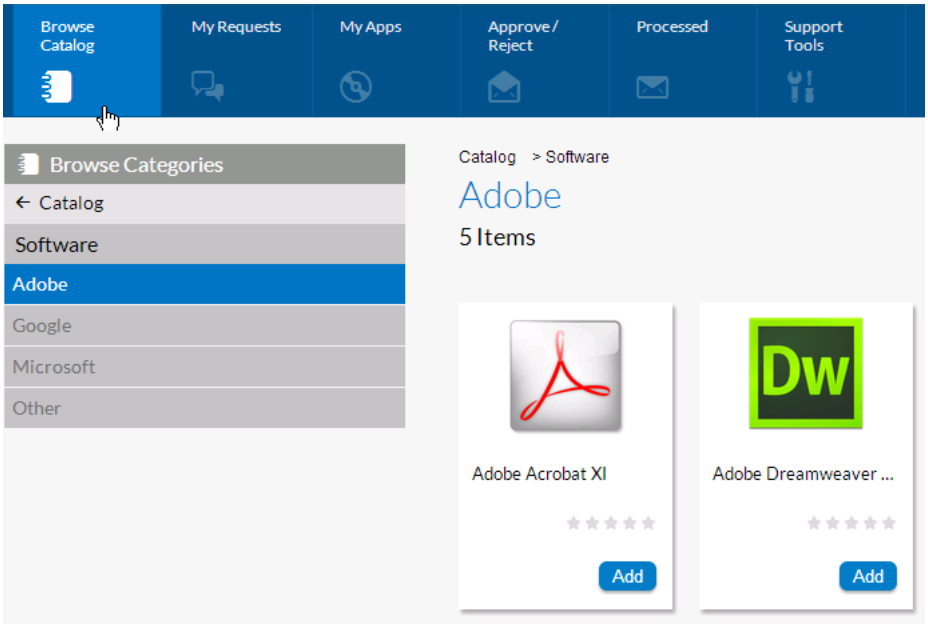


Figure 7-4: Categories Displayed on the App Portal Browse Home Tab

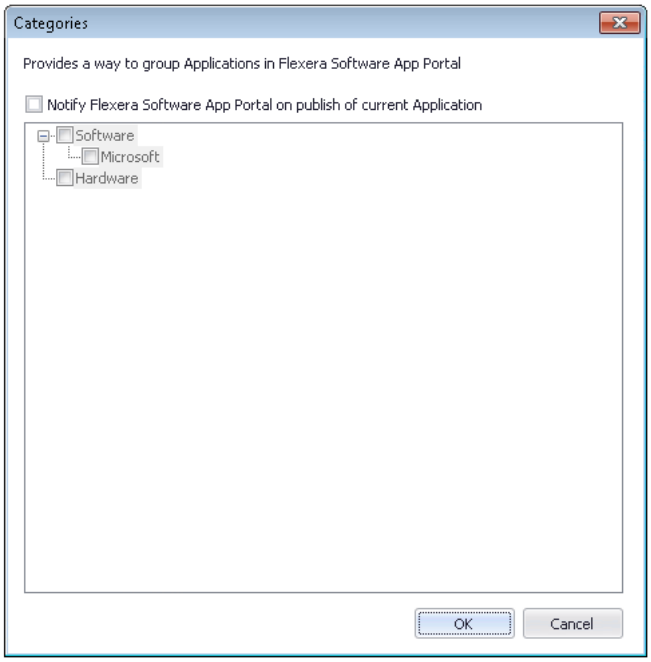
***Note** • Security permissions can be assigned to a category to control which users and groups are permitted to view and request the catalog items in that category. Also, category owners can be assigned to a category to give specific users permission to manipulate the catalog items in that category.*

When a catalog item is created in App Portal, a category must be specified. The catalog item that is created when you publish an application from AdminStudio to System Center 2012 Configuration Manager or Symantec Altiris Server will be placed in the category you selected.

You select a category or categories on the **Categories** dialog box, which is accessed from the **App Portal Information** tab of the **Application View**.

Task ***To select a catalog item's categories:***

1. Open the Application Manager **Home** tab.
2. Select an application in the tree. The **Application View** opens.
3. Select the **App Portal Information** tab. The **App Portal Information** tab opens.
4. Next to the **Categories** field, click the browse button. The **Categories** dialog box opens and displays a list of all of the categories that have been defined in App Portal.



5. Select the **Notify Flexera App Portal on publish of current Application** option (if it is not already selected). The categories are enabled.
6. Select one or more categories.
7. Click **OK**. When this application is published to System Center 2012 Configuration Manager or Symantec Altiris Server, an App Portal catalog item will be automatically created and will be placed in the category or categories you selected.

Selecting an App Portal Template

On the App Portal Catalog Item Properties dialog box, It is possible to set hundreds of different properties on a catalog item.

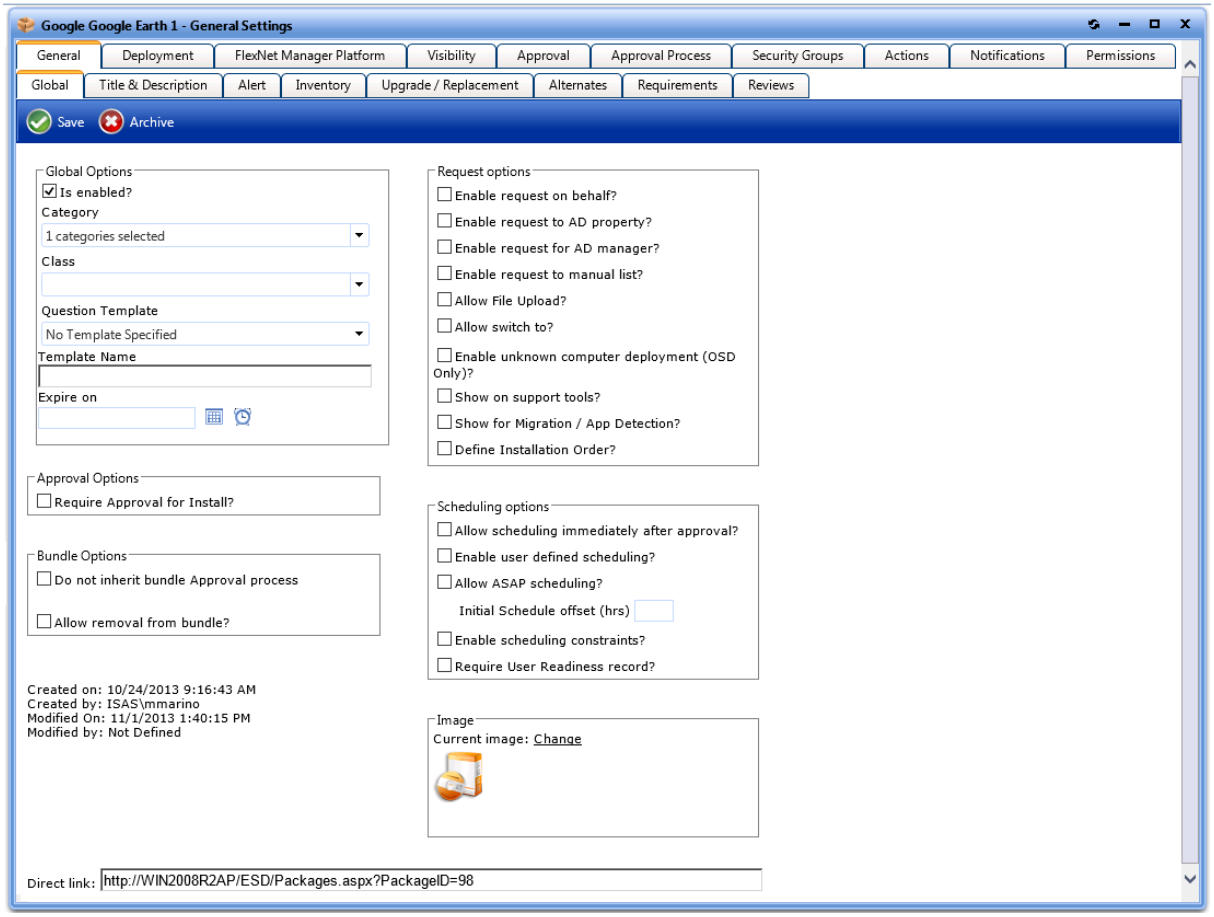


Figure 7-5: App Portal Catalog Item Properties Dialog Box

For catalog items that require a complex set of properties, it would be beneficial to create an App Portal template that contains all of those settings and properties. Then, whenever a new catalog item is created, properties and settings can be automatically loaded by selecting that template.

If templates have been created in App Portal, you can assign one of those templates to an application on the **App Portal Information** tab of the **Application View**.

Task ***To select an App Portal Template:***

1. Open the Application Manager **Home** tab.
2. Select an application in the tree. The **Application View** opens.
3. Select the **App Portal Information** tab. The **App Portal Information** tab opens.
4. Next to the **Template** field, select a template from the list.

5. Make sure that publishing to App Portal has been enabled, as described in Enabling Automatic Creation of App Portal Catalog Item, and that at least one category has been selected.
6. Click **OK**. When this application is published to System Center 2012 Configuration Manager or Symantec Altiris Server, an App Portal catalog item will be automatically created and all of the properties and settings defined in the selected template will be assigned.

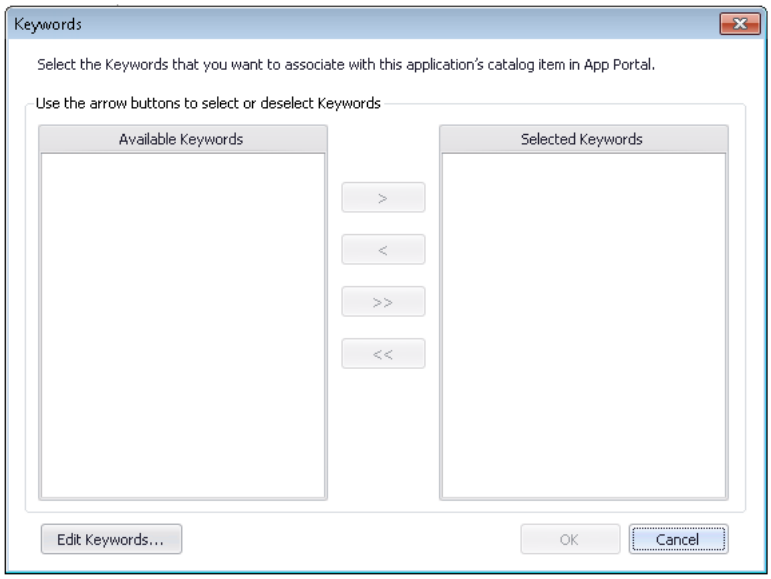
Specifying Catalog Item Keywords

When an App Portal end user performs a search on the **Browse Home** tab, App Portal performs a search on not only the **Title**, **Brief Description**, and **Full Description** fields, but also on any **Keywords** that have been specified for that catalog item.

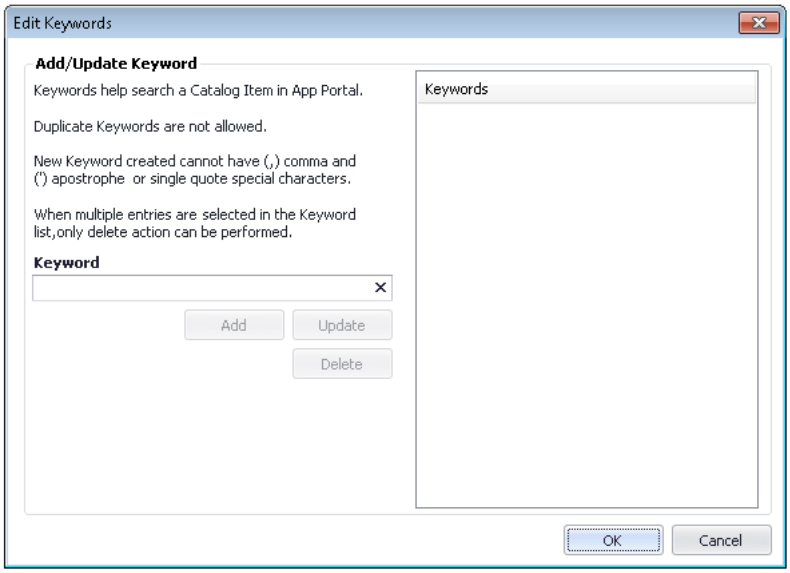
On the **App Portal Information** tab of the **Application View**, you can specify keywords for an application's App Portal catalog item.

Task ***To specify catalog item keywords:***

1. Open the Application Manager **Home** tab.
2. Select an application in the tree. The **Application View** opens.
3. Select the **App Portal Information** tab. The **App Portal Information** tab opens.
4. Next to the **Keywords** field, click the Browse button. The **Keywords** dialog box opens.



5. Click **Edit Keywords**. The **Edit Keywords** dialog box opens.



6. Enter a keyword in the **Keyword** box and click **Add**. The keyword is now listed in the **Keywords** list.

Important • *Keywords must be single words only. If you enter a multiple-word keyword, all words of the phrase will be ignored when a search is performed.*

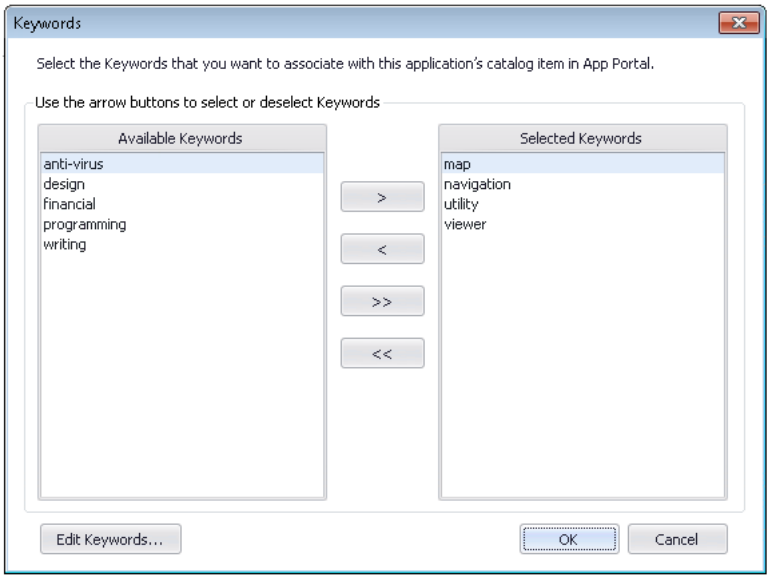
7. Repeat previous step until all desired keywords have been created.

Note • *To update an existing keyword, select it in the list, make edits in the **Keyword** box, and click **Update**.*

8. When you have finished adding keywords, click **OK**. The new keywords are now listed in the **Keywords** dialog box.

Note • *When you add a keyword on the **Edit Keywords** dialog box, it will be available to assign to any application in this Application Catalog.*

9. To add a keyword to an application, select the keyword in the **Available Keywords** list and then click the right arrow to move it to the **Selected Keywords** list.



10. When you have selected all of the desired keywords, click **OK**. The selected keywords are now listed in the **Keywords** field of the **App Portal Information** tab.

Troubleshooting: App Portal Catalog Item Not Created Upon AdminStudio Publication

There could be several reasons why an App Portal catalog item is not automatically created when AdminStudio publishes an application to ConfigMgr (Formerly called as System Center Configuration Manager) or Symantec Altiris Management Suite. One of the reasons could be if the App Portal settings were not specified correctly. The App Portal settings changed between AdminStudio 2013 and AdminStudio 2013 R2. Therefore, troubleshooting steps for both versions are described in this section.

- AdminStudio 2013 R2 or 2014: App Portal Settings Not Specified
- AdminStudio 2013: App Portal Default Category Not Specified
- Symantec Endpoint Protection Blocking Notification of App Portal

AdminStudio 2013 R2 or 2014: App Portal Settings Not Specified

If you are using AdminStudio 2013 R2 or later, an App Portal catalog item is automatically created when AdminStudio publishes an application *only if* the following App Portal settings on the **App Portal Information** tab of the **Application View** in Application Manager have been set:

- The **Categories** property must be specified.
- The **Notify Flexera App Portal on publish of current Application** option on the **Categories** dialog box must be selected.

For more information, see Enabling Automatic Creation of App Portal Catalog Item.

AdminStudio 2013: App Portal Default Category Not Specified

Note • *This also applies when using AdminStudio 2014, 2015, or 2016 to publish a* ***package*** *to System Center 2007 or 2012 Configuration Manager.*

If both AdminStudio (11.5 SP2 or 2013) and App Portal are connected via the Flexera Service Gateway, when you publish an application from AdminStudio to System Center 2012 Configuration Manager, a catalog item for that application should automatically be created in App Portal (in the default catalog category). Both the App Portal catalog item and the AdminStudio application will be identified by the same Flexera Identifier.

If a catalog item fails to be created, it may be because App Portal no longer has a **Default Category** specified. This can occur if the existing default category is deleted in App Portal. If the existing default category is deleted, the **Default Category** field on the **Settings > Web Site > General** tab is set to `-Select-`:

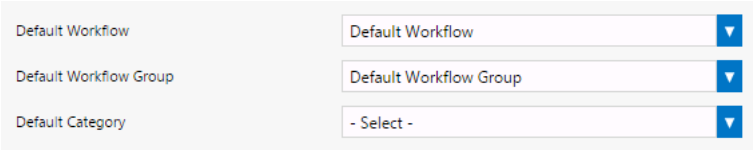


Figure 7-6: Default Category Field on Web Site > General Tab

In order for AdminStudio to automatically create an App Portal catalog item during publication, App Portal's default category must be set to a valid category. To attempt to resolve this issue, select an existing category from the **Default Category** list.

Note • *Starting with AdminStudio 2013 R2, you can choose whether or not to automatically create a catalog item for an application when you publish it to System Center 2012 Configuration Manager or Symantec Altiris Management Server. You can also specify the destination App Portal category for the new catalog item. These settings are made on the* ***App Portal Information*** *tab of the Application Manager* ***Application View****. For more information, see:*

- *Enabling Automatic Creation of App Portal Catalog Item*
- *Specifying Catalog Item Categories*

Symantec Endpoint Protection Blocking Notification of App Portal

In some instances, an App Portal catalog item is not created when AdminStudio publishes an application to ConfigMgr (Formerly called as System Center Configuration Manager).

Cause

This could be because Symantec Endpoint Protection blocked the notification of App Portal. If this is the case, the following error messages would be generated:

```
17:26:47 ERROR: AdminStudio.ESB.Integration.IntegrationService.LogException - NotifyAppPortalForGroup :
The ConnectionString property has not been initialized.

17:26:47 ERROR: AdminStudio.ESB.Integration.IntegrationService.LogException - NotifyAppPortalForGroup :
at System.Data.SqlClient.SqlConnection.PermissionDemand()

An anti-virus program (do not know which one yet) caused running tests in Analyze to fail. It prevented
extraction of CAB files from MSI files thus stopping the correct execution of tests
```

Resolution

To resolve this issue, try to disable SEP (Symantec Endpoint Protection).

Enabling Application Extended Attributes

If you want to record custom data for applications, you can edit and run a script that will define custom extended attributes and display those attributes on a new **Extended Attributes** tab of the **Application View**.

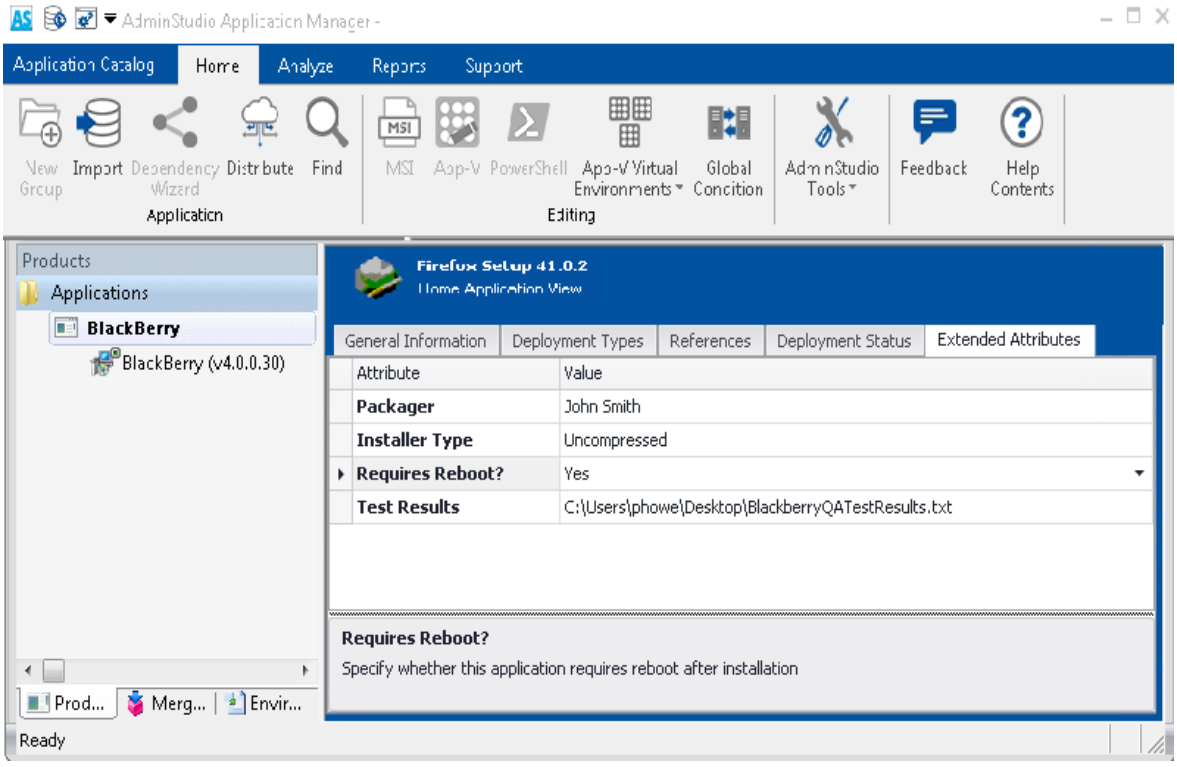


Figure 7-7: Extended Attributes Tab of the Application View

Information about enabling and viewing application extended attributes is provided in the following topics:

- Enabling the Extended Attributes Tab of the Application View
- Defining Application Extended Attributes
- Viewing and Editing Application Extended Attributes

Enabling the Extended Attributes Tab of the Application View

To enable the **Extended Attributes** tab of the **Application View**, you need to open a provided sample `ApplicationExtendedAttributes.SQL` script file, edit that script file to define your application attributes, and then run that SQL script on your Application Catalog.

After you edit and run the SQL script named `ApplicationExtendedAttributes.SQL` script, the extended attributes that you have defined are listed on the **Extended Attributes** tab of the **Application View**

Important • *The **Extended Attributes** tab will only be visible for applications imported into the Application Catalog after the `ApplicationExtendedAttributes.SQL` script is run.*

Task ***To enable the Extended Attributes tab of the Application View:***

1. Create a new Application Catalog.

   Important • *You can also run this script on an existing Application Catalog that already contains applications, but the **Extended Attributes** tab will not be visible for those existing applications; it will only be visible for applications imported after the script is run.*

2. Open Microsoft SQL Server Management Studio and connect to your database server.

3. Open the following file:

   `[AdminStudio_Installation_Directory]\Support\SQL_Scripts\ApplicationExtendedAttributes.SQL`

   You will see the following sample query (in comment form):

   ```
   /* Below is the sample query to create extended attribute*/
   --insert into ASCMExtendedAttribute
   --([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],[DefaultValue],[OptimisticLockField],[GCRecord]) values
   --('Name','DisplayText','HelpText','Text','Values','DefaultFileExtension','FileFilter','DefaultValue',0,NULL)

   --GO
   ```

4. For each extended attribute that you want to define in the `ApplicationExtendedAttributes.SQL` script, copy the following three lines of code (without the leading hyphens):

   ```
   insert into ASCMExtendedAttribute

   ([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],
       [DefaultValue],[OptimisticLockField],[GCRecord]) values

   ('Name','DisplayText','HelpText','Text','Values','DefaultFileExtension','FileFilter',
       'DefaultValue',0,NULL)
   ```

   Note the following regarding this code:

   - **Table name**—The first line of code (`insert into ASCMExtendedAttribute`) identifies the table in the Application Catalog that you are editing.

     Important • *Do not edit this line of code.*

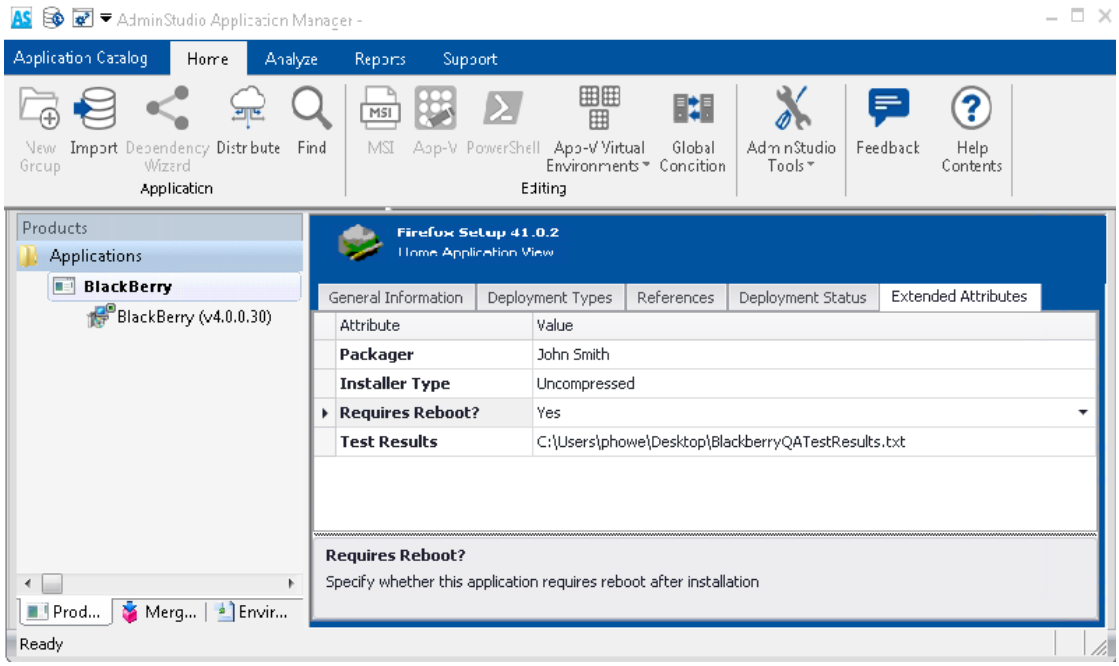
- **Column names**—The second line of code—which starts with `([Name],[DisplayText],` etc.—lists the columns in the table that need to be defined for each extended attribute.

  Important • *Do not edit this line of code.*

- **Defining the attribute**—The third line of code—which starts with `('Name','DisplayText',` etc.—defines the extended attribute. You will need to replace each of the default values in this line of code with the value appropriate for the extended attribute you are defining. Each value is separated by a comma.

  Important • *Do not delete any of the quote marks or commas from this line of code. If you do not want to define a value for a column, enter* ***`NULL`*** *or just leave a space between the quote marks.*

5. Define each attribute, as described in Defining Application Extended Attributes.
6. Under the list of attribute definitions, enter the command **GO**.
7. From the drop-down list in the toolbar, select the name of the Application Catalog you want to create these extended attributes in.
8. Click the **Execute** button in the toolbar.
9. In Application Manager, import an application and then open the **Application View**. A new **Extended Attributes** tab should now be displayed, listing the attributes you defined in the SQL file.



Defining Application Extended Attributes

For each extended attribute that you define, you have to edit the following line of code to replace the placeholder text with a value for each of the columns that were defined in the previous line of code:

```
('Name','DisplayText','HelpText','Text','Values','DefaultFileExtension','FileFilter',
    'DefaultValue',0,NULL)
```

Important • *Do not delete any of the quote marks or commas from this line of code. If you do not want to define a value for a column, enter* ***NULL*** *or just leave a space between the quote marks.*

The following table describes the possible values for each column:

Table 7-15 • ASCMExtendedAttribute Table Columns

Column	Possible Values
[Name]	Replace the default value of Name with a unique name to identify this Extended Attribute in the table. ***Note*** • *Do not use spaces or special characters. Make sure that it is a unique name.*
[DisplayText]	Replace the default value of DisplayText with the name of this extended attribute. This value will be displayed in the **Attribute** column on the **Extended Attributes** tab of the **Application View**.
[HelpText]	Replace the default value of HelpText with text to describe the purpose of this extended attribute or any other information the end user will need to know when specifying a value for this attribute on the Extended Attributes tab. This text will be displayed at the bottom of the **Extended Attributes** tab when this attribute is selected. Therefore, do not leave the default value of HelpText. If you do not want to display any help text, enter **NULL** or leave this value empty.
[Type]	Replace the default value of Type with one of the following values: • **Text**—Enter this value to define a text field. • **Selection**—Enter this value to define a drop-down list. When defining a drop-down list, you will also need to define the values of the drop down list using the [Values] column. • **File**—Enter this value to prompt the end user to browse for a file. When defining file selection field, you will also need to define the default file extension, using the [DefaultFileExtension] column, and the available file filters, using the [FileFilter] column.
[Values]	If [Type] is set to **Selection**, use this column to define the selections in the drop-down list. Separate the values using a semicolon (;), such as: **Illinois;Michigan;Wisconsin**

Table 7-15 • ASCMExtendedAttribute Table Columns

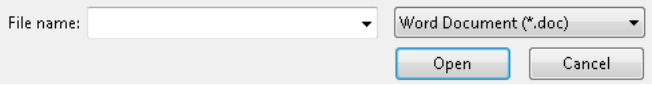
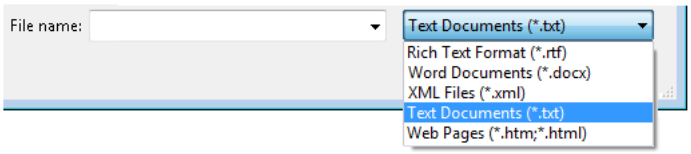
Column	Possible Values
[DefaultFileExtension]	If [Type] is set to **File**, replace the default value of DefaultFileExtension with the extension that you want to be selected, by default, in the file type drop down list.  For example, if you wanted *.doc to be the default file type, enter the following: **'*.doc'**
[FileFilter]	If [Type] is set to **File**, replace the default value of FileFilter with code to identify the valid file types for this field.  You need to enter both the name of the file type—such as **Word Documents (*.doc)**—as well as the actual file type, such as ***.doc**. You separate these two values by a pipe **|** character, such as: **Word Documents (*.doc)|*.doc** You can specify two file types for the same entry by separating them by a semi-colon, such as: Word Documents (*.doc)|*.doc;*.docx To specify multiple entries, separate them by a pipe | character, such as: **Word Documents (*.doc)|*.doc;*.docx|** **HTML Documents (*.html)|*.html;*.htm|** **Text Documents (*.txt)|*.txt** In other words, enter these values in the following sequence: **Name_A|Extension1;Extension2|Name_B|Extension|Name_C|Extension**

Table 7-15 • ASCMExtendedAttribute Table Columns

Column	Possible Values
[DefaultValue]	Enter one of the following, depending upon the value specified for [Type]: • **Text**—If [Type] is set to **Text**, replace the default value of DefaultValue with text that you want to pre-populate this field. • **Selection**—If [Type] is set to **Selection**, replace the default value of DefaultValue with one of the values defined in the [Values] column to identify a default value. For example, if the [Values] column is set to **Illinois;Michigan;Wisconsin**, and if you want Michigan to be selected as the default selection in the drop-down list, then set the value of the [DefaultValue] column to **Michigan**. • **File**—If [Type] is set to **File**, replace the default value of DefaultValue with any file path. Make sure that the file type in this path matches one of the file filters listed in the [FileFilter] column definition. If you do not want to enter a default value, set [DefaultValue] to **NULL** or just leave a space.
[OptimisticLockField]	Always set this value set to **0**.
[GCRecord]	Always set this to NULL; if you do not, this record will be treated as deleted.

Sample Script and Extended Attributes Tab

If your ApplicationExtendedAttributes.SQL script included the following code:

```
insert into ASCMExtendedAttribute
([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],
    [DefaultValue],[OptimisticLockField],[GCRecord]) values
('Packager','Packager','Enter the name of the person who repackaged this application',
    'Text',' ',' ',' ',' ',0,NULL)

insert into ASCMExtendedAttribute
([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],
    [DefaultValue],[OptimisticLockField],[GCRecord]) values
('InstallerType','Installer Type','Specify whether this is a compressed or uncompressed setup',
    'Selection','Compressed;Uncompressed',' ',' ','Uncompressed',0,NULL)

insert into ASCMExtendedAttribute
([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],
    [DefaultValue],[OptimisticLockField],[GCRecord]) values
('RequiresReboot','Requires Reboot?','Specify whether this application requires reboot after
    installation','Selection','Yes;No;Unknown',' ',' ',' ',0,NULL)

insert into ASCMExtendedAttribute
([Name],[DisplayText],[HelpText],[Type],[Values],[DefaultFileExtension],[FileFilter],
    [DefaultValue],[OptimisticLockField],[GCRecord]) values
('TestResults','Test Results','Upload file containing QA test results','File',' ','*.doc',
    'Word Documents (*.doc)|*.doc|HTML Documents (*.html)|*.html;*.htm|
    Text Documents (*.txt)|*.txt',' ',0,NULL)
```

GO

... after it was run on an Application Catalog, the **Extended Attributes** tab of the **Application View** would list the following extended attributes:

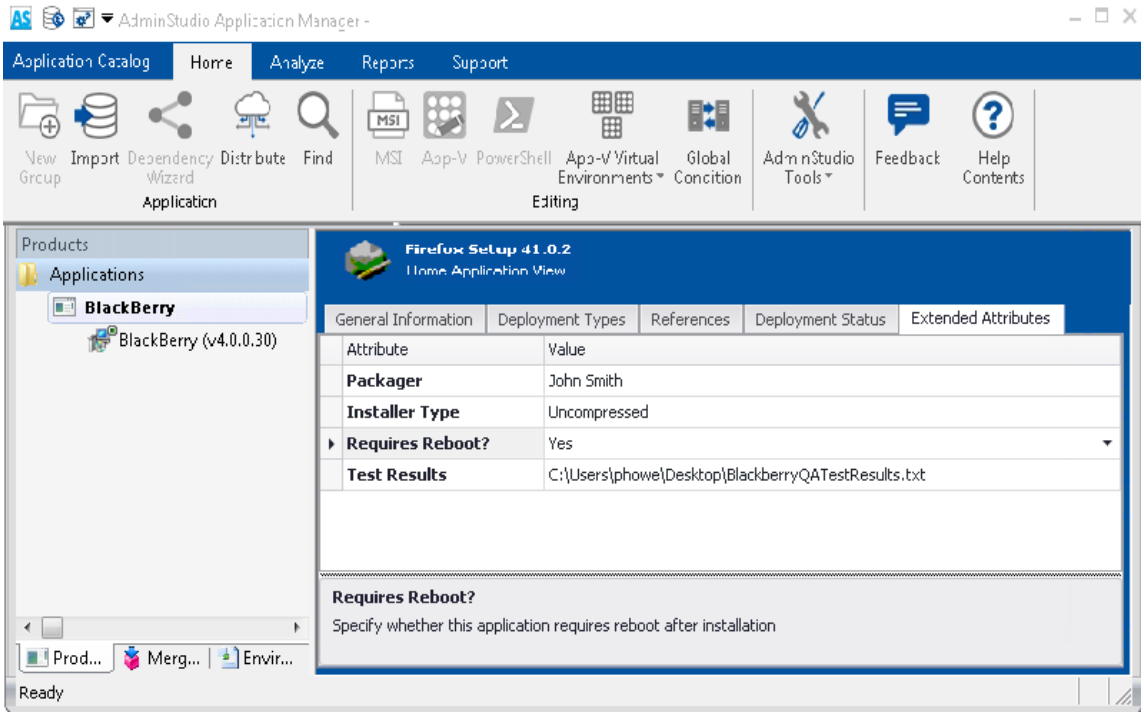


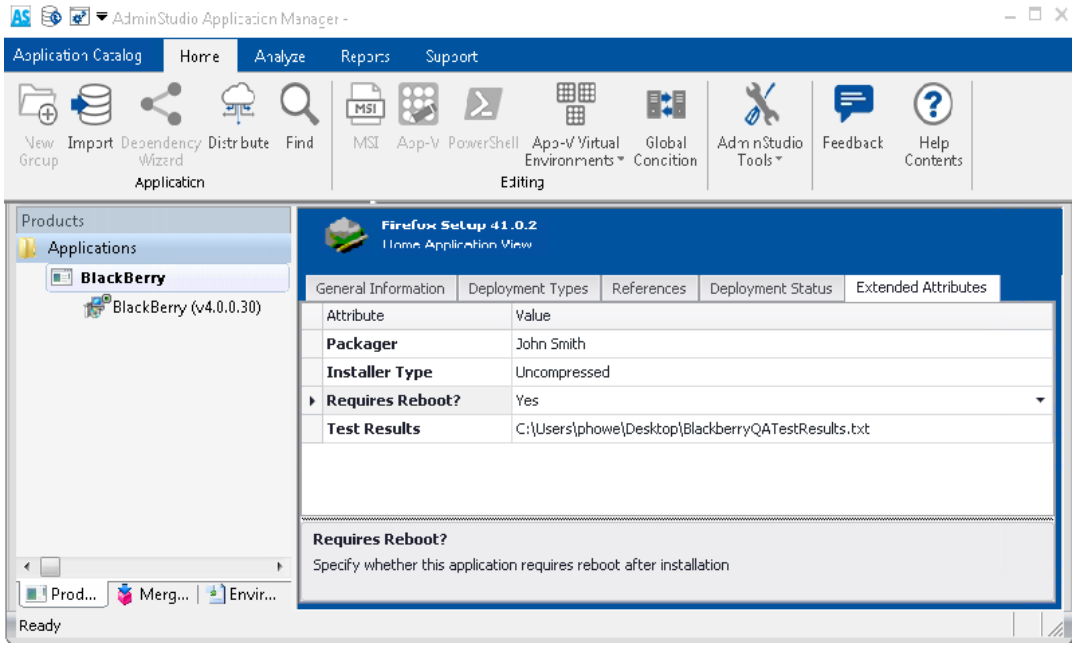
Figure 7-8: Extended Attributes Tab of the Application View

Viewing and Editing Application Extended Attributes

To view an application's extended attributes, perform the following steps:

Task ***To view an application's extended attributes:***

1. Make sure that the steps in Enabling the Extended Attributes Tab of the Application View have been performed.
2. Open Application Manager and select the **Home** tab of the ribbon.
3. Select an application node in the tree. The **Application View** opens.
4. Select the **Extended Attributes** tab. The custom extended attributes that have been defined are listed.



5. Edit the fields, if desired.

Managing ConfigMgr (Formerly called as System Center Configuration Manager) Package Deployment Data

AdminStudio displays deployment data for all of an application's packages (deployment types) in a multi-tabbed, organized format that is easy to navigate through and to update.

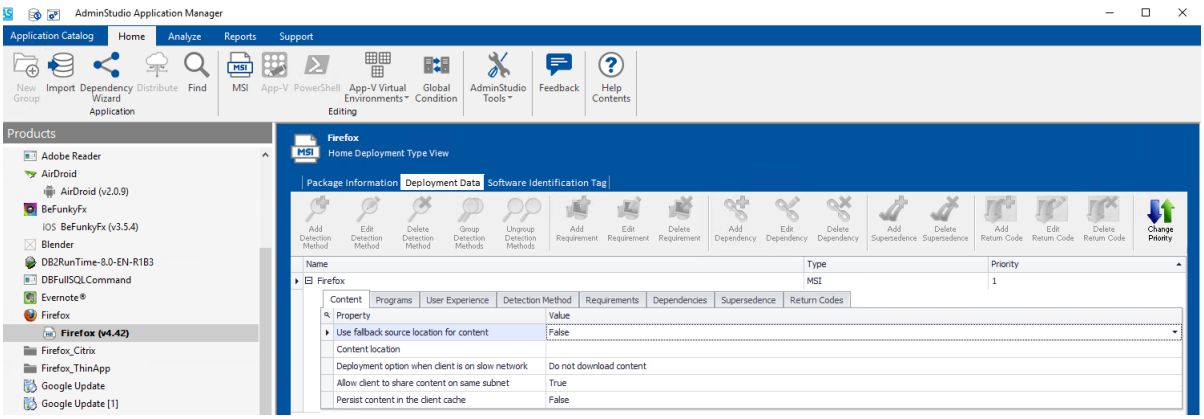


Figure 7-9: Home Deployment Type View / Deployment Data Tab

When a package is imported into the Application Catalog, Application Manager mines package elements for deployment data such as detection methods, dependencies, and requirements. You can view and modify this data and add new data by editing the properties on the subtabs of the **Deployment Data** tab and by using the easy-to-use wizards provided on the **Detection Methods**, **Requirements**, **Dependencies**, and **Supersedence** subtabs.

Using the subtabs of the **Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

- Specifying Package Content Deployment Data
- Specifying Package Programs Deployment Data
- Specifying Package User Experience Deployment Data
- Specifying Package Detection Methods Deployment Data
- Viewing Detection Methods for a Windows Store Application or MSIX Package
- Viewing a Windows Store Application's Framework Customizations
- Specifying Package Requirements Deployment Data
- Specifying Package Dependencies Deployment Data
- Specifying Package Supersedences Deployment Data
- Viewing and Editing Return Codes
- Changing the Priority of Deployment Types

Deployment Data and ConfigMgr (Formerly called as System Center Configuration Manager)

The data displayed on the **Deployment Data** tab of the **Home Deployment Type View** is used by ConfigMgr (Formerly called as System Center Configuration Manager) when deploying packages. This data corresponds to the application model data stored for applications and packages in Microsoft System Center 2012 Configuration Manager and later. When packages are published from the Application Catalog to ConfigMgr (Formerly called as System Center Configuration Manager), this data is also published, which helps to ensure successful deployment.

- **Importing external packages**—When a package is imported into the Application Catalog, Application Manager mines package elements for this Microsoft System Center 2012 Configuration Manager application model deployment data.
- **Importing packages from Microsoft System Center 2007 Configuration Manager**—When a package is imported from System Center 2007 Configuration Manager into the Application Catalog, Application Manager also imports detailed deployment data from Configuration Manager. This data will be useful to have when migrating this package to a System Center 2012 Configuration Manager application.

You can view and modify this deployment data and add new data by editing the properties on the subtabs of the **Deployment Data** tab and by using the wizards provided on the **Detection Methods**, **Requirements**, **Dependencies**, and **Supersedence** subtabs.

Setting Application Model Properties

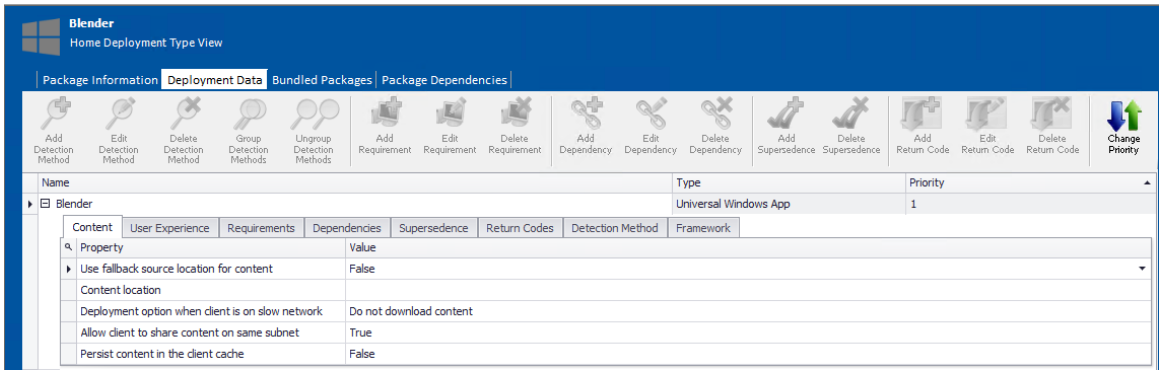
When a package is imported into the Application Catalog, AdminStudio inserts default values for various Microsoft System Center 2012 Configuration Manager application model properties which cannot be extracted from the imported package. There are several ways to set the default values for application model properties that are assigned during application import. You can also edit the application model properties for packages already in the Application Catalog. For more information, see Setting Application Model Properties.

Specifying Package Content Deployment Data

The **Content** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists general information about package contents.

Task ***To specify deployment information about package contents:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Contents** subtab.



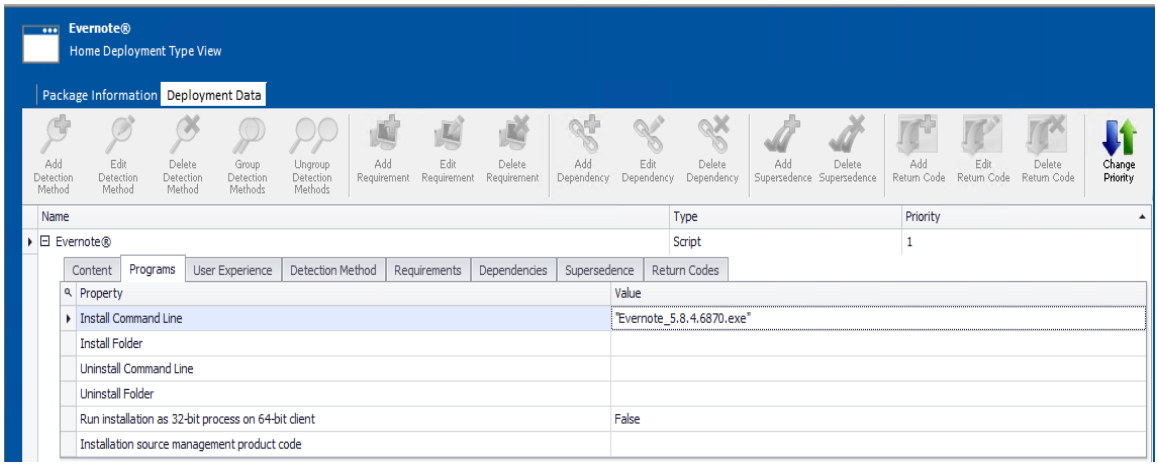
4. View and modify data, as described in Deployment Data Tab / Content Subtab.

Specifying Package Programs Deployment Data

The **Program** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists command line parameters for package installation and uninstallation.

Task ***To specify package program command line parameters:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Programs** subtab.



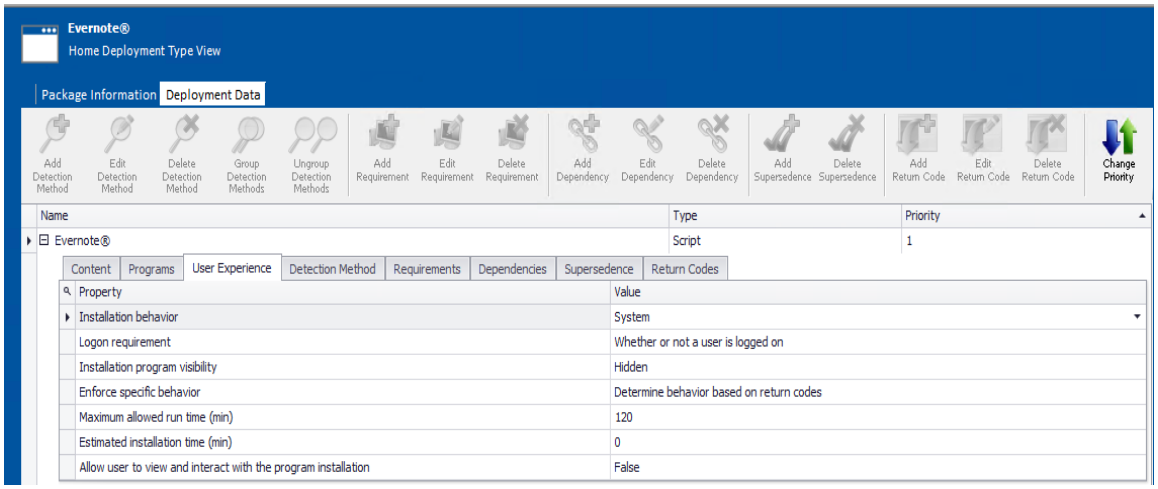
4. View and modify data, as described in Deployment Data Tab / Programs Subtab.

Specifying Package User Experience Deployment Data

The **User Experience** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists parameters relating to the user experience during installation.

Task ***To specify user experience parameters:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **User Experience** subtab.



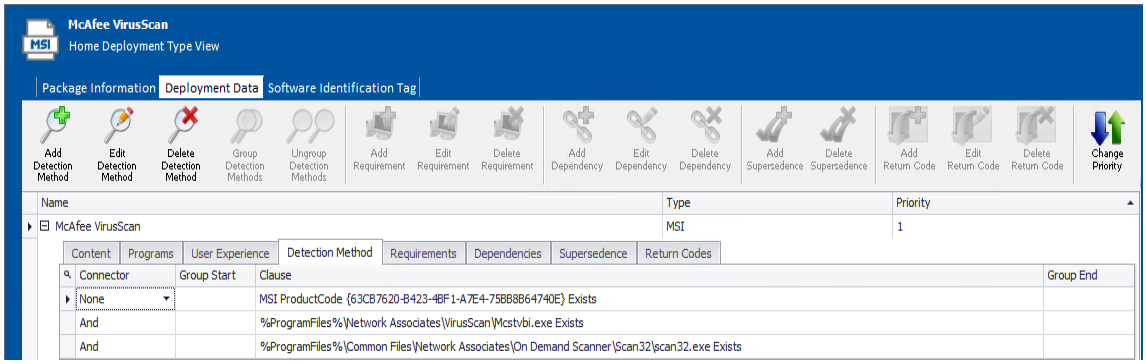
4. View and modify data, as described in Deployment Data Tab / User Experience Subtab.

Specifying Package Detection Methods Deployment Data

The **Detection Method** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists methods to detect whether this package is already installed on the target system.

Task ***To specify package detection methods:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Detection Method** subtab.



4. View existing detection methods, as described in Deployment Data Tab / Detection Method Subtab.
5. To add a detection method, click the **Add Detection Method** button in the ribbon toolbar to open the Detection Method Wizard.
6. To modify an existing detection method, select the detection method and click **Edit Detection Method**.

Note • *If you attempt to edit a Windows Installer Detection detection method, and you attempt to change the detection method property (from **Upgrade Code** to **Version** or vice versa), you may be required to required to browse to the Windows Installer file again to retrieve the new property value.*

7. To delete an existing detection method, click **Delete Detection Method**.

Viewing Detection Methods for a Windows Store Application or MSIX Package

Note • *The **Detection Method AppX** subtab is displayed for Windows Store mobile apps and MSIX packages.*

When you have a Windows Store mobile app or MSIX pacakage deployment type selected in the Application Manager tree, the **Detection Method AppX** subtab of the **Deployment Data** tab of the **Home Deployment Type View** is displayed, which lists methods to detect whether this Windows Store mobile app or MSIX package is already installed on the target system.

Task ***To view a Windows Store mobile app's detection methods:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Store mobile app in the tree and select the Windows Store deployment type. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Detection Method AppX** subtab.
4. View existing detection methods, as described in Deployment Data Tab / Detection Method AppX, MSIX Subtab.

Viewing a Windows Store Application's Framework Customizations

Edition • *Support for mobile apps is included when you purchase AdminStudio Professional or Enterprise Edition with Mobile.*

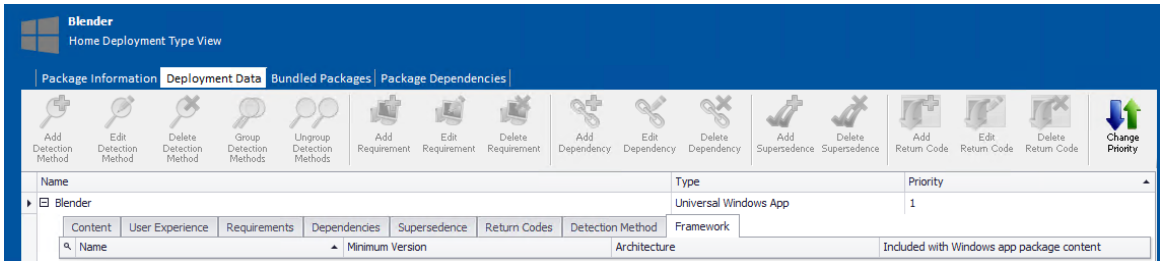
When you have a Windows Store mobile app deployment type selected in the Application Manager tree, the **Framework** subtab of the **Deployment Data** tab of the **Home Deployment Type View** is displayed, and displays any customizations that may have been added to this Windows Store mobile app.

Windows Store mobile app developers can use the application framework to customize a mobile app. With the framework, they can create a task or an extension to customize the application. They can extend existing functions within the application or embed new functionality with custom business logic.

If the selected Windows Store mobile app has any application framework customizations, they will be listed on the **Framework** subtab.

Task ***To view a Windows Store mobile app's application framework customizations:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Store mobile app in the tree and select the Windows Store deployment type. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Framework** subtab.



4. View existing application framework customizations, as described in Deployment Data Tab / Framework Subtab.

Specifying Package Requirements Deployment Data

You can use the **Requirements** subtab of the **Deployment Data** tab to add user, device, or custom requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package.

- Creating Custom Requirements Containing Global Conditions
- Creating Device Requirements
- Creating User Requirements

Creating Custom Requirements Containing Global Conditions

You can use the **Requirements** subtab of the **Deployment Data** tab to add requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package. You can set device requirements, custom device requirements, and user and group requirements.

When you open the **Requirement Wizard** and choose to create a custom requirement, you can create a requirement that contains global conditions by selecting **Expression** from the **Condition Type** list on the **Create Global Condition** dialog box. When you select **Expression** from this list, an expression builder interface is displayed. You have the option of using global conditions, containing expressions, to create complex custom requirements.

- Building Expressions When Creating Global Conditions
- Creating and Editing Global Conditions

Building Expressions When Creating Global Conditions

You can use the **Requirements Wizard** to create global conditions that use expressions (enabling you to connect clauses using AND and OR operators). Because many requirements may be common among applications, you could use this expression builder to add multiple custom requirements together as clauses in a global condition, and use this global condition in a custom requirement that you can assign to multiple deployment types, instead of creating a separate custom requirement for each.

When you open the Requirement Wizard and choose to create a custom requirement, there is an option in the **Condition Type** list on the **Create Global Condition** dialog box: **Expression**. When you select **Expression** from this list, an expression builder interface is displayed.

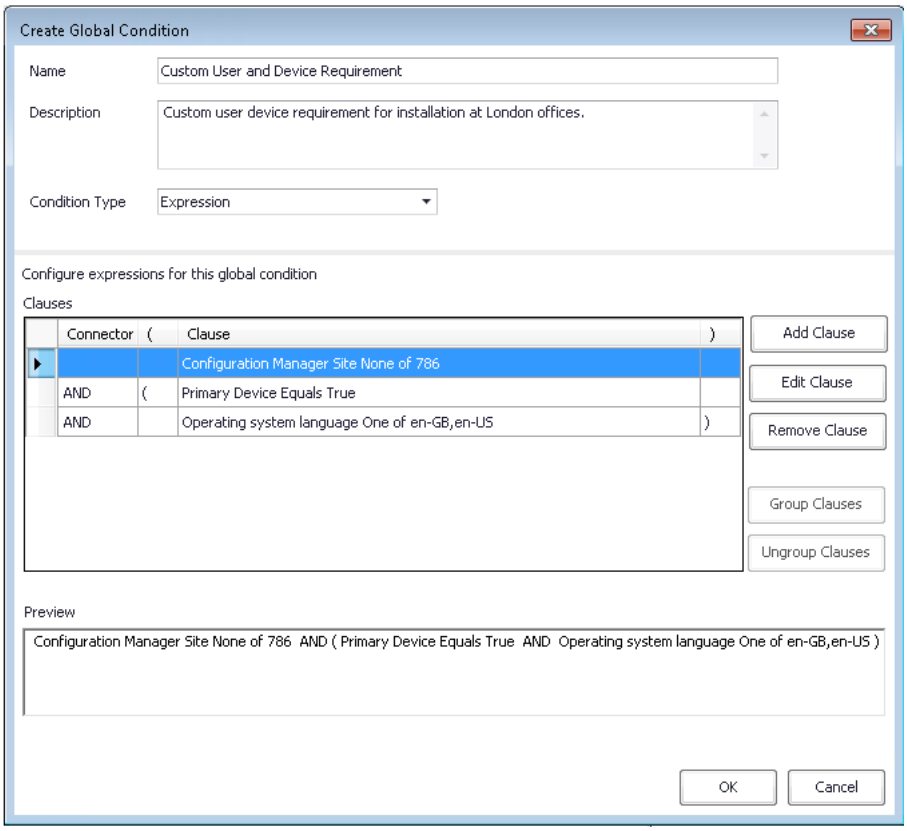


Figure 7-10: Building an Expression on the Create Global Condition Dialog Box

You can use the expression builder interface to form an expression using existing User/Device/Custom requirements. After you add multiple requirements, you can then connect them using **AND** or **OR** operators, and can group sets of clauses, which enables you to create complex requirements.

The expression building area of this dialog box includes the following options:

Table 7-16 • Create Global Condition Dialog Box

Option	Description
Add Clause	Click to open the Requirement Wizard, which you can use to add a User/Device/Custom requirement. When you click Finish on the wizard, the new requirement will be listed in the **Clauses** list. When you add the first requirement the **Connector** will be set to **None**. When adding subsequent requirements, the **Connector** will be set to **AND** by default.
Edit Clause	Click to edit the selected requirement using the Requirement Wizard.
Remove Clause	Click to delete the selected requirement.
Group Clauses	Click to group the selected requirements (if the grouping criteria matches). If grouping is successful, then the selected requirements will be marked as grouped and parentheses will be displayed the **(** and **)** columns.

Table 7-16 • Create Global Condition Dialog Box

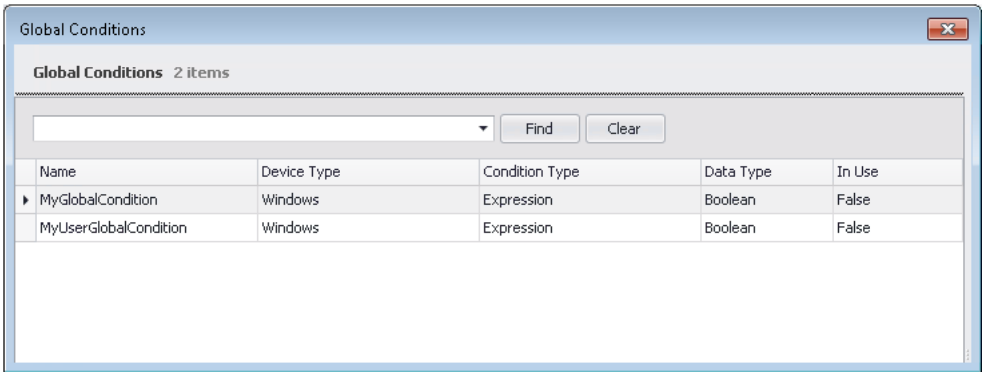
Option	Description
Ungroup Clauses	Click to ungroup the selected requirements, if the ungroup criteria matches.
Preview	Lists the full expression.

Creating and Editing Global Conditions

In addition to using the **Requirements Wizard** to create global conditions, you can create new global conditions and edit existing global conditions on the **Global Conditions** dialog box, which can be opened by clicking the **Global Conditions** button on the **Home** tab of the Application Manager ribbon.

Task ***To edit global conditions:***

1. On the **Home** tab of the Application Manager ribbon, click the **Global Conditions** button. The **Global Conditions** dialog box opens, listing all of the global conditions present in the current Application Catalog.



   On the **Global Conditions** dialog box, the name, device type, condition type, and data type of each condition is listed, as well as whether the condition is in use or not.

2. On the **Global Conditions** dialog box, you can edit or delete an existing global condition or create a new global condition:

   - **Editing an existing global condition**—Right-click on the condition and then select **Edit Condition** from the shortcut menu. The **Create Global Condition** dialog box opens, where you can edit the condition.
   - **Deleting an existing global condition**—Right-click on the condition and then select **Delete Condition** from the shortcut menu.
   - **Adding a new global condition**—Right-click anywhere on the list of conditions and select **Create New Condition** from the shortcut menu. The **Create Global Condition** dialog box opens, where you can define a new condition.
   - **View references**—If a condition is in use, right-click on the condition and select **References** from the shortcut menu to open the **References** dialog box, which lists the referring applications and the referring global conditions of the selected global condition.

3. If you have a large list of global conditions and would like to perform a search, you can use the search box and **Find** button to filter the list.

Creating Device Requirements

You can use the **Requirements** subtab of the **Deployment Data** tab to add device requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy a package. You can choose to add a custom device requirement or a device requirement from Configuration Manager.

- Creating a Custom Device Requirement
- Creating a Device Requirement from ConfigMgr (Formerly called as System Center Configuration Manager)

Creating a Custom Device Requirement

To create a custom device requirement, perform the following steps:

Task ***To create a custom device requirement:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Requirements** subtab.
4. Click the **Add Requirement** button in the ribbon toolbar to open the Requirement Wizard.
5. On the **Welcome** panel, select **Device requirements** and click **Next**. The **Select the Device Requirements Type** panel opens.
6. Select **Custom Device requirement** and click **Next**. The **Create Device Requirements** panel opens.
7. Use the following fields to build a custom device requirement:

Property	Description
Condition	Select one of the following conditions: • **Active Directory Site** • **Configuration Manager Site** • **CPU Speed (MHz)** • **Disk space** • **Number of processors** • **Operating system** • **Operating system language** • **Organizational unit (OU)** • **Total physical memory (MB)** • **Windows Store inactive**
Rule Type	Select a rule type from the list. For custom device requirements, **Value** is the only type listed.

Property	Description
Operator	Select an operator from the list. Possible sets of operators are: • **One of** or **None of** • **Equals**, **Not equal to**, **Greater than**, **Less than**, **Between**, **Greater than or Equal to**, or **Less than or equal to**
[Additional Fields]	Additional fields are displayed depending upon the **Condition** selected. Use these fields to define the requirement for the selected **Condition**. • **Active Directory Site**—Click the **Add** button and add a site to the **Active Directory Sites** list. • **Configuration Manager Site**—Click the **Add** button and add a site to the **Configuration Manager Sites** list. • **CPU Speed (MHz)**—Enter a value, in MHz, in the **Value (MHz)** text field. • **Disk space**—Select a drive from the **Select logical drive** list and enter a value, in MBs, in the **Value (MB)** text box. • **Number of processors**—Enter a number in the **Value** text box. • **Operating system**—Select operating systems from the **Select Operating System** list. You can choose just a major category (such as **Windows 8** or **Windows Server 2012**) or you can identify a specific operating system / service pack / processor type combination, such as **All Windows 8 (32-bit)**. • **Operating system language**—Select languages from the **Select Operating System Language(s)** list. • **Organizational unit (OU)**—Click the **Add** button and add a OU to the list. • **Total physical memory (MB)**—Enter a value, in MBs in the **Value (MB)** text box. • **Windows Store inactive**—Enter a value in the **Value** text box.

8. Click **Next**. The **Summary** panel opens, listing the components of your custom device requirement.
9. Click **Finish** to close the wizard.

Creating a Device Requirement from ConfigMgr (Formerly called as System Center Configuration Manager)

To create a device requirement from ConfigMgr (Formerly called as System Center Configuration Manager), perform the following steps:

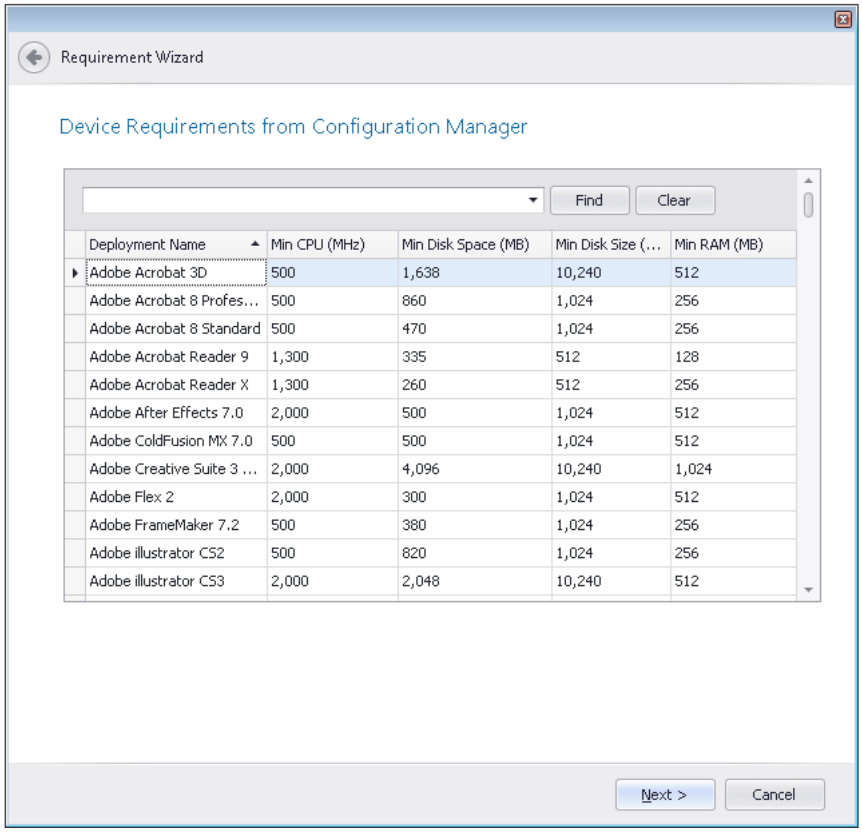
Task ***To create a device requirement from ConfigMgr (Formerly called as System Center Configuration Manager):***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Requirements** subtab.
4. Click the **Add Requirement** button in the ribbon toolbar to open the Requirement Wizard.

5. On the **Welcome** panel, select **Device requirements** and click **Next**. The **Select the Device Requirements Type** panel opens.

6. Select **Device requirement from Configuration Manager** and click **Next**. The **Configuration Manager Credentials** panel opens. The information from the System Center 2012 Configuration Manager named connection that you have set up on the **Distribution System** tab of the **Options** dialog box pre-populates the **Server**, **Site Code**, and **Username** fields.

7. Enter the required **Password** and click **Next**. The **Device Requirements from Configuration Manager** panel opens and lists those applications in the System Center 2012 Configuration Manager server that have defined device requirements.



8. Select the application in the list that matches the one that you are editing, and click **Next** to continue. The **Summary** panel opens, and lists the device requirement that you are adding.

Note • *For more information, see Device Requirements from Configuration Manager Panel.*

9. Click **Finish** to add the device requirement.

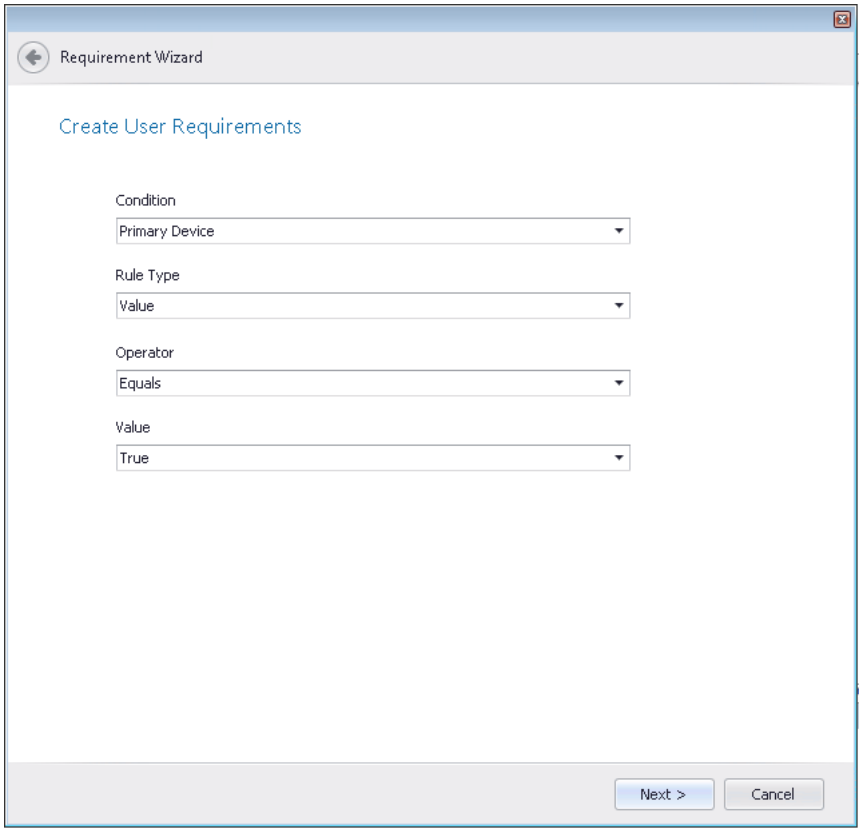
Creating User Requirements

You can use the **Requirements** subtab of the **Deployment Data** tab to add user requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy a package.

To create a user requirement, perform the following steps:

Task ***To create a user requirement:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Requirements** subtab.
4. To add a requirement, click the **Add Requirement** button in the ribbon toolbar to open the Requirement Wizard.
5. On the **Welcome** panel, select **User requirements** and click **Next**. The **Create User Requirements** panel opens.



6. Use the following fields to build a user requirement:

Property	Description
Condition	Select a condition type from the list. For user requirements, **Primary Device** is the only condition type listed.
Rule Type	Select a rule type from the list. For custom device requirements, **Value** is the only type listed.
Operator	Select a rule type from the list. For user requirements, **Equals** is the only operator listed.
Value	Select either **True** or **False** to define this user requirement.

7. Click **Next**. The **Summary** panel opens, listing the components of your user requirement.
8. Click **Finish** to close the wizard.

Specifying Package Dependencies Deployment Data

You can use the **Dependencies** subtab to view or edit a list of other packages in the Application Catalog that must also be deployed with this package onto the target machine in order for this package to successfully operate.

You can use the Dependency Wizard to add new dependencies or to scan for dependencies.

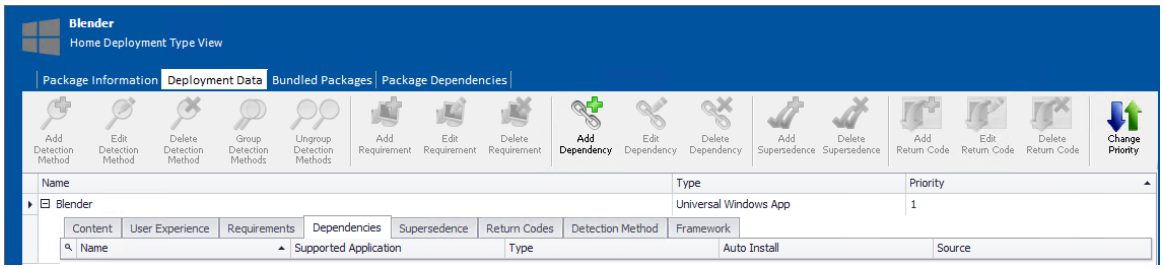
- Viewing and Editing Package Dependencies
- Adding a Dependency Using the Dependency Wizard
- Scanning for Dependencies

Viewing and Editing Package Dependencies

To view and edit package dependencies, perform the following steps:

Task ***To specify package dependencies:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Dependencies** subtab.



4. View existing dependencies, as described in Deployment Data Tab / Dependencies Subtab.
5. To modify an existing dependency, select the dependency and click **Edit Dependency**.
6. To delete an existing dependency, click **Delete Dependency**.

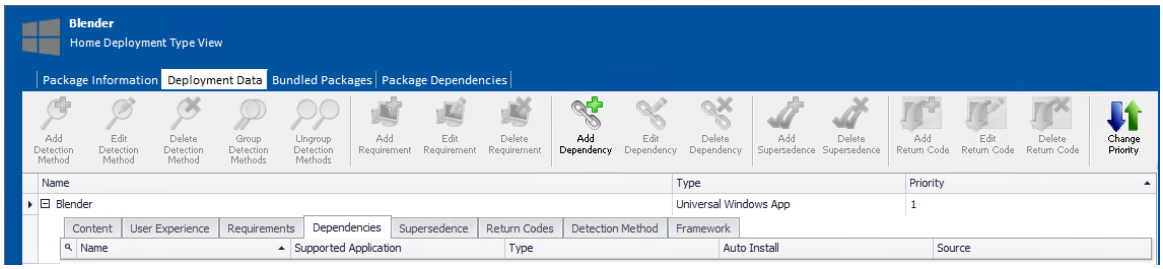
Adding a Dependency Using the Dependency Wizard

To add a dependency to a package using the **Dependency Wizard**, perform the following steps:

Task ***To add a package dependencies:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.

3. Click the **Deployment Data** tab and open the **Dependencies** subtab.



4. To add a dependency, click the **Add Dependency** button in the ribbon toolbar. The **Welcome** panel of the Dependency Wizard opens.
5. Select one of the following options:
   - **Select dependencies from Application Catalog**
   - **Select dependencies from Configuration Manager**
6. Click **Next**. The **Deployment Types in Application Catalog** or **Deployment Types in Configuration Manager** panel opens.
7. From the **Specify or select a Group for dependencies** list, either select an existing group from the list or enter the name for a new group.
8. From the list of deployment types, select those that are dependent on the selected package.
9. Click **Next**. The **Summary** panel opens.
10. Click **Finish**. The wizard exits and the dependencies you selected are now listed on the **Dependencies** tab.

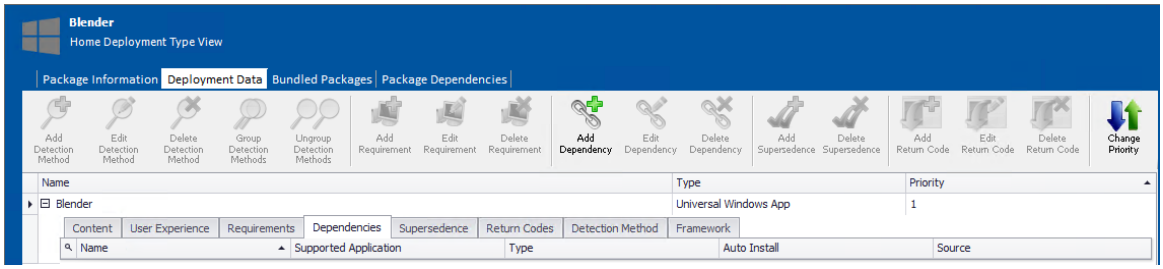
Scanning for Dependencies

To add dependencies to a package by performing a dependency scan using the **Dependency Wizard**, perform the following steps:

Note • *When you scan a Windows Installer package for dependencies, using the* ***Auto detect dependencies*** *option of the* ***Dependency Wizard****, you also populate the package's file level Dependencies View.*

Task ***To scan for dependencies:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Dependencies** subtab.



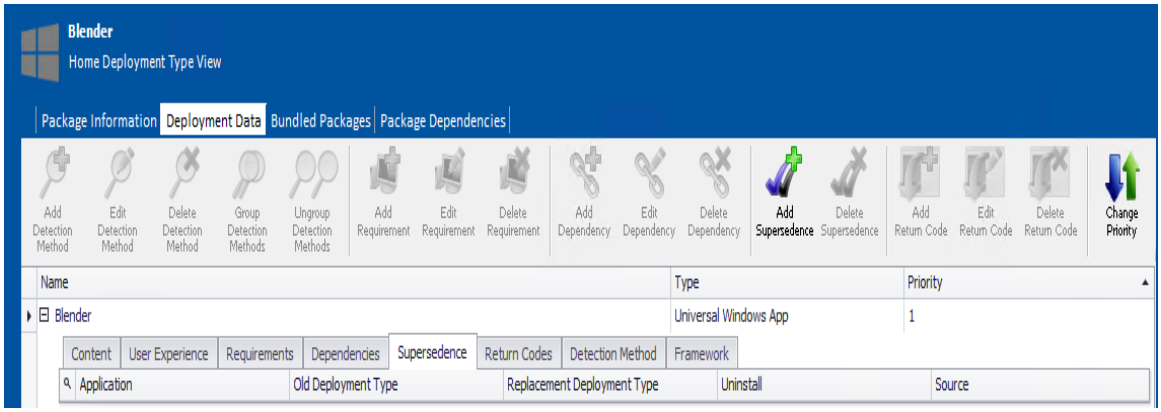
4. To scan for dependencies, click the **Add Dependency** button in the ribbon toolbar. The **Welcome** panel of the Dependency Wizard opens.
5. Select the **Auto detect dependencies** option and click **Next**. The **Auto Detect Dependencies** panel opens.
6. Click **Next** to begin scanning. The **Scanning Progress** panel opens showing the progress of the scan.
7. When scanning is complete, click **Next**. The **Auto Scan Results** panel opens.
   - **If dependencies were found**—The dependencies are listed. Select the dependencies that you want to add to the **Dependency** tab and specify a group name in the **Specify or select a Group for dependencies** list.
   - **If no dependencies were found**—Packages in the Application Catalog are listed. Select the packages that you want to specify as dependencies and specify a group name in the **Specify or select a Group for dependencies** list.
8. Click **Next**. The **System Requirements** panel opens and lists any system requirements that were detected for the selected package.
9. Click **Next**. The **Summary** panel opens, listing the selected dependencies.
10. Click **Finish** to close the wizard and add the dependencies to the list.

Specifying Package Supersedences Deployment Data

You can use the **Supersedence** subtab to view or edit a list of other packages that this package would supersede if installed on the same target machine (meaning that the package on the target system would need to be uninstalled prior to installing this package).

Task ***To specify package supersedences:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Supersedence** subtab.



4. View existing supersedences, as described in Deployment Data Tab / Supersedence Subtab.
5. To add a supersedence, click the **Add Supersedence** button in the ribbon toolbar to open the Supersedence Wizard.
6. To modify an existing supersedence, select the supersedence and click **Edit Supersedence**. To delete an existing dependency, click **Delete Supersedence**.

Viewing and Editing Return Codes

You can edit a MSI and EXE package's return codes in the Application Manager interface. Return codes are used to indicate whether a restart is required, whether an installation is complete, and to customize the text shown to users when a specific code is returned.

You can view a package's return codes on the **Return Codes** subtab of the **Deployment Types** tab on the **Application View** or **Home Deployment Type View**.

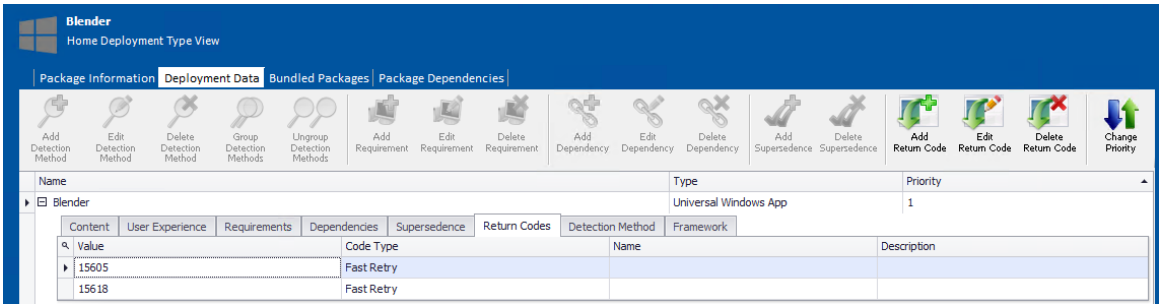


Figure 7-11: Return Codes Subtab of Deployment Types tab

The following return codes are populated by default during package import:

- **0**—Success (no reboot)
- **1707**—Success (no reboot)
- **3010**—Soft Reboot
- **1641**—Hard Reboot
- **1618**—Fast Retry

On the **Return Codes** tab, you can add, edit, and delete return codes.

- **Adding a return code**—Click **Add Return Code** in the ribbon and define a return code on the **Add Return Code** dialog box.

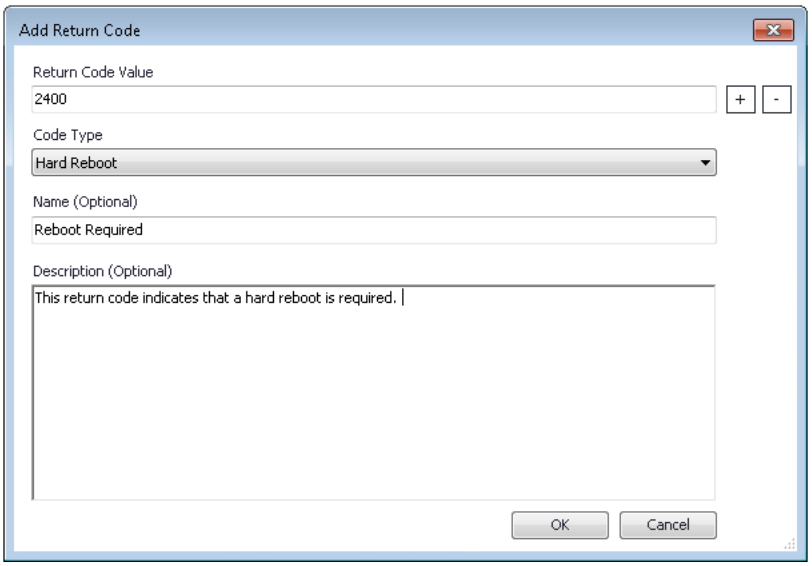


Figure 7-12: Add Return Code Dialog Box

When you create a new return code:

- The **Return Code Value** should be unique.
- The **Name** and **Description** fields are optional.

- **Editing a return code**—Select a return code, click **Edit Return Code** in the ribbon, and edit the details of the return code on the **Edit Return Code** dialog box. However, the **Return Code Value** field cannot be edited.
- **Deleting a return code**—Select a return code, click **Delete Return Code** in the ribbon, and confirm the deletion.

Changing the Priority of Deployment Types

When an application has multiple deployment types, the order in which they will be evaluated in System Center 2012 Configuration Manager depends upon the deployment type's assigned priority. When a deployment type meets the specified requirements, it will be run and then no further deployment types on the priority list will be evaluated. By default, Application Manager assigns a deployment type a priority based upon their import order.

You can modify the priority setting of an application's deployment types on the **Change Deployment Type Priority** dialog box, which is opened by clicking the **Change Priority** button in the **Deployment Types** tab ribbon.

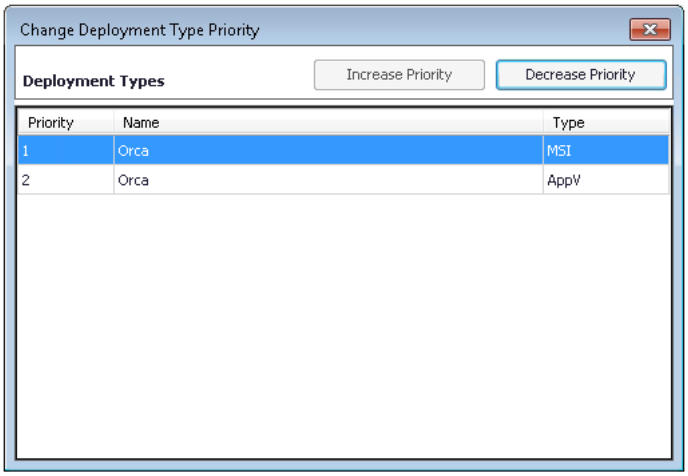


Figure 7-13: Change Deployment Type Priority Dialog Box

Just select the deployment type and click **Increase Priority** or **Decrease Priority** to move it up and down in the list.

Note • *You can only assign a priority to Windows Installer, App-V, and .exe packages. All other packages are assigned a priority of -1, which cannot be changed.*

Setting Application Model Properties

Application Manager stores Microsoft System Center 2012 Configuration Manager application model deployment data for each application in the Application Catalog. Much of this data is displayed on the **Deployment Data** tab of the **Home Deployment Type View**.

When a package is imported into the Application Catalog, AdminStudio inserts default values for various Microsoft System Center 2012 Configuration Manager application model properties which cannot be extracted from the imported package.

Setting Default Application Model Properties

There are several ways to set the default values for the application model properties that are assigned during application import:

- **Options dialog box**—You can set these defaults by editing the values on the **Import / Application Model Defaults** tab of the Application Manager **Options** dialog box. See Setting Default Application Model Properties on the Options Dialog Box.
- **SQL script**—You can also set these defaults by editing and running an SQL script and run this script against your Application Catalog. The defaults will be applied to all new packages imported into the Application Catalog. You can also choose to automatically run this script each time a new Application Catalog is created. See Setting Default Application Model Properties Using an SQL Script.

Setting the Application Model Properties of Existing Packages

You can also edit the application model properties for packages already in the Application Catalog in the Application Manager user interface, as described in Managing ConfigMgr (Formerly called as System Center Configuration Manager) Package Deployment Data, or by using the AdminStudio PowerShell Cmdlets, as described in Setting Application Model Properties Using the AdminStudio PowerShell Cmdlets.

Setting Default Application Model Properties

When a package is imported into the Application Catalog, AdminStudio inserts default values for various Microsoft System Center 2012 Configuration Manager application model properties which cannot be extracted from the imported package.

You can set the defaults for these properties using the Application Manager user interface or by running an SQL script:

- Setting Default Application Model Properties on the Options Dialog Box
- Setting Default Application Model Properties Using an SQL Script

Setting Default Application Model Properties on the Options Dialog Box

You can specify the default values for the Microsoft System Center 2012 Configuration Manager application model properties that are assigned to a package when it is imported into the Application Catalog on the **Import Options / Application Model Defaults** tab of the Application Manager **Options** dialog box.

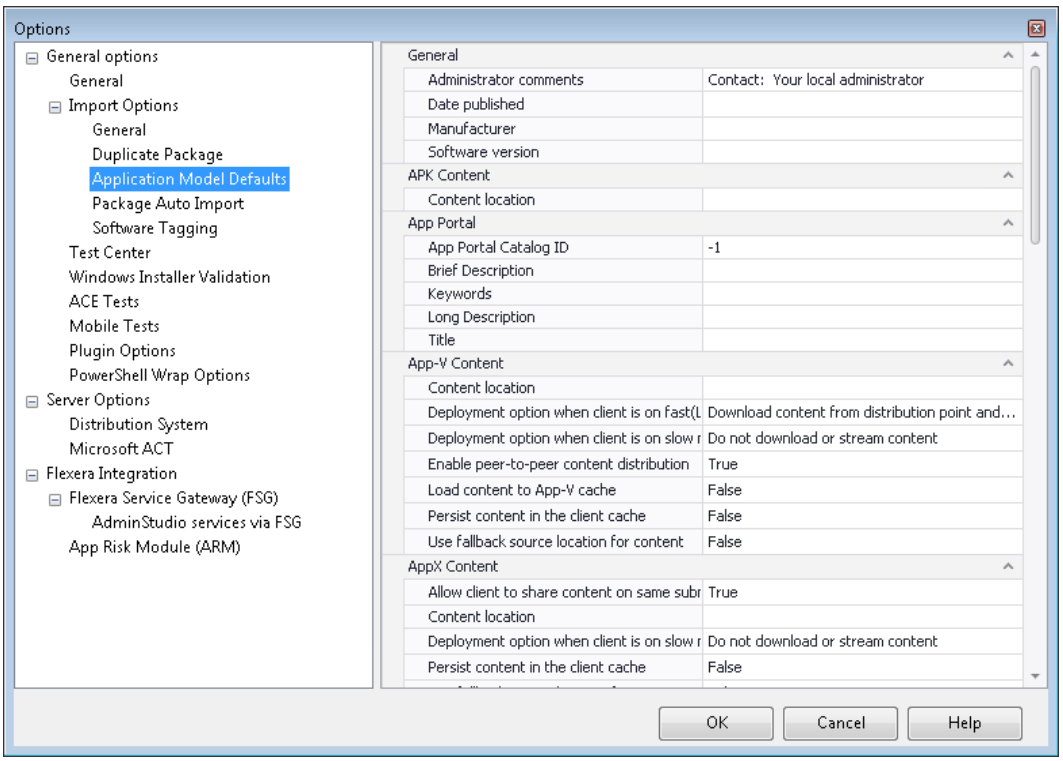


Figure 7-14: Options Dialog Box / Application Model Defaults Tab

For a detailed list of all of the properties that you can edit on the **Application Model Defaults** tab, see Import Options / Application Model Defaults Tab.

Setting Default Application Model Properties Using an SQL Script

When a package is imported into the Application Catalog, AdminStudio inserts default values for various Microsoft System Center 2012 Configuration Manager application model properties which cannot be extracted from the imported package.

You can use an SQL script to set these default values. To do this, you just need to update an existing SQL script file (`CustomDefaultValues.SQL`) and then run that SQL file to update your Application Catalog. The defaults will then be applied to all new packages imported into the Application Catalog.

You can also choose to run the script each time you create a new Application Catalog.

- Editing the CustomDefaultValues.SQL Script File
- Adding the CustomDefaultValues.SQL Script to the Scripts Run During Application Catalog Creation

Editing the CustomDefaultValues.SQL Script File

To specify default values for application model properties, you need to modify the `CustomDefaultValues.SQL` script file, which can be found in the following location:

```
C:\Program Files (x86)\AdminStudio\2022 R2\Support\SQL_Scripts
```

Tip • *If you run this script, all of the properties in the script file will be updated. If you want to update just a few of the properties, you can create a script file that only contains a subset of the properties and then run that script.*

The following application model properties are defined in the `CustomDefaultValues.SQL` file:

Table 7-17 • Application Model Properties in the CustomDefaultValues.SQL File

Property	Values	SQl Statement to Set Default Value
AutoInstall	True False	UPDATE [ASCMProperty] SET [DefaultValue]='True' WHERE Name='AutoInstall'
Classification	Desktop Server	UPDATE [ASCMProperty] SET [DefaultValue]='Desktop' WHERE Name='Classification'
DisplaySupersedes	True False	UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='DisplaySupersedes'
DistributionPriority	High Medium Low	UPDATE [ASCMProperty] SET [DefaultValue]='Medium' WHERE Name='DistributionPriority'
EnforceBehaviour	BasedOnExitCode NoAction ProgramReboot ForceReboot	UPDATE [ASCMProperty] SET [DefaultValue]='BasedOnExitCode' WHERE Name='EnforceBehaviour' AND [Class] = 'ASCMMsiUserExperience'
ExecuteTime	Any integer value	UPDATE [ASCMProperty] SET [DefaultValue]='120' WHERE Name='ExecuteTime' AND [Class] = 'ASCMMsiUserExperience'
FallbackToUnprotectedDP	True False	UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='FallbackToUnprotectedDP' AND [Class] = 'ASCMMSIContent'

Table 7-17 • Application Model Properties in the CustomDefaultValues.SQL File (cont.)

Property	Values	SQl Statement to Set Default Value
InstallBehaviour	User System Any	`UPDATE [ASCMProperty] SET [DefaultValue]='System' WHERE Name='InstallBehaviour' AND [Class] = 'ASCMMsiUserExperience'`
InstallFolder	Any string value	`UPDATE [ASCMProperty] SET [DefaultValue]='' WHERE Name='InstallFolder' AND [Class] = 'ASCMMsiInstaller'`
LogonRequirement	True Null False	`UPDATE [ASCMProperty] SET [DefaultValue]='Null' WHERE Name='LogonRequirement' AND [Class] = 'ASCMMsiUserExperience'`
MaxExecuteTime	Any integer value	`UPDATE [ASCMProperty] SET [DefaultValue]='60' WHERE Name='MaxExecuteTime' AND [Class] = 'ASCMMsiUserExperience'`
OnFastNetwork	Download DownloadContentForStreaming	`UPDATE [ASCMProperty] SET [DefaultValue]='Download' WHERE Name='OnFastNetwork' AND [Class] = 'ASCMMSIContent'`
OnSlowNetwork	DoNothing Download DownloadContentForStreaming	`UPDATE [ASCMProperty] SET [DefaultValue]='DoNothing' WHERE Name='OnSlowNetwork' AND [Class] = 'ASCMMSIContent'`
PeerCache	True False	`UPDATE [ASCMProperty] SET [DefaultValue]='True' WHERE Name='PeerCache' AND [Class] = 'ASCMMSIContent'` ***Note*** • *To set the* ***Allow client to share content on the same subnet*** *property (for MSI or Script packages), pass* PeerCache *as a property name along with MSI/Script package ID.* *To set the* ***Enable peer-to-peer content distribution*** *property (for an App-V package), pass* PeerCache *as a property name along with the App-V package ID.*
PinOnClient	True False	`UPDATE [ASCMProperty] SET [DefaultValue]='True' WHERE Name='PinOnClient' AND [Class] = 'ASCMMSIContent'`
PreferredDistribute	True False	`UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='PreferredDistribute'`
PrestagedDPSetting	Auto OnlyContentChange ManualCopy	`UPDATE [ASCMProperty] SET [DefaultValue]='ManualCopy' WHERE Name='PrestagedDPSetting'`
ProgramVisibility	Maximized Normal Minimized Hidden	`UPDATE [ASCMProperty] SET [DefaultValue]='Hidden' WHERE Name='ProgramVisibility' AND [Class] = 'ASCMMsiUserExperience'`

Table 7-17 • Application Model Properties in the CustomDefaultValues.SQL File (cont.)

Property	Values	SQl Statement to Set Default Value
RequireLoad	True False	UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='RequireLoad' AND [Class] = 'ASCMMSIContent'
RequiresUserInteraction	True False	UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='RequiresUserInteraction'
RunAs32	True False	UPDATE [ASCMProperty] SET [DefaultValue]='False' WHERE Name='RunAs32' AND [Class] = 'ASCMMsiInstaller'
SourceUpdateProductCode	Any valid GUID	UPDATE [ASCMProperty] SET [DefaultValue]='' WHERE Name='SourceUpdateProductCode' AND [Class] = 'ASCMMsiInstaller'
UninstallCommandLine	Any string value	UPDATE [ASCMProperty] SET [DefaultValue]='' WHERE Name='InstallCommandLine' AND [Class] = 'ASCMMsiInstaller'
UninstallCommandLine	Any string value	UPDATE [ASCMProperty] SET [DefaultValue]='' WHERE Name='UninstallCommandLine' AND [Class] = 'ASCMMsiInstaller'
UninstallFolder	Any string value	UPDATE [ASCMProperty] SET [DefaultValue]='' WHERE Name='InstallFolder' AND [Class] = 'ASCMMsiInstaller'

Adding the CustomDefaultValues.SQL Script to the Scripts Run During Application Catalog Creation

When you create a new Application Catalog, the scripts listed in the <Create> section under <AdminStudio> in the Upgrade.xml file are run. The Upgrade.xml file is found in the following location:

```
C:\Program Files (x86)\AdminStudio\2022 R2\Support
```

If you want to use an SQL script to set the default Application Model properties for all applications that are imported into a new Application Catalog, you need to add the name of the CustomDefaultValues.SQL file to this script.

Task ***To add the CustomDefaultValues.SQL script to the Upgrade.xml file:***

1. Edit the CustomDefaultValues.SQL script file, as described in Editing the CustomDefaultValues.SQL Script File, and copy it to the following location:

   ```
   C:\Program Files (x86)\AdminStudio\2022 R2\Support\SQL_Scripts
   ```

2. Open the following file in a text editor:

   ```
   C:\Program Files (x86)\AdminStudio\2022 R2\Support\Upgrade.xml
   ```

3. Locate the <AdminStudio> element in the Upgrade.xml file.

4. Add the CustomDefaultValues.SQL file to the list of scripts in the <SQLServer> child element of the <Create> element, as highlighted below:

```
<AdminStudio>
  <Version version="12.01" target="13286599"/>
  <Create>
    <SQLServer>
      <Script internal="yes">AS_System_Schema.SQL</Script>
      <Script>Seed_Data.SQL</Script>
      <!-- <Script internal="no">AS_Schema.sql</Script> -->
      <!-- MUST RUN CustomReportWizard.SQL LAST-->
      <Script>CustomReportWizard.SQL</Script>
      <Script>AS_TestCenter.sql</Script>
      <Script>AS_TestEngine.sql</Script>
      <Script>Reporting.StoredProcedures.sql</Script>
      <Script>CustomDefaultValues.SQL</Script>
    </SQLServer>
    ...
  </Create>
```

5. Save the `Upgrade.xml` file. The next time you create a new Application Catalog, the `CustomDefaultValues.SQL` script will be run, setting the default application model properties that you have defined.

Setting Application Model Properties Using the AdminStudio PowerShell Cmdlets

You can use the `Set-ASProperty` command of the AdminStudio PowerShell Cmdlets to set application model properties for applications which have already been imported into the Application Catalog.

For detailed information and instructions, see the Set-ASProperty section of the AdminStudio PowerShell Cmdlets.

Managing App-V Package Deployment Data

Note • *Because Microsoft App-V server only supports App-V 5.0 packages, the* ***App-V Deployment Data*** *subtab is only displayed for App-V 5.0 packages.*

When an App-V 5.0 package is imported into the Application Catalog, Application Manager mines package elements for App-V-specific deployment data. You can view and modify data for App-V 5.0 packages and add new data by editing the properties on the subtabs of the **App-V Deployment Data** tab. AdminStudio displays App-V deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

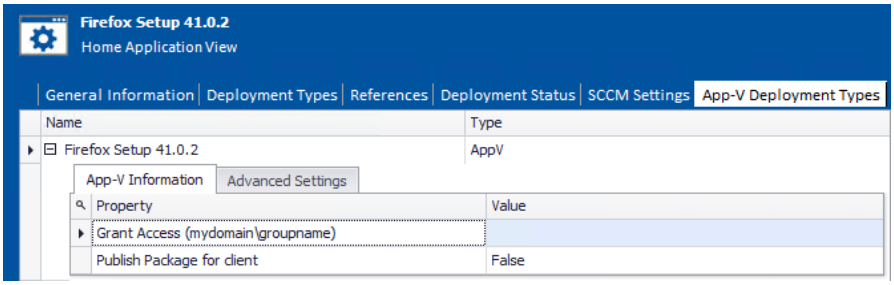


Figure 7-15: App-V Deployment Data

Using the subtabs of the **App-V Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

- Specifying a Package's App-V Deployment Settings
- Specifying a Package's Advanced App-V Deployment Settings

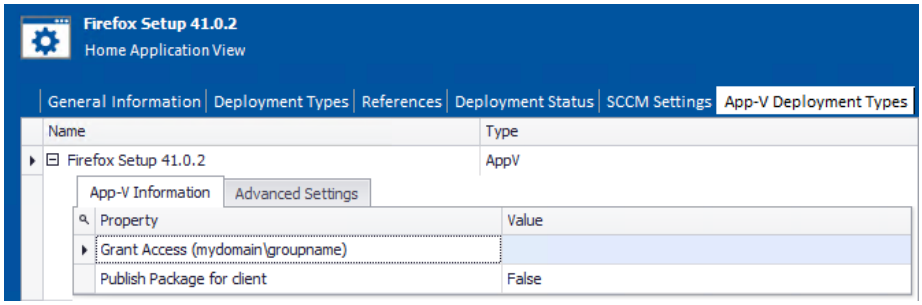
Specifying a Package's App-V Deployment Settings

Note • *Because Microsoft App-V server only supports App-V 5.0 packages, the **App-V Deployment Data** subtab is only displayed for App-V 5.0 packages.*

The **App-V Information** subtab of the **App-V Deployment Data** tab of the **Home Deployment Type View** lists parameters relating to package deployment on a Microsoft App-V Server.

Task ***To specify a package's App-V deployment settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an App-V 5.x package in the tree. The **Home Deployment Type View** opens.
3. Click the **App-V Deployment Data** tab and open the **App-V Information** subtab.



4. View and modify any desired data, as described in App-V Deployment Data Tab.

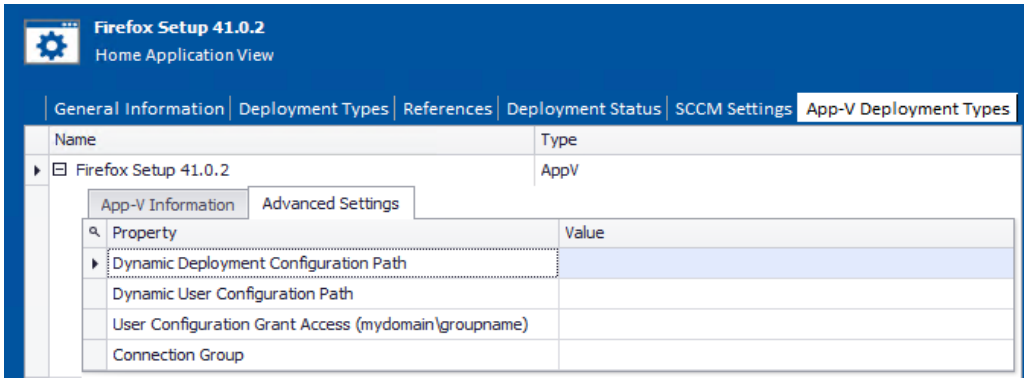
Specifying a Package's Advanced App-V Deployment Settings

Note • *Because Microsoft App-V server only supports App-V 5.0 packages, the **App-V Deployment Data** subtab is only displayed for App-V 5.0 packages.*

The **Advanced Settings** subtab of the **App-V Deployment Data** tab of the **Home Deployment Type View** lists advanced parameters relating to package deployment on a Microsoft App-V Server.

Task ***To specify a package's advanced App-V Server deployment settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an App-V 5.0 package in the tree. The **Home Deployment Type View** opens.
3. Click the **App-V Deployment Data** tab and open the **Advanced Settings** subtab.



4. View and modify data, as described in App-V Deployment Data Tab.

Managing Casper Package Deployment Data

Note • *Because Casper only supports macOS desktop packages, the* ***Casper Deployment Data*** *subtab is only displayed for* `.pkg` *files,* `.dmg` *files, and links to Apple Mac App Store apps.*

When a macOS desktop package is imported into the Application Catalog, Application Manager mines package elements for Casper-specific deployment data. You can view and modify deployment data for macOS desktop packages and add new data by editing the properties on the subtabs of the **Casper Deployment Data** tab. AdminStudio displays Casper deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

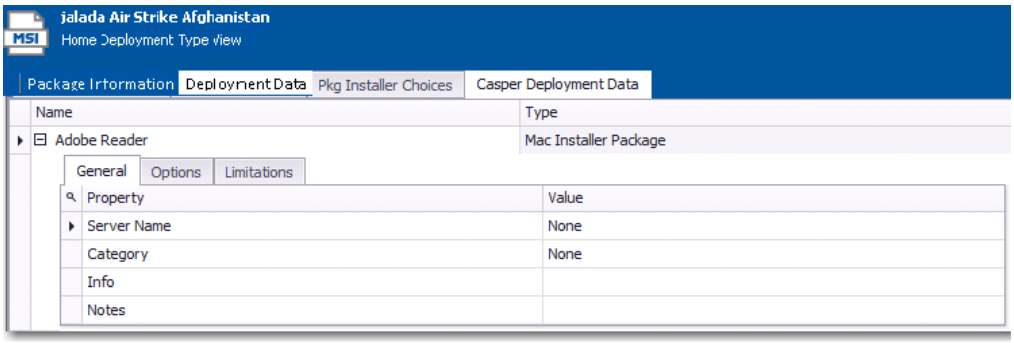


Figure 7-16: Casper Deployment Data

The **Casper Deployment Data** subtab of the **Home Deployment Type View** can include up to three subtabs that display Casper deployment data: **General**, **Options**, and **Limitations**. The **Options** and **Limitations** subtabs are not displayed for Mac App Store apps.

- General Tab
- Options Tab
- Limitations Tab

General Tab

The **General** tab of the **Casper Deployment Data** tab is displayed for all macOS desktop applications.

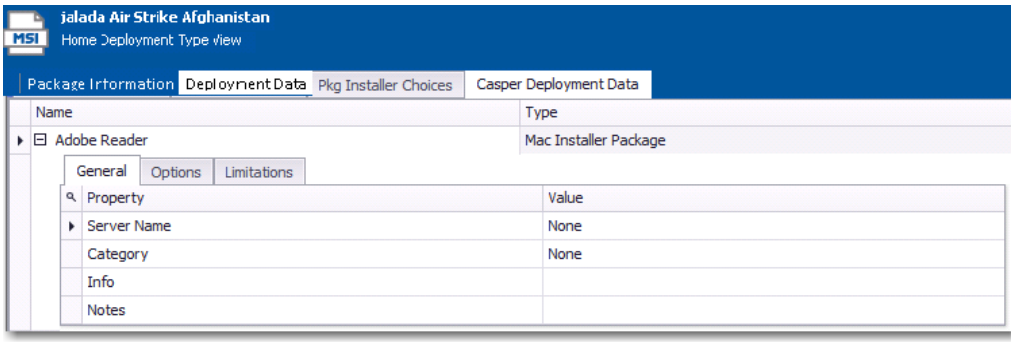


Figure 7-17: Casper Deployment Data / General Tab

The **General** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-18 • Casper Deployment Data Tab / General Subtab

Property	Description
Server Name	Name of the Casper server.
Category	Category in Casper that the package will be added to. ***Note*** • *Casper lets you create custom categories. If AdminStudio has matched this application to an entry in the Application Recognition Library (ARL), AdminStudio will use the ARL category when publishing to Casper, creating it if necessary.*
Info	Information to display to the administrator when the package is deployed or uninstalled
Notes	Notes to display about the package (such as the name of the person who built it and when it was built). ***Note*** • *Not displayed for Mac App Store Apps.*
Free	Indicates whether or not the Mac App Store app is available for free (**True**) or whether it requires payment (**False**). ***Note*** • *Only displayed for Mac App Store Apps.*

Options Tab

The **Options** tab of the **Casper Deployment Data** tab is only displayed for PKG and DMG packages.

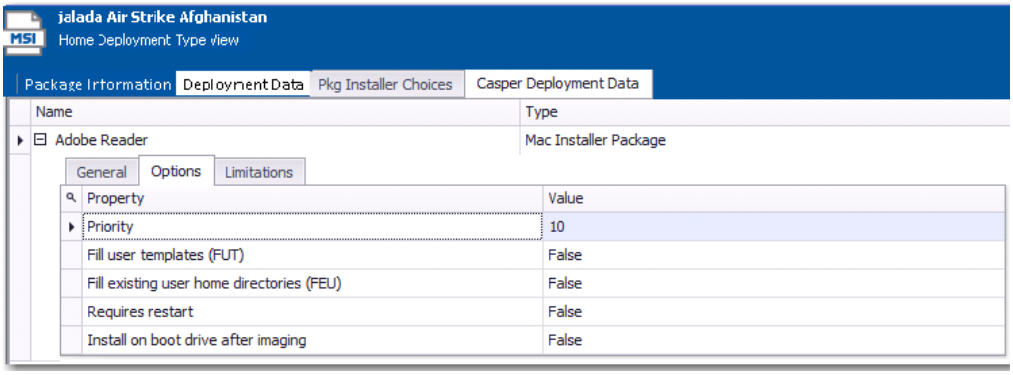


Figure 7-18: Casper Deployment Data / Options Tab

The **Options** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-19 • Casper Deployment Data Tab / Options Subtab

Property	Description
Priority	Priority to use for deploying or uninstalling the package. For example, a package with a priority of **1** is deployed or uninstalled before other packages. When several applications are deployed together, the one with the highest priority is installed first. Therefore, if one application requires that another application be installed first before it can be successfully installed, you should assign the required application a higher priority (lower number) than the dependent application.
Fill user templates (FUT)	Set this property to **True** to fill new home directories with the contents of the home directory in the package's Users folder. This setting can be changed when deploying or uninstalling the package using a policy.l ***Note*** • *Only applicable to DMG packages.*
Fill existing user home directories (FEU)	Set this property to **True** to fill existing home directories with the contents of the home directory in the package's **Users** folder. This setting can be changed when deploying or uninstalling the package using a policy ***Note*** • *Only applicable to DMG packages.*
Requires restart	Set this property to **True** to require that computers must be restarted after installing the package.

Table 7-19 • Casper Deployment Data Tab / Options Subtab

Property	Description
Install on boot drive after imaging	Set this property to **True** to ensure that the package is installed on the boot drive after imaging. ***Note*** *• This setting is only used when deploying a package with an OS image, like with an OSD. It does not affect day-to-day package delivery.*

Limitations Tab

The **Limitations** tab of the **Casper Deployment Data** tab is only displayed for PKG and DMG packages.

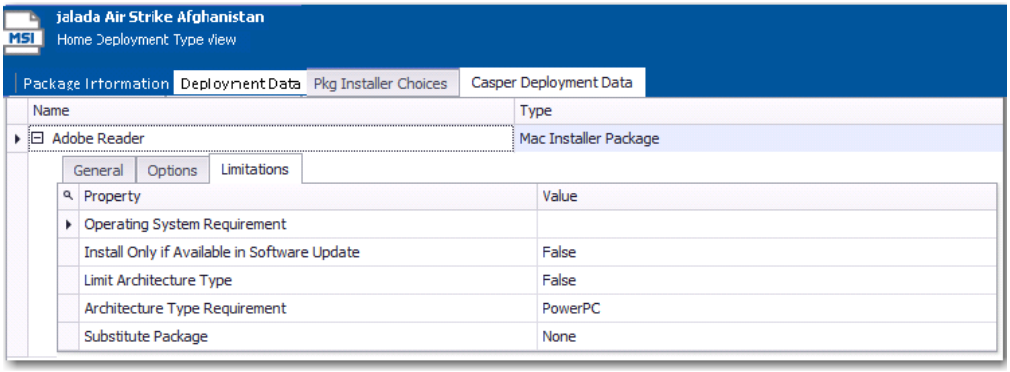


Figure 7-19: Casper Deployment Data / Limitations Tab

The **Limitations** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-20 • Casper Deployment Data Tab / Limitations Subtab

Property	Description
Operating System Requirement	Enter operating system version numbers, separated by commas, to specify that the package only be permitted to be deployed to computers with these operating system versions. To restrict installation to OS X 10.6.8, 10.7.x, or 10.8, you would enter the following: `10.6.8, 10.7.x, 10.8`
Install Only if Available in Software Update	Set to **True** to require that this package only be installed if it is available in a software update.
Limit Architecture Type	Set to **True** to require that this package only be installed on machines matching the selected **Architecture Type Requirement**.

Table 7-20 • Casper Deployment Data Tab / Limitations Subtab

Property	Description
Architecture Type Requirement	If **Limit Architecture Type** is set to **True**, select the one of the following to specify the architecture type required to deploy the package: • PowerPC • Intel/X86
Substitute Package	If you want to specify a different package to deploy to computers that do not meet the architecture type requirement, click the Browse button in this field to open the Select Substitute Package Dialog Box, and select a substitute package from either Casper or the Application Catalog.

Select Substitute Package Dialog Box

The **Substitute Package** field on the **Casper Deployment Data > Limitations** tab specifies the package to deploy to computers that do not have the required architecture type.

If you click on the **Substitute Package** field (which, by default, is set to **None**), the **Select Substitute Package** dialog box opens, prompting you to select a substitute package from either the Casper Server or the Application Catalog.

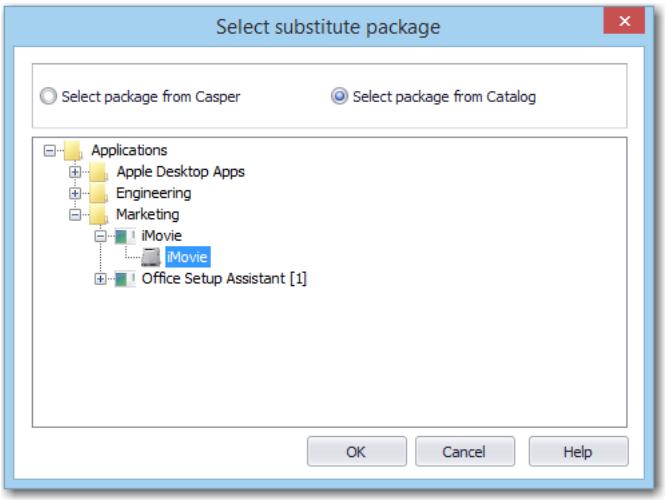


Figure 7-20: Select Substitute Package Dialog Box

Managing Citrix XenApp Package Deployment Data

Note • *Because Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles, the* ***XenApp Deployment Data*** *subtab is only displayed for App-V 4.x packages and Citrix XenApp profiles.*

When a XenApp profile or App-V 4.x package is imported into the Application Catalog, Application Manager mines package elements for Citrix XenApp-specific deployment data. You can view and modify data for Citrix XenApp profiles and App-V 4.x packages and add new data by editing the properties on the subtabs of the **XenApp Deployment Data** tab. AdminStudio displays XenApp deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

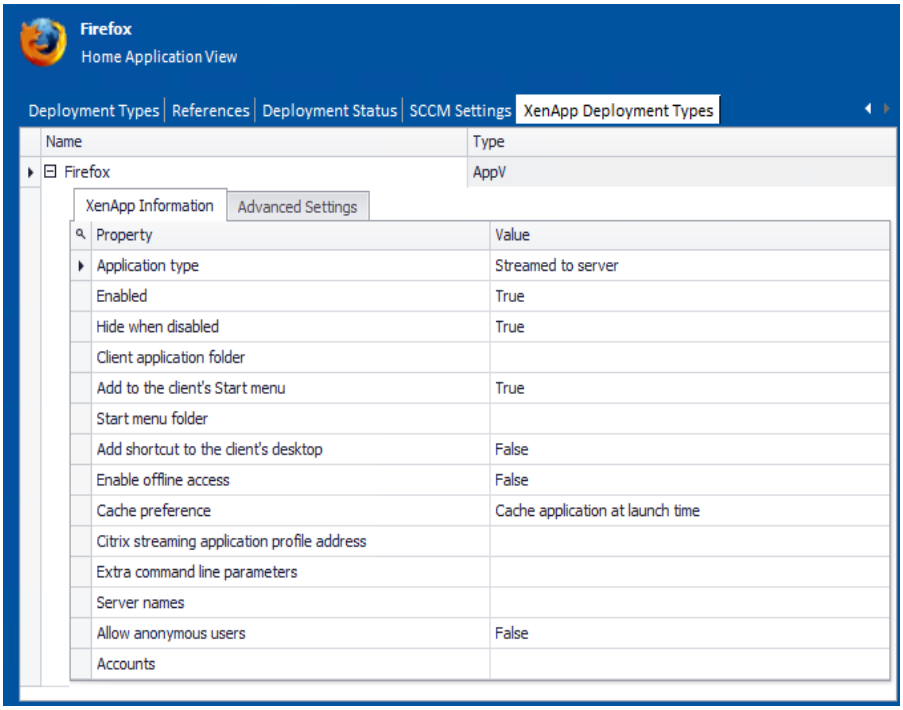


Figure 7-21: XenApp Deployment Data

Using the subtabs of the **XenApp Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

- Specifying a Package's XenApp Deployment Settings
- Specifying a Package's Advanced XenApp Deployment Settings

Important • *In order to publish an application to a Citrix XenApp Server, there are several mandatory properties which must be set on the* ***XenApp Information*** *subtab of the* ***XenApp Deployment Data*** *tab. If these properties are not set, distribution will fail.*

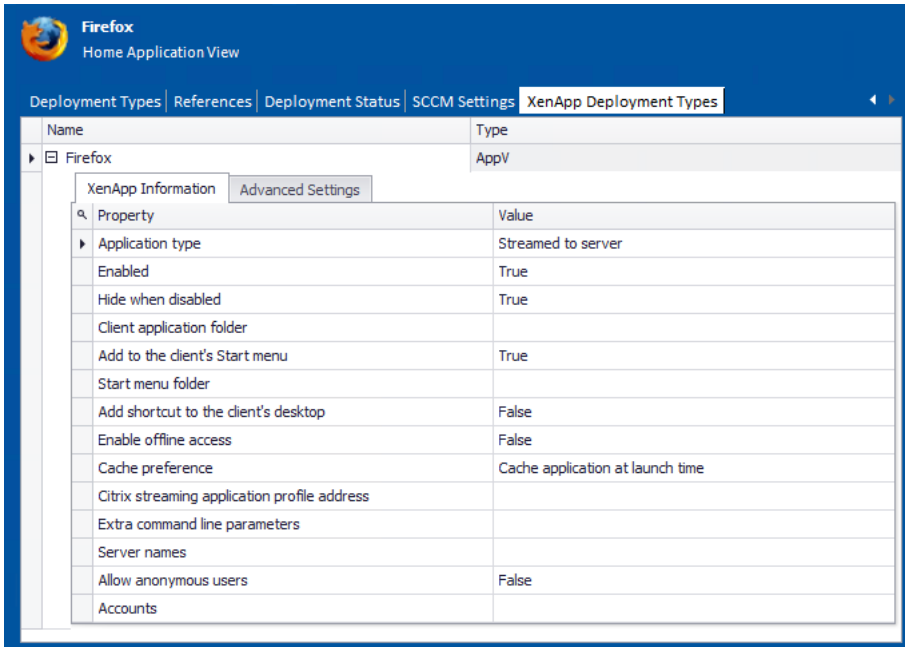
Specifying a Package's XenApp Deployment Settings

Important • *Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles. If you select another type of package in the tree, the* ***XenApp Deployment Data*** *tab of the* **Home** ***Deployment Type View*** *is not displayed.*

The **XenApp Information** subtab of the **XenApp Deployment Data** tab of the **Home Deployment Type View** lists parameters relating to package deployment on a Citrix XenApp Server.

Task ***To specify a package's XenApp deployment settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a Citrix XenApp or App-V 4.x package in the tree. The **Home Deployment Type View** opens.
3. Click the **XenApp Deployment Data** tab and open the **XenApp Information** subtab.



4. In the **Server names** field, you need to enter the Citrix XenApp server names where this application will be available. Click the browse button to open the **Servers** dialog box, where you can enter multiple server names or import a list of servers from an application server list file (`*.asl`).

Important • *This is a required field. If you only want to make this application available on the Citrix XenApp server that you are publishing to, enter that server name on the* ***Servers*** *dialog box. You can copy it from the* ***Distribution System*** *tab of the Application Manager* ***Options*** *dialog box.*

5. Set the **Allow anonymous users** field to one of the following values:
   - **False**—Do not grant access to anonymous users. (Default)
   - **True**—Grant access to anonymous users.
6. If the **Allow anonymous users** field is set to **False**, enter the accounts that you want to have access to this XenApp profile in the **Accounts** field. To do this, click the browse button to open the **Users** dialog box, where you can enter multiple user accounts or import a list of users from an application user list file (`*.aul`).

Note • *If* ***Allow anonymous users*** *is set to* ***True****, this field is not required. If* ***Allow anonymous users*** *is set to* ***False****, this is a mandatory field.*

7. If you are publishing an App-V package, enter the **Citrix streaming application profile address**, including the location of the manifest file (`.profile`). For example, enter the UNC path, such as:

   ```
   \\MyCitrixServer\Shared\App-V_IntegrationKit\
   AppStreamingToAppVConduit\AppStreamingToAppVConduit.profile
   ```

   Note • *If you are publishing an App-V package, this is a mandatory field.*

8. View and modify any other desired data, as described in XenApp Deployment Data Tab / XenApp Information Subtab.

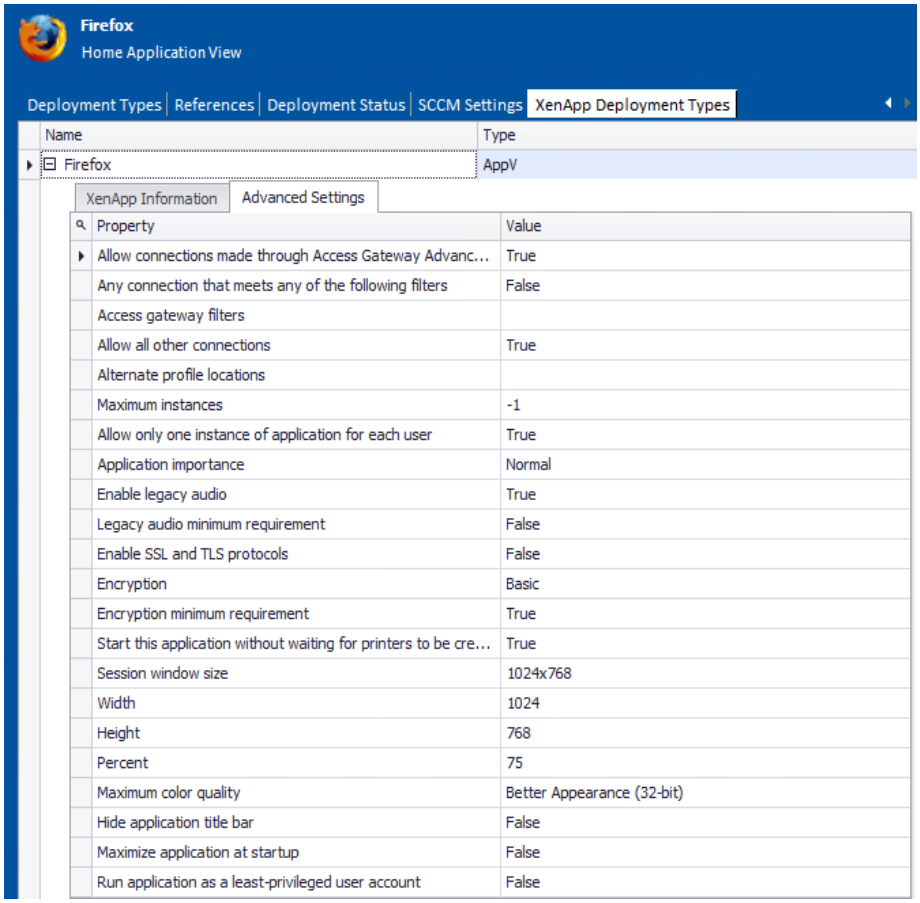
Specifying a Package's Advanced XenApp Deployment Settings

Important • *Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles. If you select another type of package in the tree, the **XenApp Deployment Data** tab of the **Home** **Deployment Type View** is not displayed.*

The **XenApp Information** subtab of the **XenApp Deployment Data** tab of the **Home Deployment Type View** lists parameters relating to package deployment on a Citrix XenApp Server.

Task ***To specify a package's advanced XenApp deployment settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a Citrix XenApp or App-V 4.x package in the tree. The **Home Deployment Type View** opens.
3. Click the **XenApp Deployment Data** tab and open the **Advanced Settings** subtab.



4. View and modify data, as described in XenApp Deployment Data Tab / Advanced Settings Subtab.

Managing Altiris Package Deployment Data

Note • *Because Symantec Altiris server only supports Windows Installer, VMware ThinApp, and legacy installer packages, the* ***Altiris Deployment Data*** *subtab is only displayed when a package of one of those deployment types is selected.*

When a Windows Installer, VMware ThinApp, or legacy installer package is imported into the Application Catalog, Application Manager mines package elements for Altiris-specific deployment data. You can view and modify data for these packages and configure command line settings by editing the properties on the **Package Information** and **Command Lines** subtabs of the **Altiris Deployment Data** tab.

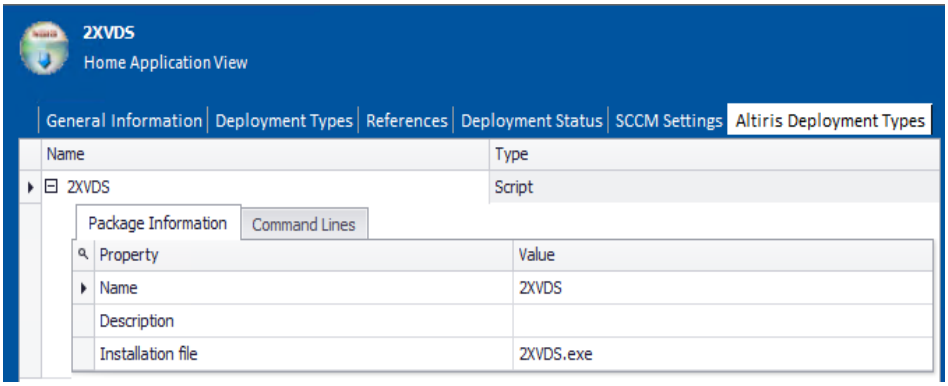


Figure 7-22: Altiris Deployment Data / Package Information Tab

Using the subtabs of the **Altiris Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

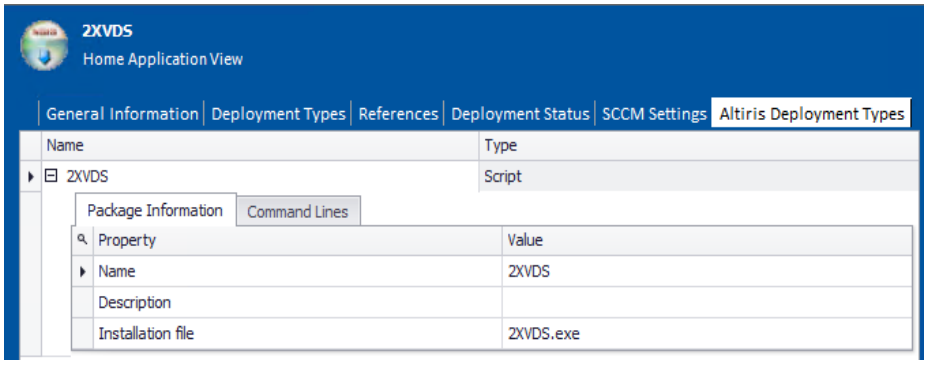
- Specifying a Package's Altiris Deployment Settings
- Specifying a Package's Altiris Deployment Command Line Settings

Specifying a Package's Altiris Deployment Settings

On the **Package Information** subtab of the **Altiris Deployment Data** tab, you can view and modify Altiris-specific data for packages.

Task ***To specify a package's Altiris deployment settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a Windows Installer, VMware ThinApp, or legacy installer package in the tree. The **Home Deployment Type View** opens.
3. Click the **Altiris Deployment Data** tab and open the **Package Information** subtab.



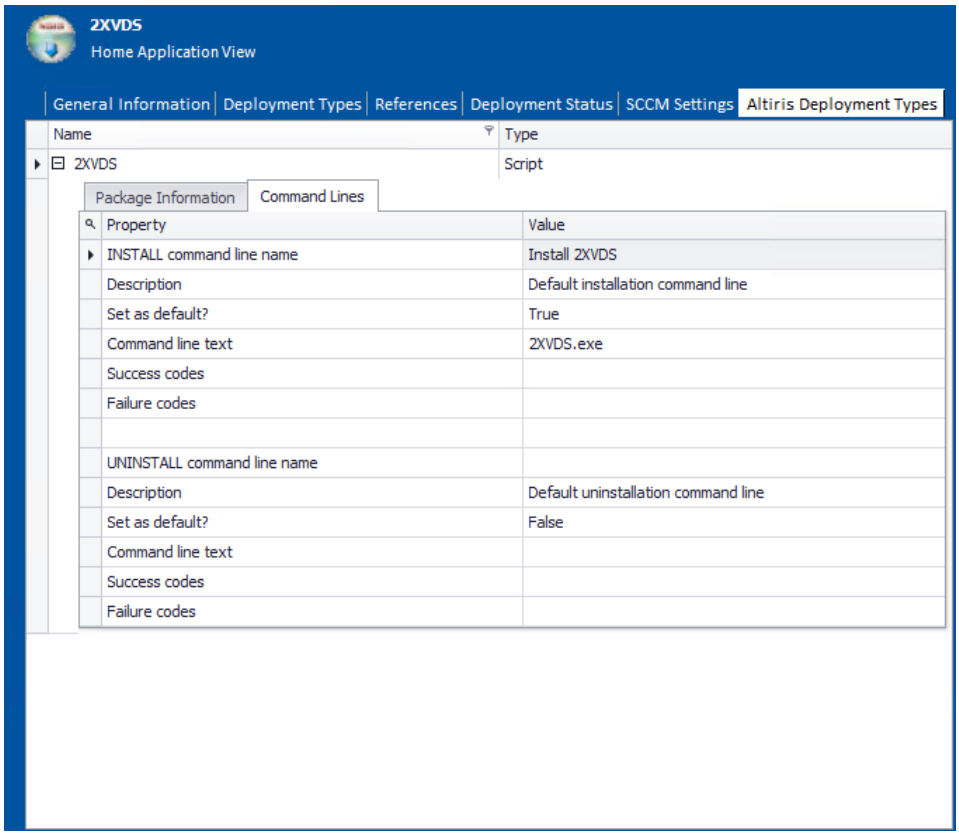
4. View and modify data, as described in Package Information Subtab in the Altiris Deployment Data Tab section.

Specifying a Package's Altiris Deployment Command Line Settings

On the **Command Lines** subtab of the **Altiris Deployment Data** tab, you can configure a package's Altiris-related command line settings.

Task ***To specify a package's Altiris command line settings:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select a Windows Installer, VMware ThinApp, or legacy installer package in the tree. The **Home Deployment Type View** opens.
3. Click the **Altiris Deployment Data** tab and open the **Command Lines** subtab.



4. View and modify data, as described in Command Lines Subtab in the Altiris Deployment Data Tab section.

Managing Workspace ONE Package Deployment Data

Edition • *Support for Workspace ONE integration is included in AdminStudio Enterprise Edition when you purchase Mobile.*

Note • *Because Workspace ONE server only supports Apple iOS, MSI, EXE, and Google Android packages, the* ***Workspace ONE Deployment Data*** *subtab is only displayed when an iOS or MSI, EXE, or Android package is selected.*

Workspace ONE is a leading global Mobile Device Management (MDM) provider. Using AdminStudio, you can manage and publish Apple iOS (local and public store), MSI package, EXE Packages, and Google Android (local and public store) mobile apps to Workspace ONE. You can view and modify data for these packages by editing the properties on the **Workspace ONE Deployment Data** tab.

MST files (excluding SoftwareId.mst) are the only MSI supporting files that will be published to Workspace ONE. Other supporting files like MSPs will be skipped during publish.

Files dependent on MSI (other than MSTs) will not be published to Workspace ONE.

Example: cabinet files, configuration xml files, .dlls, .configs etc.

Using the subtabs of the **Workspace ONE Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

- Specifying Package Details
- Specifying Package Files
- Specifying Package Deployment Options

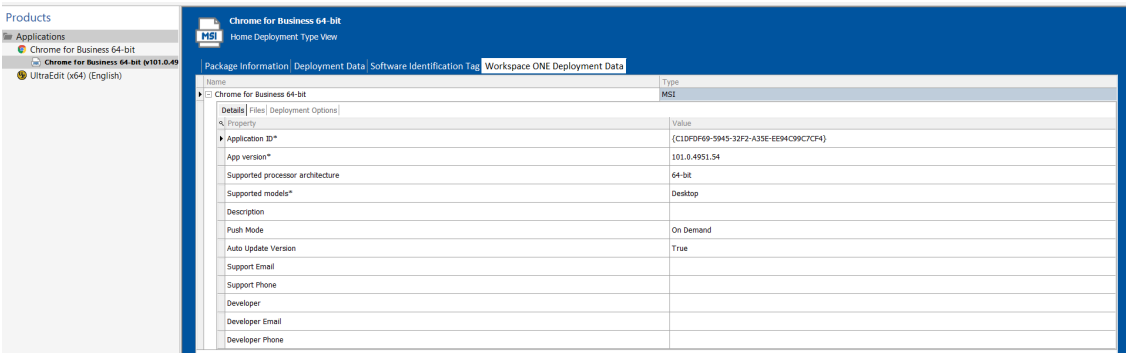
Note • *If you are using an Application Catalog that has been upgraded from a release prior to AdminStudio 2013, and the iOS application was imported prior to the upgrade, you will need to reimport the iOS application before you will be able to successfully publish it to Workspace ONE Server.*

Specifying Package Details

The **Details** subtab of the **Workspace ONE Deployment Data** tab lists general information about the package and application ID, application version, and supported modules.

Task ***To specify package details:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an iOS, EXE, MSI, or Android package in the tree. The **Home Deployment Type View** opens.
3. Click the **Workspace ONE Deployment Data** tab and open the **Details** subtab.



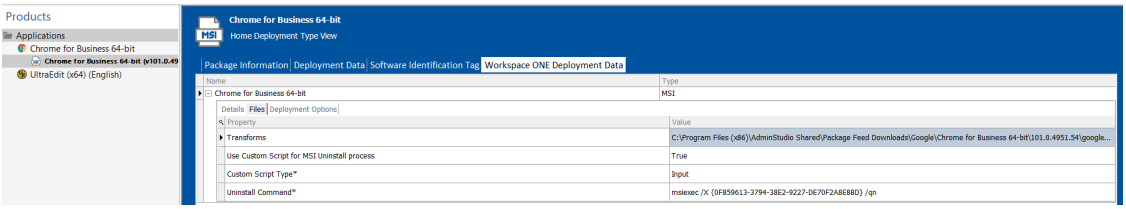
4. View and modify data, as described in Workspace ONE Deployment Data tab / Details Subtab

Specifying Package Files

The **Files** subtab of the **Workspace ONE Deployment Data** tab lists custom script type, uninstall command, and specifies the custom script for MSI and EXE uninstall process.

Task ***To specify package files:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an MSI or EXE package in the tree. The **Home Deployment Type View** opens.
3. Click the **Workspace ONE Deployment Data** tab and open the **Files** subtab.



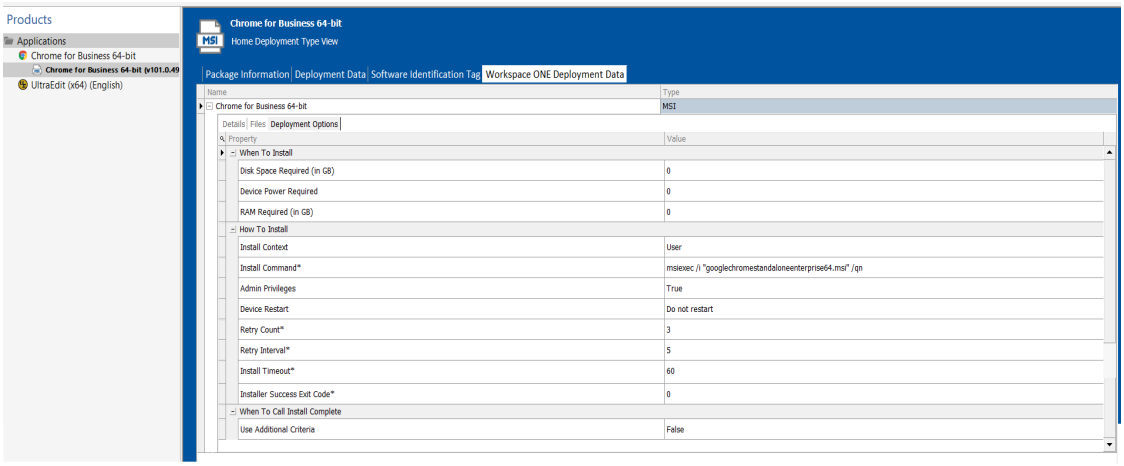
4. View and modify data, as described in Workspace ONE Deployment Data tab / Files Subtab

Specifying Package Deployment Options

The **Deployment Options** subtab of the **Workspace ONE Deployment Data** tab lists install command, When to Install, How to install, and When to Call Install Complete options to deploy this package successfully.

Task ***To specify package deployment options:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an MSI or EXE package in the tree. The **Home Deployment Type View** opens.
3. Click the **Workspace ONE Deployment Data** tab and open the **Deployment Options** subtab.

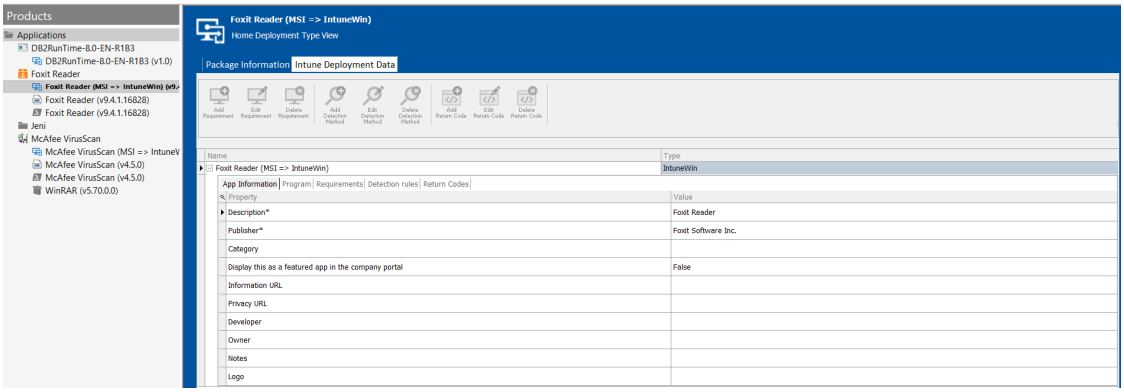


4. View and modify data, as described in Workspace ONE Deployment Data tab / Deployment Options Subtab

Managing Microsoft Intune Deployment Data

Note • *The **Intune Deployment Data** tab is only displayed when you set up Intune connection in the Distribution Systems.*

AdminStudio supports publishing of MSI, MSIX and Intunewin package types to Microsoft Intune. The Intune Deployment Data tab appears only for the supported package types. The Intune Deployment Data Tab displays various deployment related properties organized in subtabs for easy navigation. You can view and edit properties values either inline or by using the easy-to-using wizards in Detection rules, Requirements and Return code subtabs.



Using the subtabs of the **Intune Deployment Data** tab of the **Home Deployment Type View**, you can perform the following tasks:

- Specifying App Information
- Specifying Package Programs

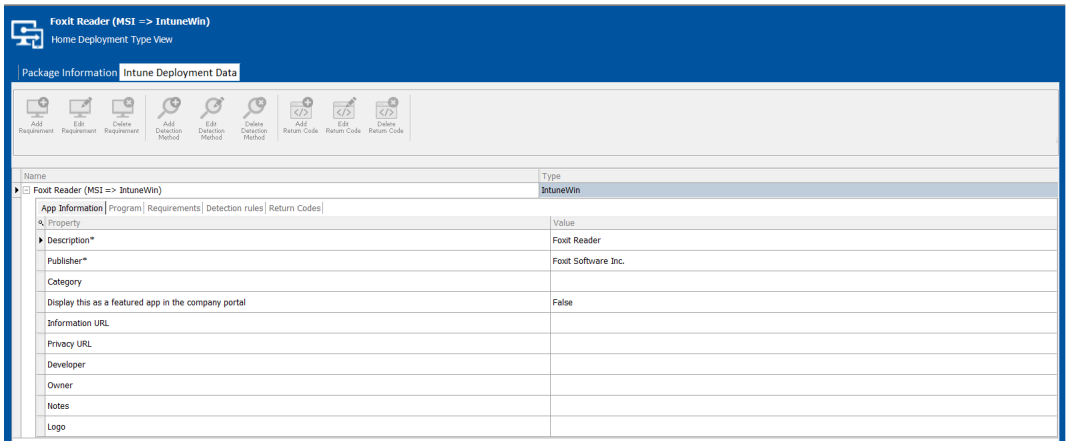
- Specifying Package Requirements
- Specifying Package Detection Rules
- Specifying Return Codes

Specifying App Information

The App Information subtab of the Intune Deployment Data tab of the Home Deployment Type View lists general information about package.

Task ***To specify app information:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an Intunewin package in the tree. The **Home Deployment Type View** opens.
3. Click the **Intune Deployment Data** tab and open the **App Information** subtab.



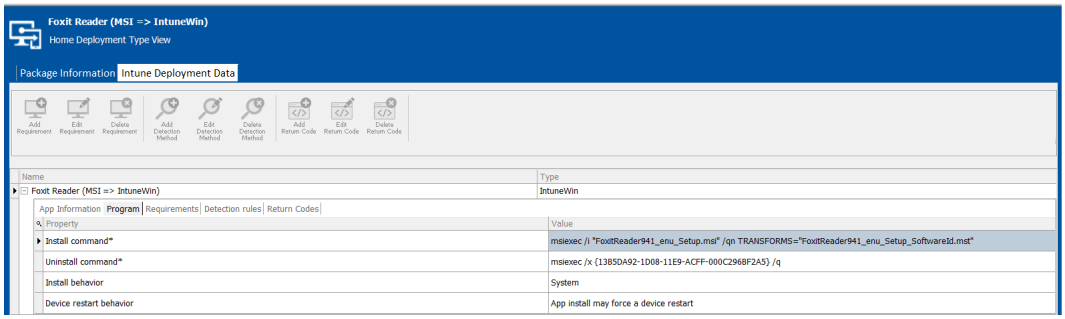
4. View and modify data, as described in Intune Deployment Data Tab / App Information Subtab.

Specifying Package Programs

The Program subtab of the Intune Deployment Data tab of the Home Deployment Type View lists command line parameters for package installation and uninstallation.

Task ***To specify package program command line parameters:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an Intunewin package in the tree. The **Home Deployment Type View** opens.
3. Click the **Intune Deployment Data** tab and open the **Program** subtab.



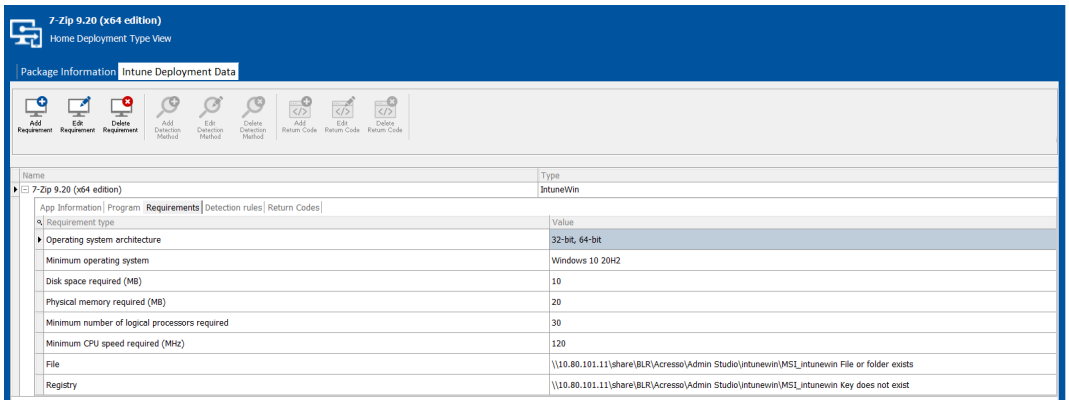
4. View and modify data, as described in Intune Deployment Data Tab / Program Subtab.

Specifying Package Requirements

You can use the **Requirements** subtab of the **Intune Deployment Data** tab to add device, or additional requirements that the target system needs to successfully deploy this package.

Task ***To specify package requirements:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an Intunewin package in the tree. The **Home Deployment Type View** opens.
3. Click the **Intune Deployment Data** tab and open the **Requirements** subtab.



4. To add requirements, select either **Device Requirement** or **Additional Requirement** and follow the steps. For more information, see Requirement Wizard of Intune Deployment Data Tab.
5. To modify an existing requirement, select the Requirement type, and click **Edit Requirement**.
6. To delete, select the **Requirement type**, and click **Delete Requirement**.

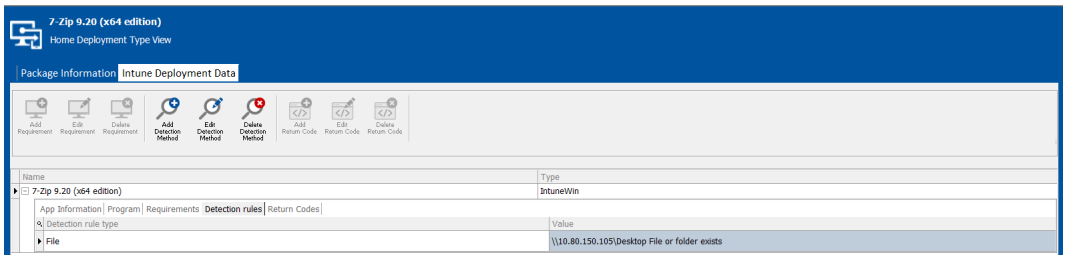
Note • *The mandatory properties can not be deleted.*

Specifying Package Detection Rules

The Detection Rules subtab of the Intune Deployment Data tab of the Home Deployment Type View lists methods to detect whether this package is already installed on the target system.

Task ***To specify package detection rules:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an Intunewin package in the tree. The **Home Deployment Type View** opens.
3. Click the **Intune Deployment Data** tab and open the **Detection rules** subtab.



4. View existing detection methods, as described in Intune Deployment Data Tab / Detection rules Subtab.
5. To add a detection rule, click the **Add Detection Method** button in the ribbon toolbar to open the Detection Rules Wizard and follow the steps. For more information, see Detection Rule Wizard of Intune Deployment Data Tab.

Note • *You can add either a custom detection script as a detection rule or manually configured detection rules. If you attempt to add a manually configured detection rule with a custom detection script, a warning message will be thrown*

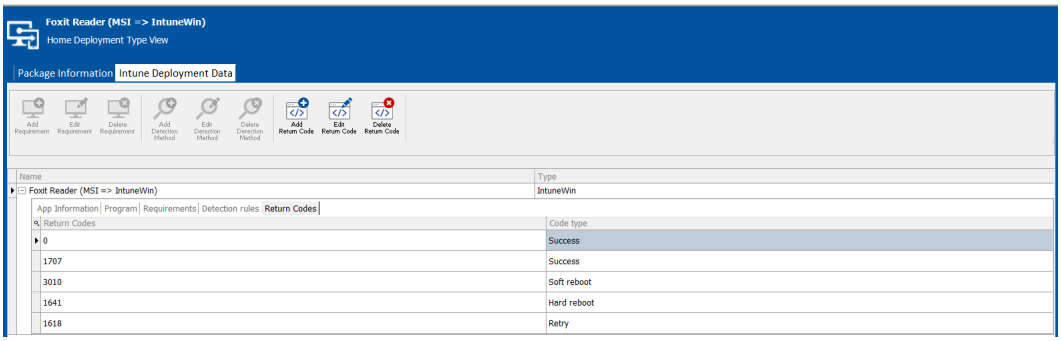
Note • *There can be a maximum 25 manually configured detection rules added to a package, attempting to add more than 25 detection rules will result in a warning message.*

6. To modify an existing detection method, select the detection method, and click **Edit Detection Method**.
7. To delete an existing detection method, click **Delete Detection Method**.

Specifying Return Codes

You can edit an Intunewin package's return codes in the Application Manager interface. Return codes are used to indicate whether a restart is required, whether an installation is complete, and to customize the text shown to users when a specific code is returned.

You can view a package's return codes on the **Return Codes** subtab of the Intune Deployment Types tab on the Home Application View or Home Deployment Type View.



The following return codes are populated by default during package import:

- 0—Success
- 1707—Success
- 3010—Soft Reboot
- 1641—Hard Reboot
- 1618—Retry

On the Return Codes tab, you can add, edit, and delete return codes.

- **Adding a return code**—Click **Add Return Code** in the ribbon to open the Return code wizard and follow the steps. For more information, see Return Code Wizard of Intune Deployment Data Tab.
- **Editing a return code**—Select a return code, click **Edit Return Code** in the ribbon, and edit the details of the return code. However, the **Return Code Value** field cannot be edited.
- **Deleting a return code**—Select a return code, click **Delete Return Code** in the ribbon, and confirm the deletion.

Managing App-V Virtual Environments

In Application Manager, you can create App-V virtual environments for App-V 5.0 packages for both Microsoft App-V Servers and Microsoft System Center 2012 Configuration Manager Servers.

App-V virtual environments enable deployed virtual applications to share the same file system and registry on client computers. This means that unlike standard virtual applications, these applications can share data with each other.

Tip • *Using virtual environments to group dependent packages together in App-V 5.0 is similar to the Dynamic Suite Composition feature used with App-V 4.x packages.*

Virtual environments are created or modified on client computers when the application is installed or when clients next evaluate their installed applications. You can order these applications so that when multiple applications attempt to modify the same file system or registry value on a client computer, the application with the highest order takes precedence.

For information on how to create App-V virtual environments, see the following topics:

- Creating an App-V Server Virtual Environment
- Creating a ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment

Creating an App-V Server Virtual Environment

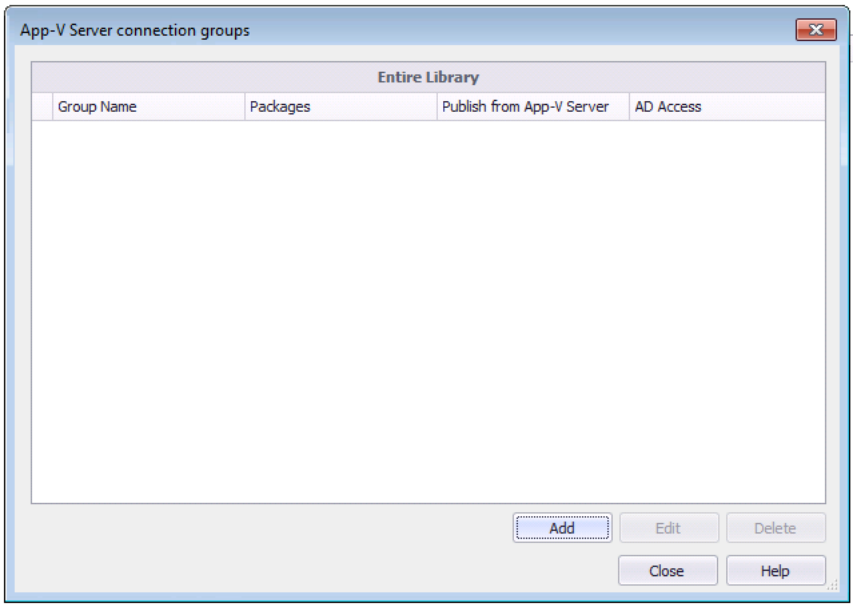
App-V Server virtual environments are called *connection groups*. Connection groups contain the name of groups that a package is associated with. You can create or edit a connection group using the **App-V Server Connection Groups** dialog box, which can be opened using either of these methods:

- **From the ribbon**—On the Application Manager **Home** tab, click **App-V Virtual Environments > App-V Server Environment** in the ribbon.
- **From the Home Deployment Type View**—With the Application Catalog **Home** tab selected, select an App-V 5.0 package in the tree to open the **Home Deployment Type View**. Then open the **App-V Deployment Data > Advanced Settings** tab and click in the **Connection Group** field.

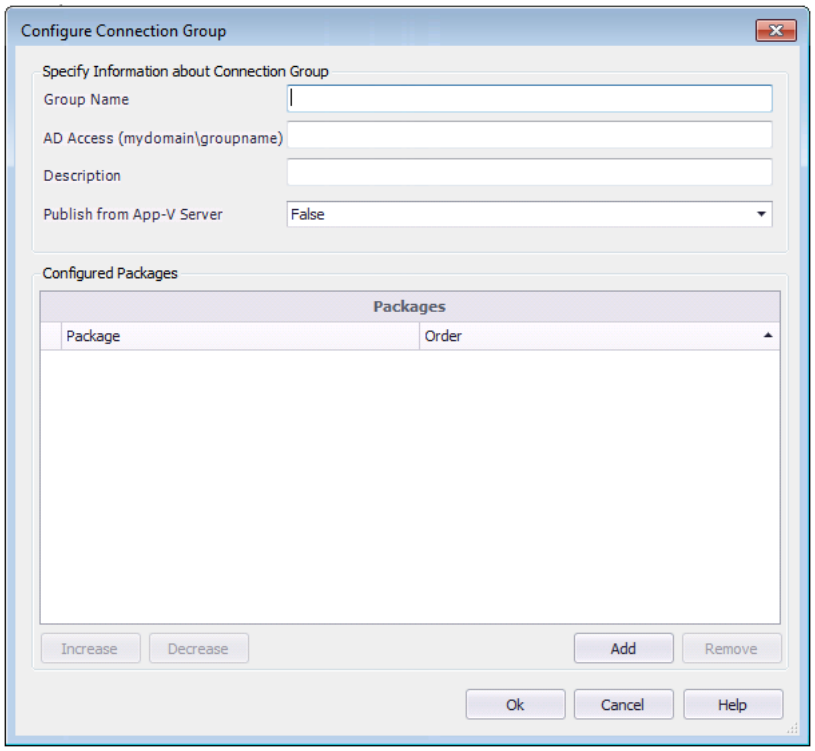
To create an App-V Server connection group for App-V 5.0 packages, perform the following steps:

Task ***To create an App-V Server connection group:***

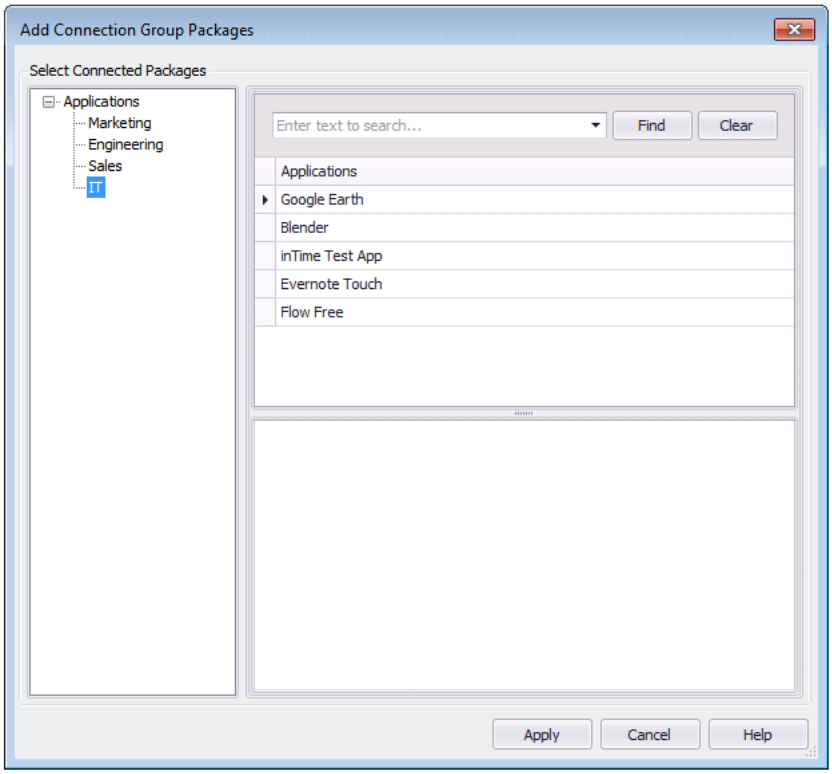
1. Open the **Home** tab of Application Catalog.
2. In the ribbon, click **App-V Virtual Environments > App-V Server Environment**. The **App-V Server Connection Groups** dialog box opens.



3. Click **Add** to create a new connection group. The **Configure Connection Group** dialog box opens.



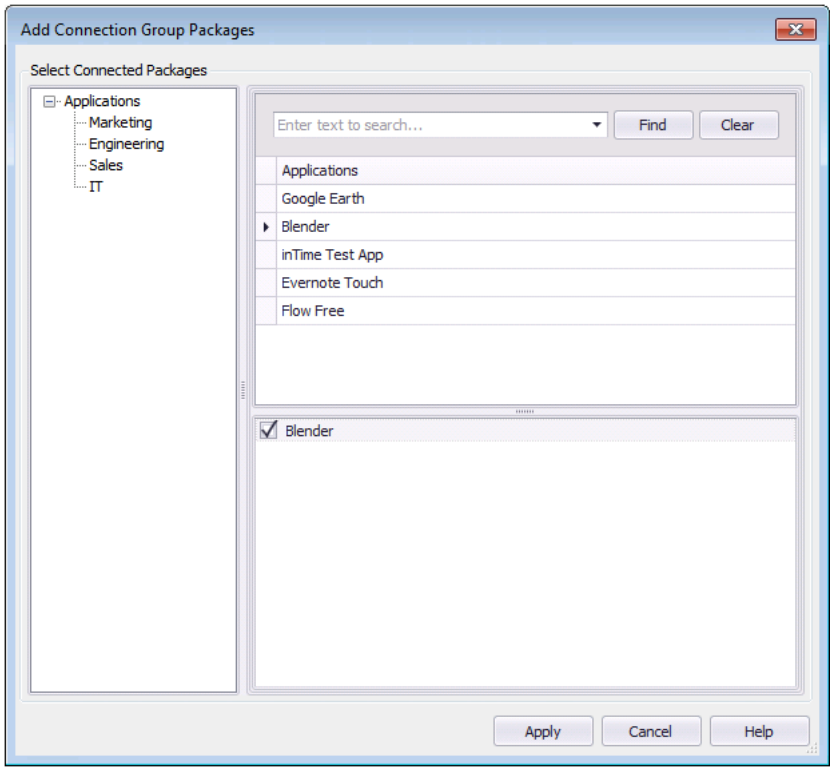
4. In the **Group Name** field, enter a name to identify this new connection group.
5. In the **AD Access** field, enter the name of the Active Directory group that will have permission to access this connection group.
6. In the **Description** field, enter a description of the purpose of this connection group.
7. From the **Publish from App-V Server** list, select one of the following options:
   - **False**—Do not publish from App-V server.
   - **True**—Publish from App-V server.
8. To add App-V packages to this connection group, click **Add**. The **Add Connection Group Packages** dialog box opens.
9. Under **Select Connected Packages**, select the group in the tree that contains the App-V package that you want to add to the connection group. The packages in that group are listed under **Applications**.



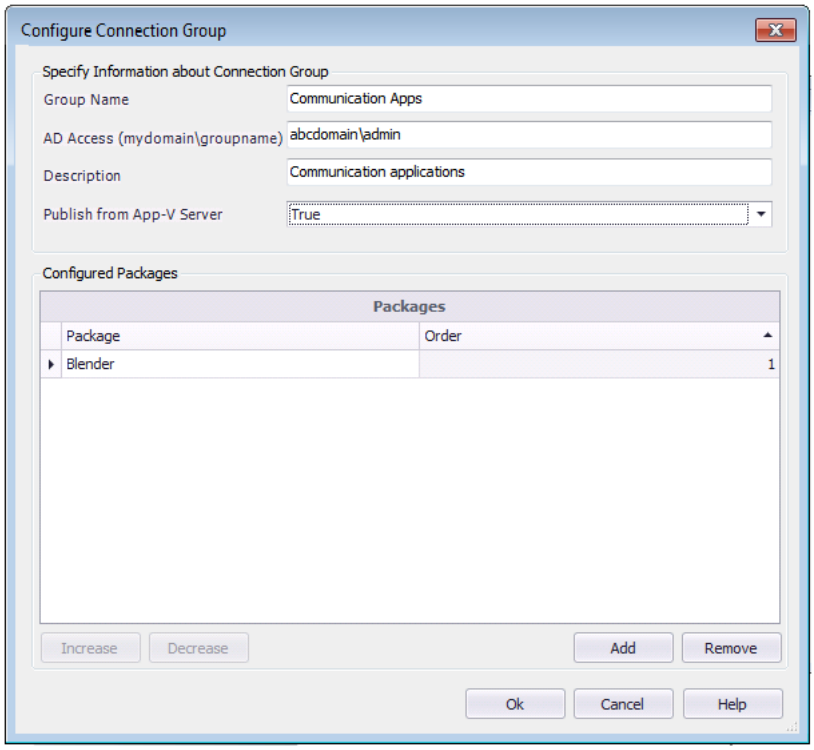
10. Under **Applications**, select an application that contains an App-V 5.0 package. The App-V 5.0 package is listed in the lower pane.

Note • *If you select an application that does not have an App-V 5.0 deployment type, nothing will be listed in the lower pane.*

11. Select the App-V 5.0 package and click **Apply**.



The App-V 5.0 package is now listed under **Configured Packages** on the **Configure Connection Group** dialog box.



12. Repeat above steps to add additional App-V 5.0 packages to the connection group.

Note • *The order of packages in the connection group is important. This determines the order in which the package contents are merged. So, if there was a conflict (example: same registry value), the content of the first package would be used.*

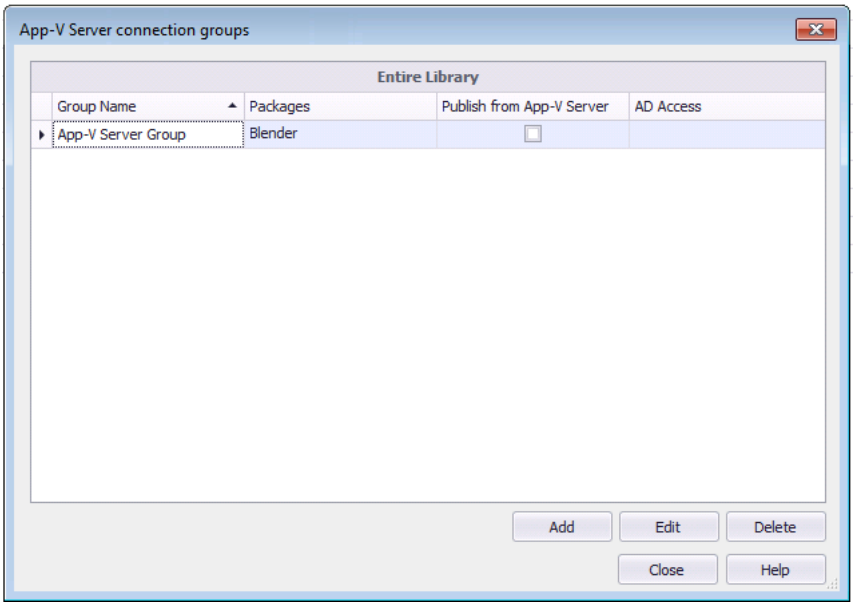
13. When you are done adding App-V 5.0 packages to the connection group, click **OK**. The new connection group is now listed on the **App-V Server Connection Groups** dialog box.
14. Click **Close**.

Editing an Existing App-V Server Virtual Environment

To edit an existing App-V Server connection group, perform the following steps:

Task ***To edit an existing App-V Server connection group:***

1. Open the **Home** tab of Application Catalog.
2. Click **App-V Virtual Environments > App-V Server Environment** in the ribbon. The **App-V Server Connection Groups** dialog box opens, listing any defined App-V Server connection groups.



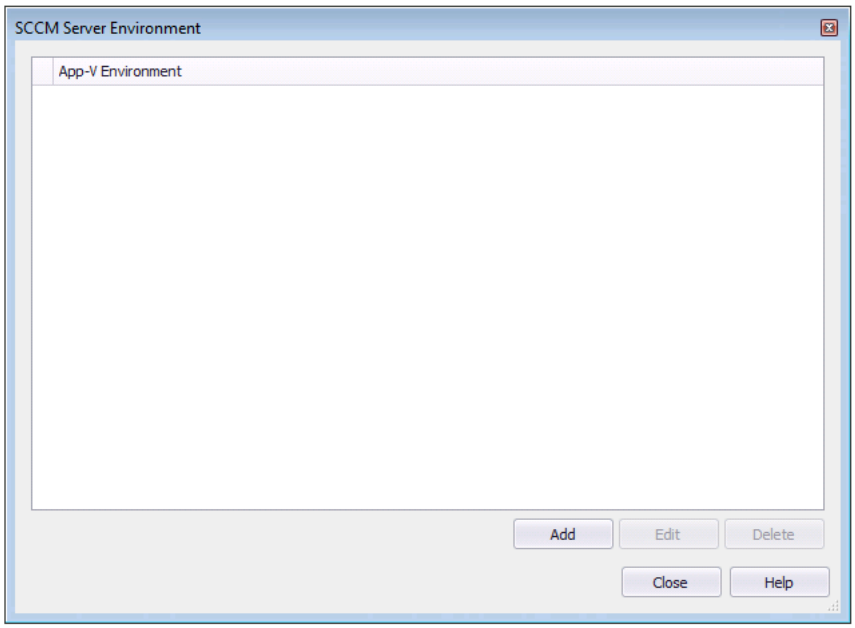
3. Select the App-V Server connection group that you want to edit and click **Edit**.
4. Proceed with your edits, as described in Creating an App-V Server Virtual Environment.

Creating a ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment

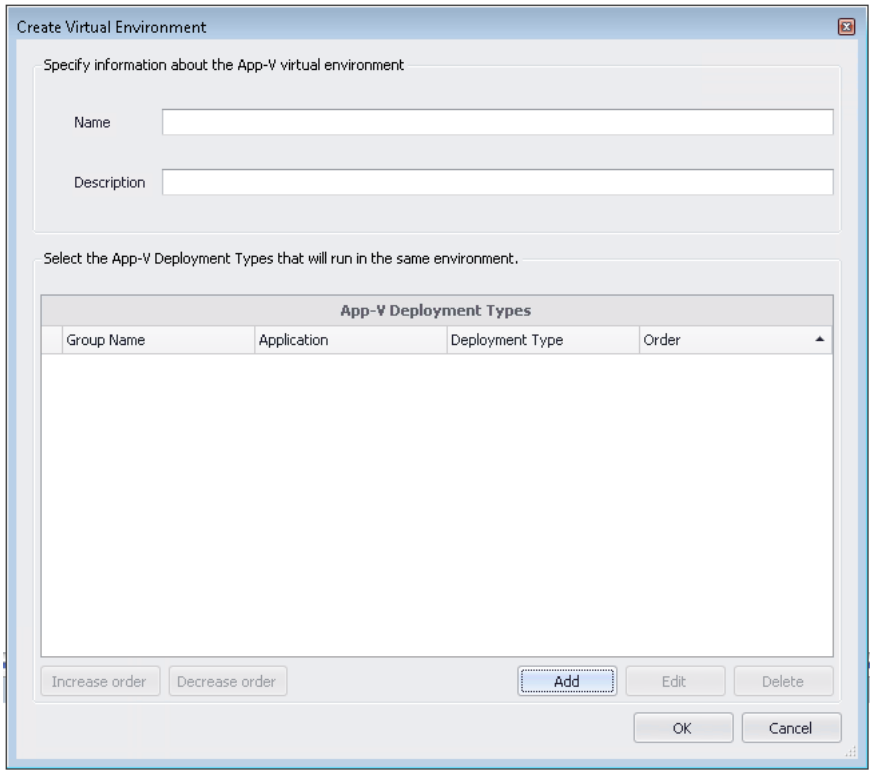
To create a ConfigMgr (Formerly called as System Center Configuration Manager) Server App-V virtual environment for App-V 5.0 packages, perform the following steps:

Task ***To create a ConfigMgr (Formerly called as System Center Configuration Manager) Server App-V virtual environment:***

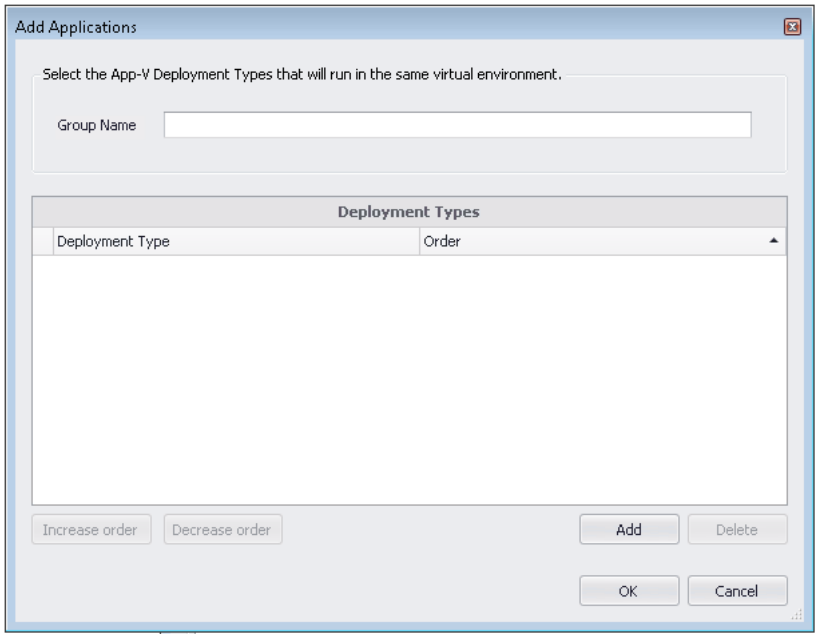
1. Open the **Home** tab of Application Catalog.
2. Select a package in the tree and open the **Deployment Data** subtab of the **Home Deployment Type View**.
3. Click **App-V Virtual Environments** in the ribbon and then select **SCCM Server Environment**. The **SCCM Server Environment** dialog box opens.



4. Click **Add**. The **Create Virtual Environment** dialog box opens.

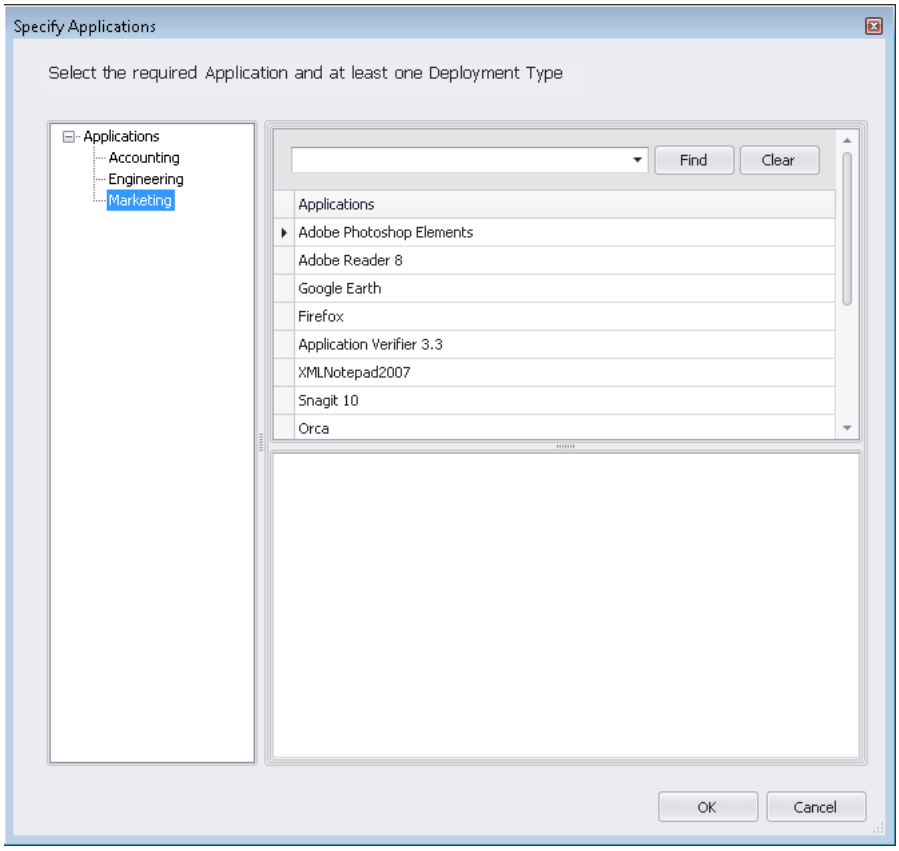


5. In the **Name** field, enter a name to identify this virtual environment.
6. In the **Description** field, enter a description of the purpose of this virtual environment.
7. Click **Add** to add an App-V deployment type group. The **Add Applications** dialog box opens.



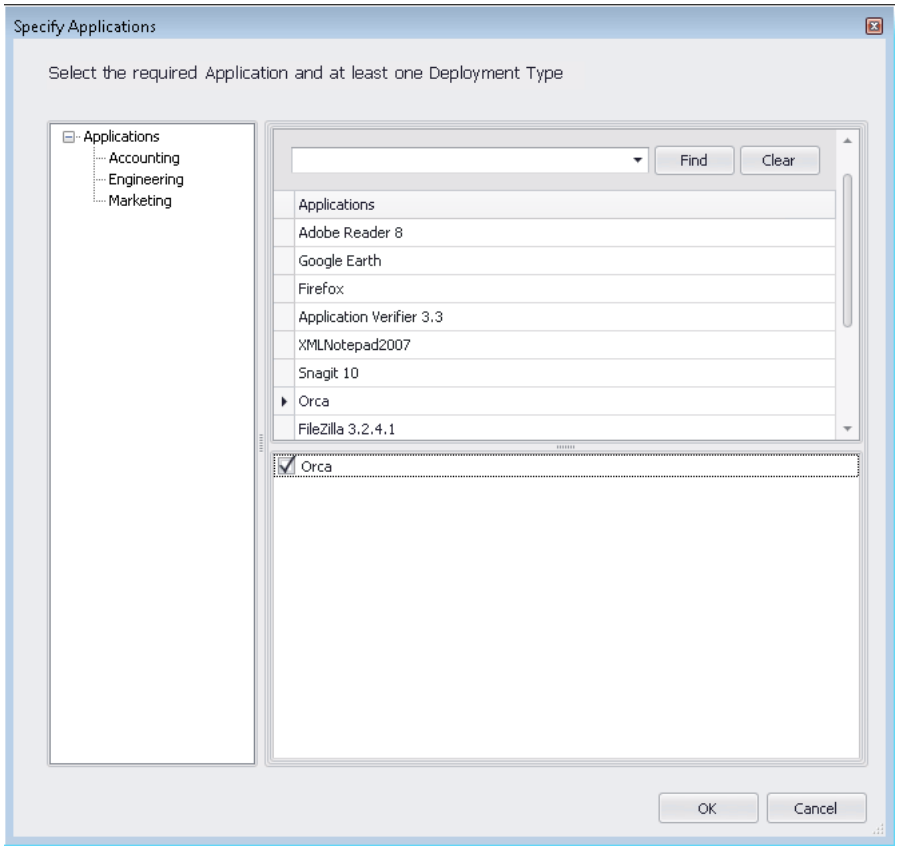
8. Enter a **Group Name** to identify the group of App-V 5.0 packages that you are going to add.

9. Click **Add**. The **Specify Applications** dialog box opens, which provides a tree structure that you can use to select an App-V 5.0 application.



10. In the tree structure on the left, select the group that contains the App-V 5.0 package that you want to add. The names of the applications in that group are listed in the top pane.

11. In the top pane, select the application that contains the App-V 5.0 package that you want to add. The App-V 5.0 deployment type is listed in the lower pane.



12. Select the App-V 5.0 deployment type and click **OK**. The selected package is now listed on the **Add Applications** dialog box.

Note • *If more than one deployment type is listed on the **Add Applications** dialog box, you could use the **Increase order** and **Decrease order** buttons to reorder the list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.*

13. Click **OK**. The group containing the selected App-V 5.0 package is now listed on the **Create Virtual Environment** dialog box.

Note • *If more than one group is listed on the **Create Virtual Environment** dialog box, you could use the **Increase order** and **Decrease order** buttons to reorder the list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.*

14. Click **OK**. The new App-V virtual environment is now listed on the **SCCM Server Environment** dialog box.

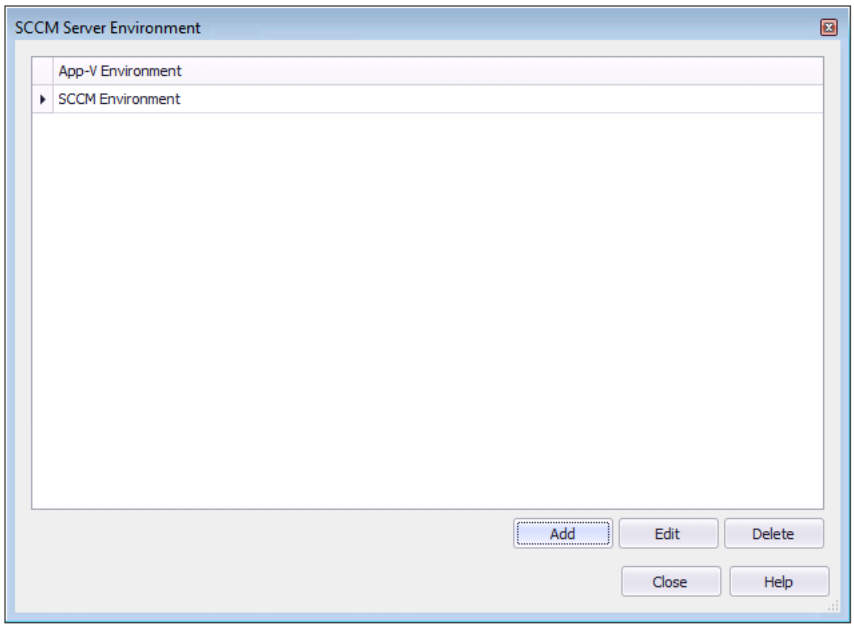
15. Click **Close**.

Editing an Existing ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment

To edit an existing ConfigMgr (Formerly called as System Center Configuration Manager) Server virtual environment, perform the following steps:

Task ***To edit an existing ConfigMgr (Formerly called as System Center Configuration Manager) Server virtual environment:***

1. Open the **Home** tab of Application Catalog.
2. Select a package in the tree and open the **Deployment Data** subtab of the **Home Deployment Type View**.
3. Click **App-V Virtual Environments** in the ribbon and then select **SCCM Server Environment**. The **SCCM Server Environment** dialog box opens, listing any defined App-V virtual environments.



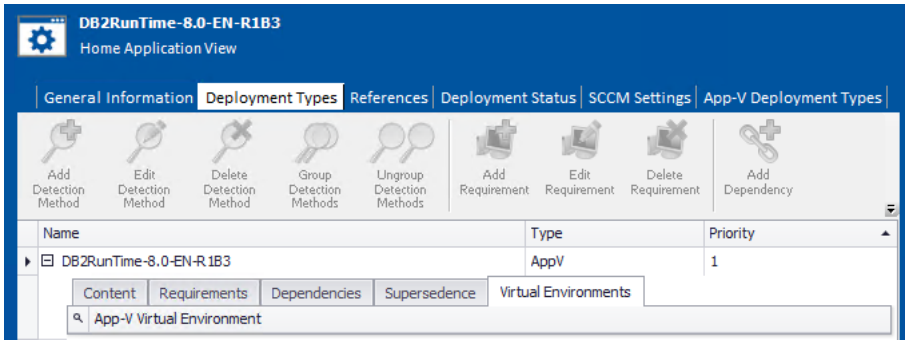
4. Select the virtual environment that you want to edit and click **Edit**.
5. Proceed with your edits, as described in Creating a ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environment.

Viewing a Package's ConfigMgr (Formerly called as System Center Configuration Manager) Server Virtual Environments

You can view the virtual environments that a package is a member of on the **Virtual Environments** subtab of the **Home Deployment Type View > Deployment Data** tab of an App-V 5.0 package.

Task ***To view an existing App-V virtual environment:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Select an App-V 5.0 package in the tree. The **Home Deployment Type View** opens.
3. Click the **Deployment Data** tab and open the **Virtual Environments** subtab. Existing virtual environments that the package is a member of are listed.



Viewing Additional Package Data

If you click on the plus sign to expand a package in the Application Catalog **Home Deployment Type View**, a node is listed for each available constituent view. For example, for a Windows Installer package when the **Home** tab is selected, the following nodes are listed:

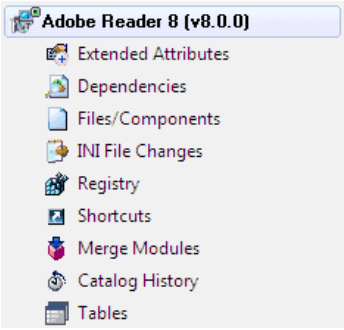


Figure 7-23: Windows Installer Package Nodes / **Home** Tab

When you select one of these nodes, a constituent view opens in the right pane. The following types of package data can be viewed by selecting a package subnode in the **Home Deployment Type View**:

- Viewing and Editing Package Extended Attributes
- Viewing Package Dependencies
- Viewing Package Files, Components, and Directories
- Viewing Windows Installer Package INI File Changes
- Viewing Registry Information

- Viewing Package Shortcuts
- Viewing Windows Installer Package Merge Modules
- Viewing Package Catalog History
- Viewing App-V Package History
- Viewing Package Tables
- Viewing App-V Package File Type Associations
- Viewing App-V Package Environment Variables

Viewing and Editing Package Extended Attributes

Extended attributes are optional attributes for packages, defined by an Package Extended Attribute Description File (in XML format). Because you can manually create the description file, you have the flexibility to include information about each package that may be specific to your organization—such as the users or business groups that receive the package.

Extended attributes can be edited in the **Extended Attributes View** on a package-by-package basis.

The following tasks and concepts relate to extended attributes:

- Using Package Extended Attributes
- Package Extended Attribute Description File
- Integrating Package Extended Attribute Data with a Workflow Request

Note • *In addition to Windows Installer packages, you can also view and edit Extended Attributes for Microsoft App-V, Citrix XenApp, and VMware ThinApp virtual packages, as well as for non-Windows Installer legacy packages.*

Using Package Extended Attributes

Assuming you have created a package extended attribute description file (or are using the default provided file), you can configure Application Catalog to use it with the current Application Catalog.

Task ***To use extended attributes in Application Manager:***

1. On the Application Catalog tab menu, click **Options**. The **Options** dialog box opens.
2. Under **General Options**, click **General**. The **General** tab opens.
3. In the **Extended Attribute Description File** field, specify or browse to the extended attribute description file (`.xml`) containing the extended attributes you want to use for Application Catalog.
4. Click **OK**.

The **Extended Attributes View** is available under each package in Application Manager.

Caution • *The default Extended Attribute description file is named `EA_Default.xml`, and is installed in the `AdminStudio Shared` folder. You can modify the data displayed in the* ***Extended Attributes*** *view, but to do this, do not edit the `EA_Default.xml` file. Instead, copy the `EA_Default.xml` file, rename it, make your edits to the new file, and then enter the new file name and location in the* ***Extended Attribute Description File*** *field on the* ***General*** *tab of the Application Manager Options dialog box.*

Package Extended Attribute Description File

Application Catalog uses an XML file to describe the data that appears in a package's **Extended Attributes** view. The name and location of this XML file can be specified on the **General** tab of the Application Manager Options dialog box.

Caution • *The default package Extended Attribute description file is named `EA_Default.xml`, and is installed in the `AdminStudio Shared` folder. You can modify the data displayed in the* ***Extended Attributes*** *view, but to do this, do not edit the `EA_Default.xml` file. Instead, copy the `EA_Default.xml` file, rename it, make your edits to the new file, and then enter the new file name and location in the* ***Extended Attribute Description File*** *field on the* ***General*** *tab of the Application Manager Options dialog box.*

Description File Properties

The description file, which is in XML format, contains tags for each extended attribute (up to a limit of 400 attributes). It supports text or file values. The following list explains each tag available in the description file.

Table 7-21 • Description File Tags

Attribute	Description
UniqueIdentifier	This value, which Application Catalog uses to validate that the XML file is for extended attributes, must be set to ISASEA40.
Name	The name of the attribute as it appears in the Extended Attributes view. This cannot exceed 255 characters.
Type	The extended attribute type. This can be Text, File, or Selection. If no type is specified, then Application Catalog defaults the attribute to text.
DefaultValue	This tag, available only for Text types, provides the default value for the attribute. This optional value cannot exceed 512 characters.
DefaultFileExtension	This tag, available only for File types, provides the default file extension when you browse for the file. Examples of this could be *.txt, *.bmp, *.doc, or *.* (representing all files).

Table 7-21 • Description File Tags

Attribute	Description
FileFilter	Provide the file types to populate the File type filter in the Browse dialog box. These must be in pairs, and in the format Longname (*.ext)|*.ext. Before the closing </FileFilter> tag, you must have two pipe symbols (||). For example, to include filters for text files, bitmaps, and all files, use the following line: `<FileFilter>Text Document (*.txt)|*.txt|Bitmap (*.bmp)|*.bmp|All Files (*.*)|*.*||</FileFilter>` This value cannot exceed 255 characters.
Caption	The caption for the Browse dialog box when using File types. This cannot exceed 255 characters.
Values	Used only for Selection types, this is a semicolon-delimited list of possible values for the selection. These will appear in a drop-down list for the extended attribute. The first value is used as the default. The total number of characters of all the values and necessary semicolons cannot exceed 255 characters.
HelpText	Text that appears below the value field for either Text or Selection attributes. You can use it to provide additional information to help users know what to input. This cannot exceed 512 characters.

Description File Format

An example of an extended attribute description file follows:

```
<Extended_Attribute UniqueIdentifier="ISASEA40">
   <AttributeDetails>
      <Name>Owner</Name>
      <Type>Text</Type>
      <DefaultValue></DefaultValue>
      <HelpText>Provide the name of the package's owner.
      </HelpText>
   </AttributeDetails>
   <AttributeDetails>
      <Name>Test Script</Name>
      <Type>File</Type>
      <DefaultFileExtension>*.*</DefaultFileExtension>
      <FileFilter>All Files (*.*)|*.*||</FileFilter>
      <Caption>Test Script Files</Caption>
   </AttributeDetails>
   <AttributeDetails>
      <Name>Program Type</Name>
      <Type>Selection</Type>
      <Values>Office Application;Utility;
      Graphic Application;Programming Application;Game;Other
      </Values>
      <HelpText>Select the type of application from the
      above list.
      </HelpText>
```

```
   </AttributeDetails>
</Extended_Attribute>
```

Integrating Package Extended Attribute Data with a Workflow Request

Note • *AdminStudio Workflow Manager is a Web-based application management system that has integrated functionality with AdminStudio.*

Application Catalog allows you to integrate extended attributes with AdminStudio Workflow Manager. This option is enabled by selecting the **Integrate with Workflow Manager** option on the **General** tab of the Application Manager Options dialog box.

When the **Integrate with Workflow Manager** option is selected, you can associate extended attribute data for packages in Application Manager with workflows in Workflow Manager. This is accomplished by right-clicking on the package name in the Application Catalog tree and selecting **Associate with Workflow Manager Workflow Request** from the shortcut menu.

Task ***To associate a package with a Workflow Manager workflow request:***

1. Open Application Manager.
2. Connect to the AdminStudio Enterprise Server Application Catalog. See Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog.
3. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.
4. In the **Extended Attributes** area of the **General** tab, confirm that the **Integrate with Workflow Manager** option is selected.
5. In the Application Catalog tree, right-click on the product that you want to link to Workflow Manager and select **Associate with Workflow Manager Workflow Request** from the shortcut menu.

   The **Associate with Workflow Manager Workflow Request** dialog box opens.
6. Pick the application in Workflow Manager with which you want to associate this product.

Assuming that Workflow Manager is configured to use the same extended attributes file, if you enter new data into the **Extended Attributes** view in Application Manager for a package, Workflow Manager automatically detects it; if changes are made in Workflow Manager, they are automatically reflected in Application Manager.

By design, extended attributes data in Application Manager and Workflow Manager data have a one-to-one relationship. You can only associate one Workflow Manager workflow with a package in Application Manager; once the workflow is associated, it is no longer available for association with other Application Catalog packages.

Note • *Another integration feature between AdminStudio and Workflow Manager is that when you associate a package with a Workflow Manager workflow request and then view that workflow request's Workflow Report, there is a link to open the Package Report of its associated package. There is also a link on the Package Report to open the Workflow Report of its associated workflow request.*

Viewing Package Dependencies

You can view package dependency information for both Windows Installer and App-V 4.x packages:

- Viewing Windows Installer Package Dependencies
- Viewing App-V Package Dependencies

Viewing Windows Installer Package Dependencies

On the **Dependencies View**, which is accessed by selecting the **Dependencies** node under a Windows Installer package in the **Home Deployment Type View,** you can view a list of all of the files of a selected package that have dependencies with files used by other packages or operating systems in the Application Catalog. This view is displayed for Windows Installer *EA_Default.xml* packages in which file dependency information exists.

Note • *If the **Only Display View Nodes With Data** option on the **General** tab of the Application Manager Options dialog box is selected, if no dependencies are found, the **Dependencies** node will not be displayed.*

File-level package dependency information is extracted using the **Auto detect dependencies** option of the **Dependency Wizard**, as described in Scanning for Dependencies.

Task ***To view Windows Installer package dependencies:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **Dependencies** node. The **Dependencies View** opens.
3. Make selections from the **Files With Dependencies** list to further refine this listing, or select **(All)** to display all dependencies.
4. Review the following information:

Option	Description
File Name	Name of the file contained in the Windows Installer package; all other columns describe dependencies for this file.
Architecture	Machine architecture for the file.
16 bit	Signifies whether the file is meant for 16-bit machines.
Terminal Server Aware	Signifies whether the file is Terminal Server aware or not.
.NET Assembly	Shows **NotCLR** if the file is not a .NET assembly; otherwise it displays the version of .NET it depends upon.
SubSystem	Signifies the sub-system for the file.
Signed	Signifies whether the file is digitally signed.

Option	Description
Signee	If the file is signed, this columns lists the name of the signee.
PE Dependent on	Lists other files that this one depends on.
PE Language	The language for the file.

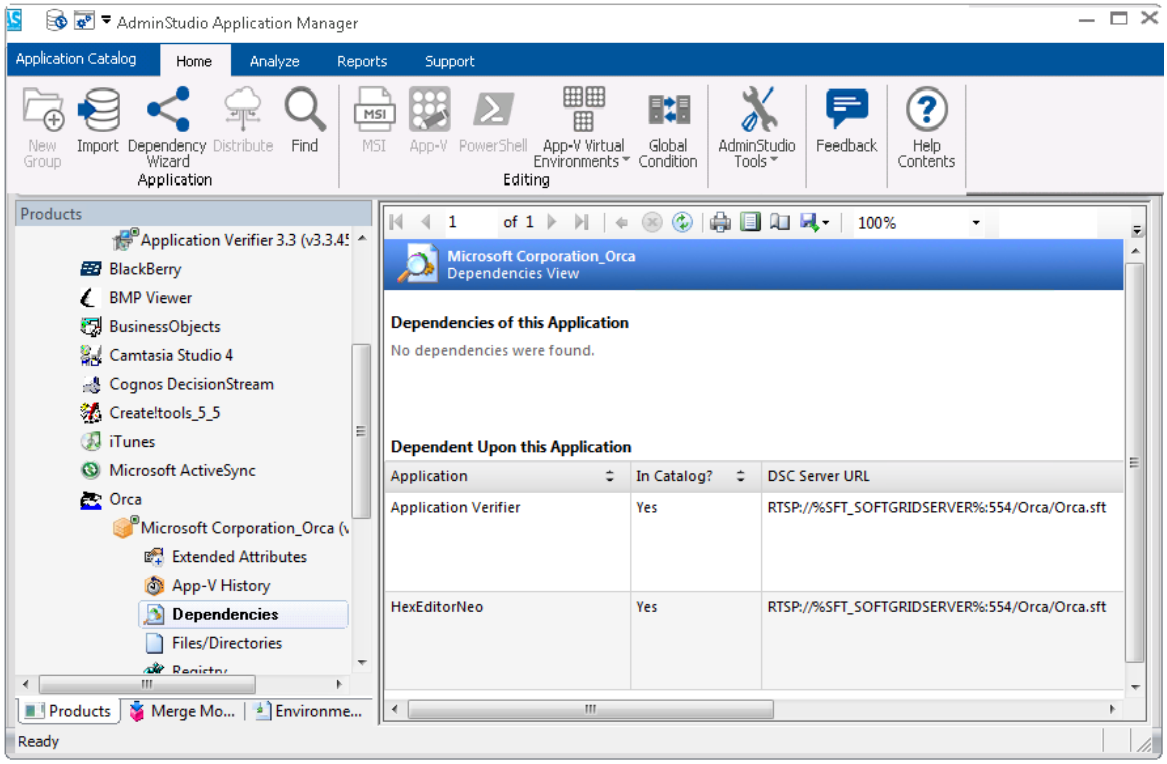
Viewing App-V Package Dependencies

Note • *This information applies to App-V 4.x packages.*

The App-V **Dependencies View** lists both the applications the App-V package is dependent on and the applications dependent upon this App-V package. To view App-V package dependencies, perform the following steps.

Task ***To view App-V package dependencies:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. In the tree, expand an App-V package and select the **Dependencies** node. The **Dependencies View** opens.



This view lists both the applications this App-V package is dependent on and the applications dependent upon this App-V package. For each dependency, the following information is listed:

- Application

- In Catalog? (Yes / No)
- DSC Server URL
- Server URL
- Mandatory? (Yes / No)

Viewing Package Files, Components, and Directories

You can view files, components, and directories information for both Windows Installer and App-V packages:

- Viewing Windows Installer Package Files and Components
- Viewing App-V Package Files and Directories

Viewing Windows Installer Package Files and Components

To display the files and components in a Windows Installer package, expand the Windows Installer package in the Application Catalog tree and select the **Files/Components** node.

Task ***To view Windows Installer package Files/Components:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **Files/Components** node. The **Files/Components View** opens.
3. Review the following information:

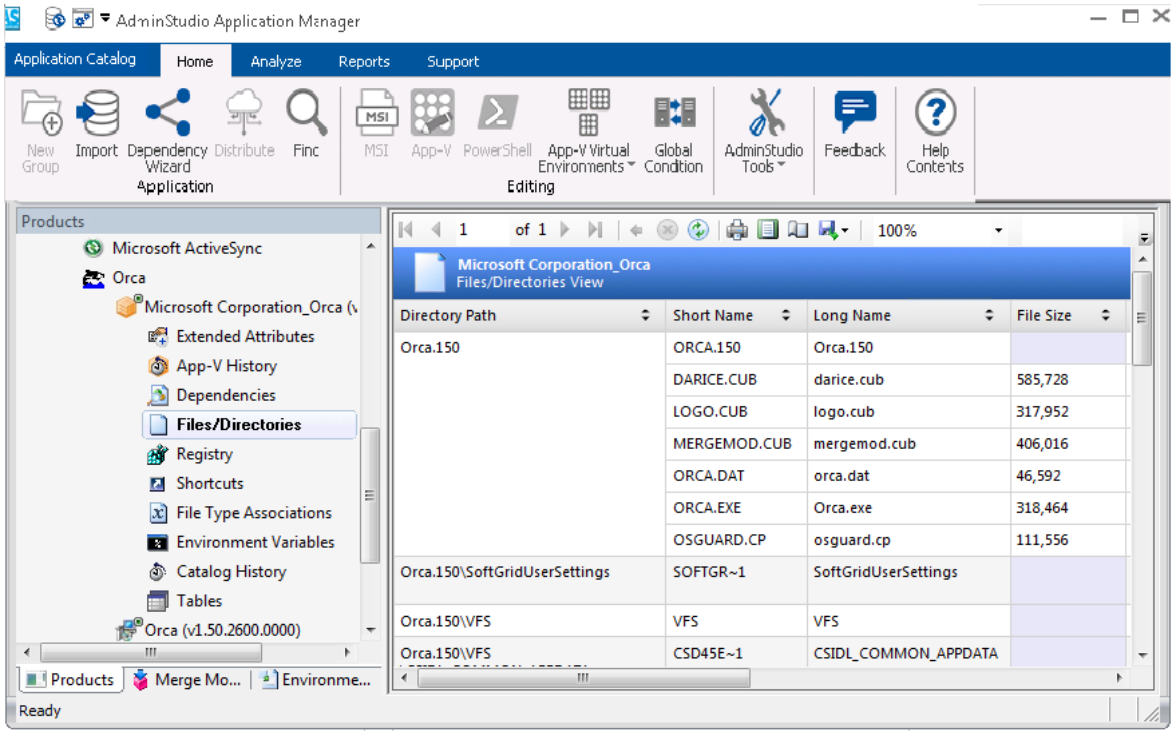
Column	Description
Component	Name of component that the file listed in the FileName column is associated with.
FileName	Name of file.
FileSize	Size of the file listed in the FileName column.
Version	Version of the file listed in the FileName column.
Path	Installation location of the file listed in the FileName column.

Viewing App-V Package Files and Directories

The App-V **Files/Directories View** lists the files and directories included in the App-V package.To view App-V package files and directories, perform the following steps.

Task ***To view App-V package files and directories:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. In the tree, expand an App-V package and select the **Files/Directories** node. The **Files/Directories View** opens.



The following information is listed for each file/directory:

Column	App-V 4.x	App-V 5
Directory Path	X	X
Short Name	X	
Long Name	X	
File Name		X
File Size	X	X
VFS Path	X	
Feature Block 1	X	X
App-V Version	X	
App-V Data Type	X	

Viewing Windows Installer Package INI File Changes

To display any INI file changes made by a Windows Installer package, expand the Windows Installer package in the Application Catalog tree and select the **INI File Changes** node.

Task ***To view Windows Installer package INI file changes:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **INI File Changes** node. The **INI File Changes View** opens.
3. Review the following information:

Column	Description
Component	Name of component that makes an entry in the INI File.
FileName	Name of INI File that the component listed in the Component column makes an entry in.
DirProperty	The directory location where the INI File will be installed.
Section	The section of the INI file where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Viewing Registry Information

You can view registry information for both Windows Installer and App-V packages:

- Viewing Windows Installer Package or Microsoft UWP App Package Registry Information
- Viewing App-V Package Registry Information

Note • *AdminStudio must be installed on a Windows 8 or higher operating system in order to import registry information for Microsoft UWP app packages (.appx) or Windows 8 app packages (.appx).*

Viewing Windows Installer Package or Microsoft UWP App Package Registry Information

Note • *AdminStudio must be installed on a Windows 8 or higher operating system in order to import registry information for Microsoft UWP app packages (.appx) or Windows 8 app packages (.appx).*

To display any registry entries created or changed by a Windows Installer package, expand the Windows Installer package in the Application Catalog tree and select the **Registry** node.

Task ***To view Windows Installer package Registry information:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **Registry** node. The **Registry View** opens.
3. Review the following information:

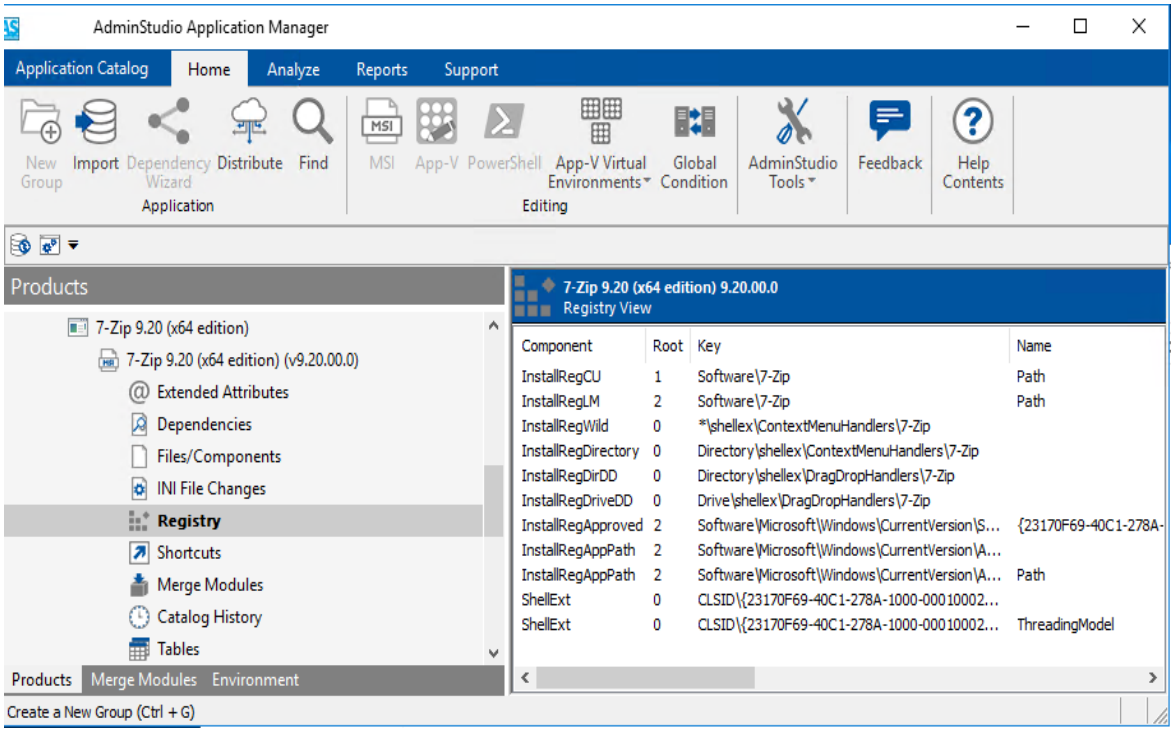
Column	Description
Component	The name of the component that is creating a Registry entry.
Root	Default value of Key.
Key	Key of the Registry Entry that this component is making.
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Viewing App-V Package Registry Information

The App-V **Registry View** lists any registry entries created or changed by the App-V package. To view the App-V **Registry View**, perform the following steps.

Task ***To view App-V package Registry information:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. In the tree, expand an App-V package and select the **Registry** node. The **App-V Registry View** opens.



For each registry entry, the following information is listed:

- Key Path
- App-V Override
- Value Name
- Data
- Type

Viewing Package Shortcuts

You can view shortcut information for both Windows Installer and App-V packages:

- Viewing Windows Installer Package Shortcuts
- Viewing App-V Package Shortcuts

Viewing Windows Installer Package Shortcuts

To display any shortcuts created by a Windows Installer package, expand the Windows Installer package in the Application Catalog tree and select the **Shortcuts** node.

Task ***To view Windows Installer package Shortcuts:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **Shortcuts** node. The **Shortcuts View** opens.
3. Review the following information:

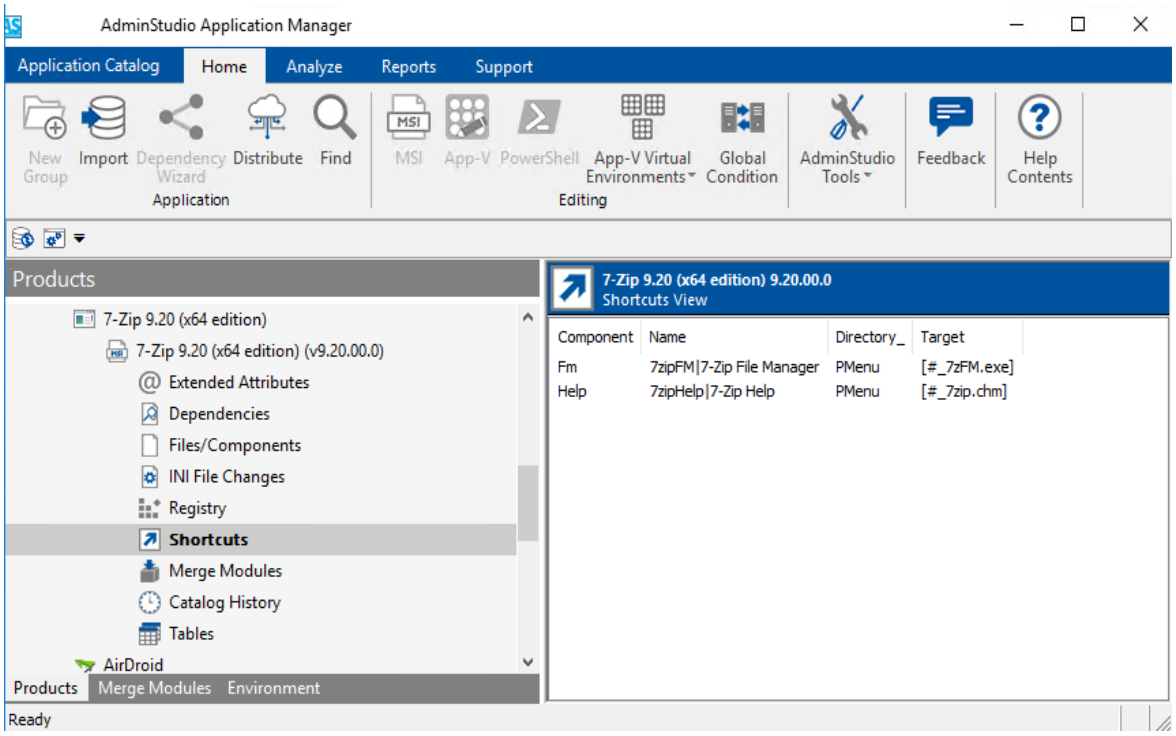
Column	Description
Component	Name of the component that the shortcut listed in the Name column is associated with.
Name	Name of the shortcut.
Directory_	Directory where the shortcut will exist.
Target	Directory and executable that the shortcut invokes.

Viewing App-V Package Shortcuts

The App-V **Shortcuts View** lists any shortcuts created by the App-V package. To view the App-V **Shortcuts View**, perform the following steps.

Task ***To view App-V package shortcuts:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. In the tree, expand an App-V package and select the **Shortcuts** node. The **Shortcuts View** opens.



Under the **Shortcuts** subheading, the following information is listed for each shortcut:

- Shortcut Name
- Target
- Arguments
- Working Directory
- Target Version
- Location

Under the **Shortcut Dependency** subheading, the following information is listed for each shortcut dependency:

- Shortcut Name
- Href
- GUID
- Is Mandatory

Under the **Shortcut Script** subheading, the following information is listed for each shortcut script:

- Shortcut Name
- Body (of script)

Important • *The **Shortcut Dependency** and **Shortcut Script** subheadings only apply to App-V 4.x packages.*

Viewing Windows Installer Package Merge Modules

To display any merge modules included with a Windows Installer package, expand the Windows Installer package in the Application Catalog tree and select the **Merge Modules** node.

Task ***To view a Windows Installer package's Merge Modules:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer package in the Application Catalog tree and select the **Merge Modules** node. The **Merge Modules View** opens.
3. Review the following information:

Column	Description
Title	The title of the Merge Module included with this package.
ModuleID	The number which uniquely identifies the Merge Module listed in the Title column.
Version	The version of the Merge Module listed in the Title column.
Language	The language that the Merge Module listed in the Title column was written for.

Viewing Package Catalog History

The tracking of change history is a critical operation within the Enterprise environment. Maintaining this information, displaying it, and allowing it to be a filter will gives you the information you need to monitor and maintain the integrity of your software packages.

In AdminStudio, any operation that materially changes a software package or the data associated with the package is tracked, and can be viewed in the Application Catalog **Catalog History** view.

Task ***To view a package's catalog history:***

1. Open Application Manager and select the **Home** tab of the ribbon.
2. In the tree, expand a Windows Installer or App-V package node and select the **Catalog History** node. The **Catalog History** view opens, displaying the following information:

Item	Description
Action	Name of the event which was logged: • Import/Reimport • Validation • Conflict Detection • Conflict Resolution • Extended Attribute Modification • Package Description Modification • Package Move/Copy • Patch Impact Analysis
Date	Date and time logged event occurred.
User	User who performed the logged event.
Description	Description providing details of the logged event.

Note • *If a package was replicated into another Application Catalog, its history data would not be replicated.*

Deleting Package History

To delete all of the entries in a package's History Log File, perform the following steps:

Task ***To delete package history:***

1. Open Application Manager.
2. Select the package and right-click to open the shortcut menu.
3. On the shortcut menu, point to **Delete** and click **History Log Information**.

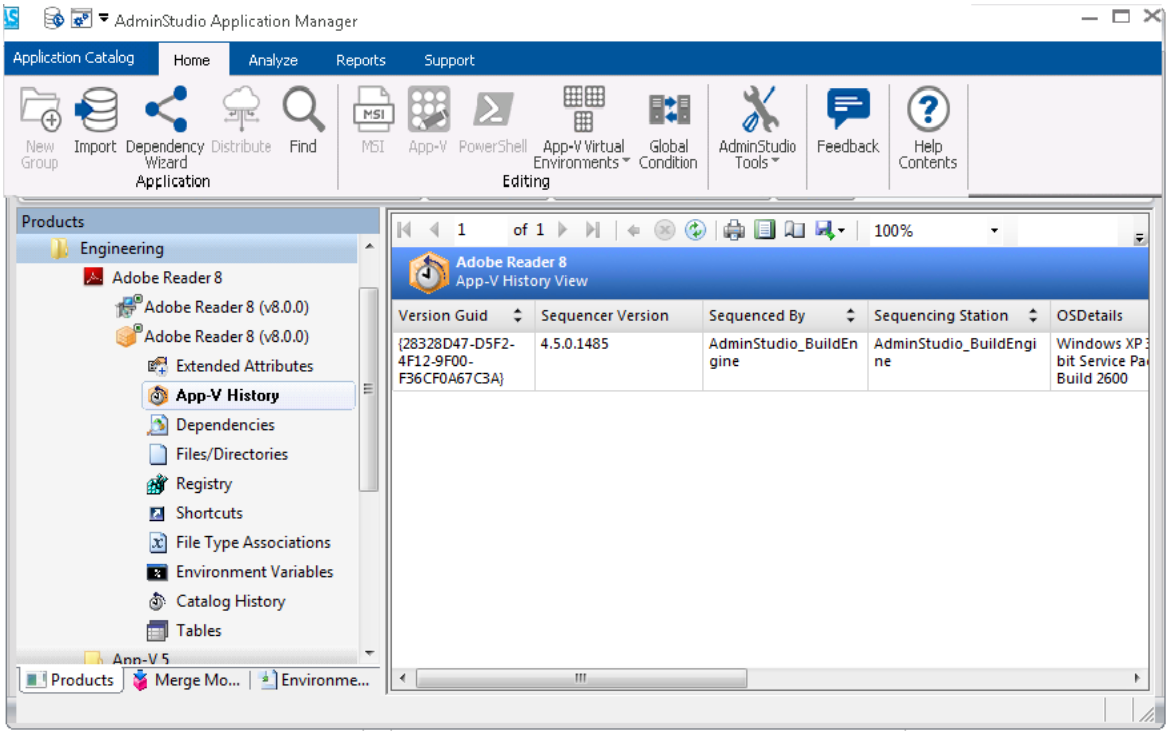
Viewing App-V Package History

Note • *This information applies to App-V 4.x packages.*

The **App-V History View** lists an entry for each time the App-V package has been saved. To view App-V package history, perform the following steps.

Task ***To view App-V package history:***

1. Open Application Manager and select the **Home** tab in the ribbon.
2. In the tree, expand an App-V package and select the **App-V History** node. The **App-V History View** opens, listing an entry for each time this App-V package has been saved.



For each entry, the following information is displayed:

- Version GUID
- Sequencer Version
- Sequenced By
- Sequencing Station
- OSDetails
- System Folder
- Windows Folder
- User Folder
- .Net Framework Version
- IEVersion

Viewing Package Tables

The **Tables View**, which opens in the Application Catalog **Home** tab when a package's **Tables** node is selected in the tree, provides a way to view table data for Windows Installer packages, macOS packages, App-V packages, and mobile apps in the Application Catalog.

Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the `ASCMPackageJavaSummary` table in the **Tables View**.

Note • *Most tables for Windows Installer packages are derived directly from standard MSI tables, as described in the Windows Installer SDK online help. When building your own ACE rules to use for conflict identification, it is important to understand the data available for packages so you can construct the necessary rule.*

To display table information for a package, expand the package in the Application Catalog tree and select the **Tables** node.

Task ***To view package tables:***

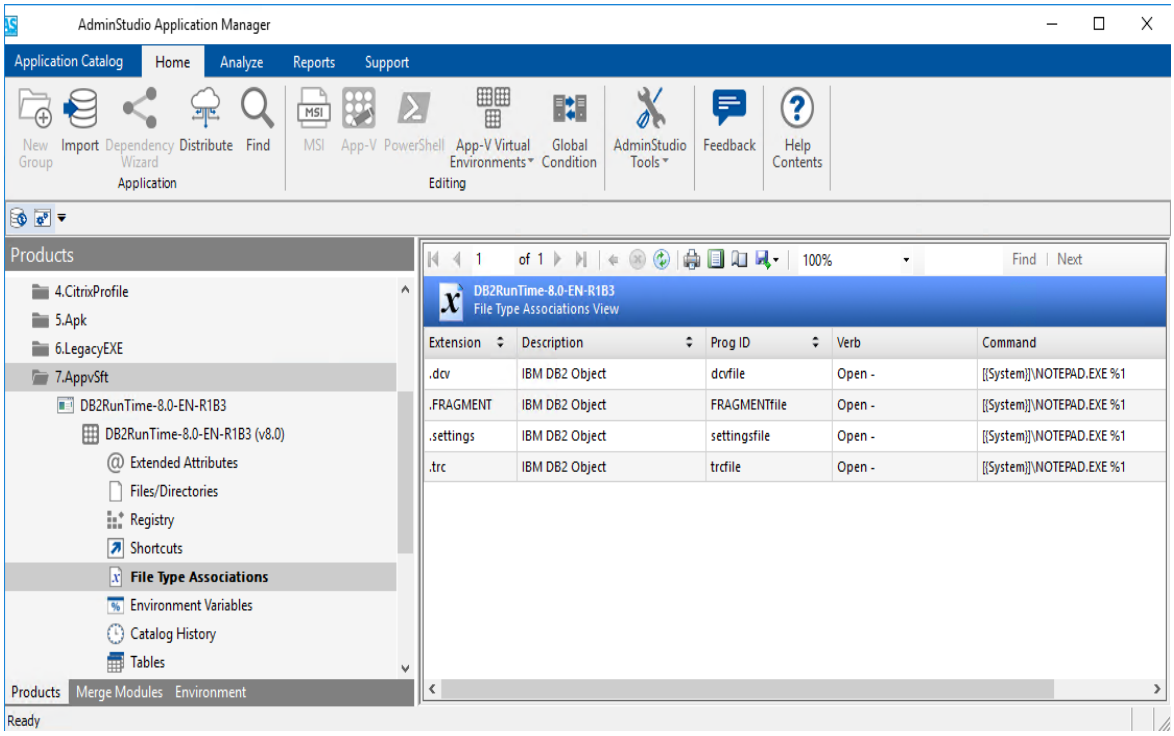
1. Open Application Manager and select the **Home** tab of the ribbon.
2. Expand a Windows Installer or App-V package in the Application Catalog tree and select the **Tables** node. The **Tables** view opens.
3. Select the specific table you want to view from the **Tables** list at the top of the view.

Viewing App-V Package File Type Associations

The App-V **File Type Associations View** lists an App-V package's file type associations. To view the App-V **File Type Associations View**, perform the following steps.

Task ***To view App-V package file type associations:***

1. Open Application Manager and select the **Home** tab in the ribbon.
2. In the tree, expand an App-V node and select the **File Type Associations** node. The **File Type Associations View** opens.



The following information is listed for each file type association:

- Extension
- Description
- Prog ID
- Verb
- Command

Viewing App-V Package Environment Variables

The App-V **Environment Variables View** lists an App-V package's environment variables. To view the App-V **Environment Variables View**, perform the following steps.

Task ***To view App-V package file type associations:***

1. Open Application Manager and open the **Home** tab in the ribbon.
2. In the tree, expand an App-V node and select the **Environment Variables** node. The **Environment Variables View** opens. The following information is listed for each variable:
   - Name
   - Value
   - h1

Creating, Importing, and Managing PowerShell-Wrapped Packages

AdminStudio provides support for PowerShell-wrapped packages built within adminstudio, including converting Windows Installer packages (.msi) and complex installation packages (.exe) to PowerShell wrapped packages (PowerShell script .ps1 files) using the PowerShell App Deployment Toolkit.

PowerShell is a task automation and configuration management framework from Microsoft. The PowerShell App Deployment Toolkit provides a set of functions to perform common application deployment tasks and to interact with the user during a deployment. By wrapping an existing Windows Installer or complex installation package in a PowerShell script, you are able to automatically perform deployment tasks both before and after installation.

AdminStudio also supports importing existing PowerShell-wrapped packages into the Application Catalog, testing a PowerShell-wrapped package's bundled child packages using Analyze, testing PowerShell-wrapped packages on a virtual machine, and distributing PowerShell wrapped packages using Distribution Wizard. You can also convert a PowerShell-wrapped package to a virtual package using the Conversion Wizard.

Information about AdminStudio's support for PowerShell-wrapped packages is organized in the following topics:

- Setting Wrap Options
- PowerShell Script Template Files
- Customizing the PowerShell Installer
- Wrapping a Windows Installer (.msi) or Installer Executable (.exe)
- Importing PowerShell Wrapped Packages
- Viewing a Power Shell Wrapped Package's Bundled Child Applications
- Testing PowerShell Wrapped Packages
- PowerShell Wrapped Packages Report
- Distributing PowerShell Wrapped Packages
- Virtualizing PowerShell Wrapped Packages
- Testing PowerShell Wrapped Packages on a Virtual Machine
- Editing a PowerShell-Wrapped Package
- AdminStudio PowerShell Cmdlet Support for Performing PowerShell Wrapping: Invoke-ASPowerShellWrap
- Updating PowerShell App Deploy Toolkit

Setting Wrap Options

On a new tab of the Application Manager Options dialog box, **Wrap Options**, you can select the required wrapped package, you can also specify the location of a template in the template directory and the output directory for the wrapped packages, and select the required option to wrap a package during import.

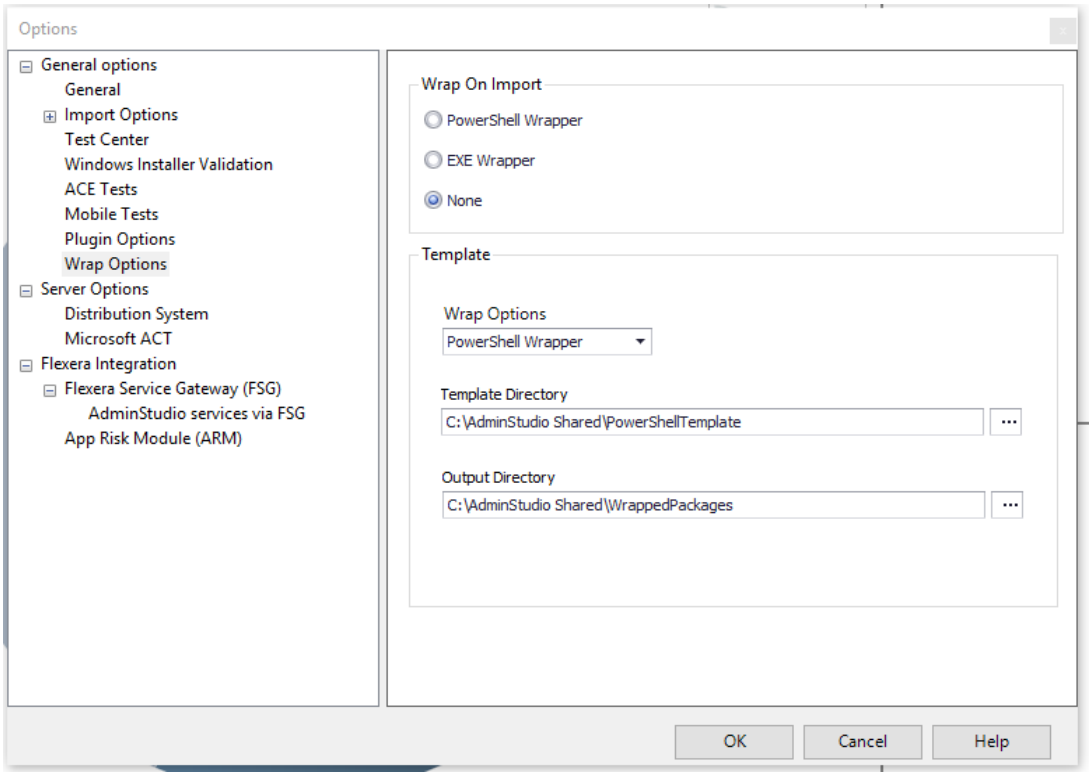


Figure 8: Wrap Options Tab of Options Dialog Box

The **Wrap Options** tab includes the following properties:

Option	Description

Template Directory	Specify the location of the template that you want to use to create the script files. A default template, `Deploy-Application.ps1`, along with other necessary files are provided in the following location:
	`For PowerShell - C:\AdminStudio Shared\PowerShellTemplate\`
	`For Exe Wrapper - C:\AdminStudio Shared\ExeTemplate\`

Output Directory	Specify the location where the wrapped packages that you create using AdminStudio will be stored. By default, the location is: `For PowerShell/Exe - C:\\AdminStudio Shared\WrappedPackages\` You can click the browse button and select a different directory.

Option	Description
Wrap on Import	Select this option if you want to automatically convert Windows Installer packages (`.msi`) and complex installation packages (`.exe`) to PowerShell wrapped packages (PowerShell script `.ps1` files) and Exe wrapped packages upon import into the Application Catalog. By default, this option is not selected.

PowerShell Script Template Files

A PowerShell script template file, `Deploy-Application.ps1`, is provided in the following location:

```
C:\AdminStudio Shared\PowerShellTemplate\
```

You can customize this script to meet your deployment requirements by editing it in the Windows PowerShell ISE application.

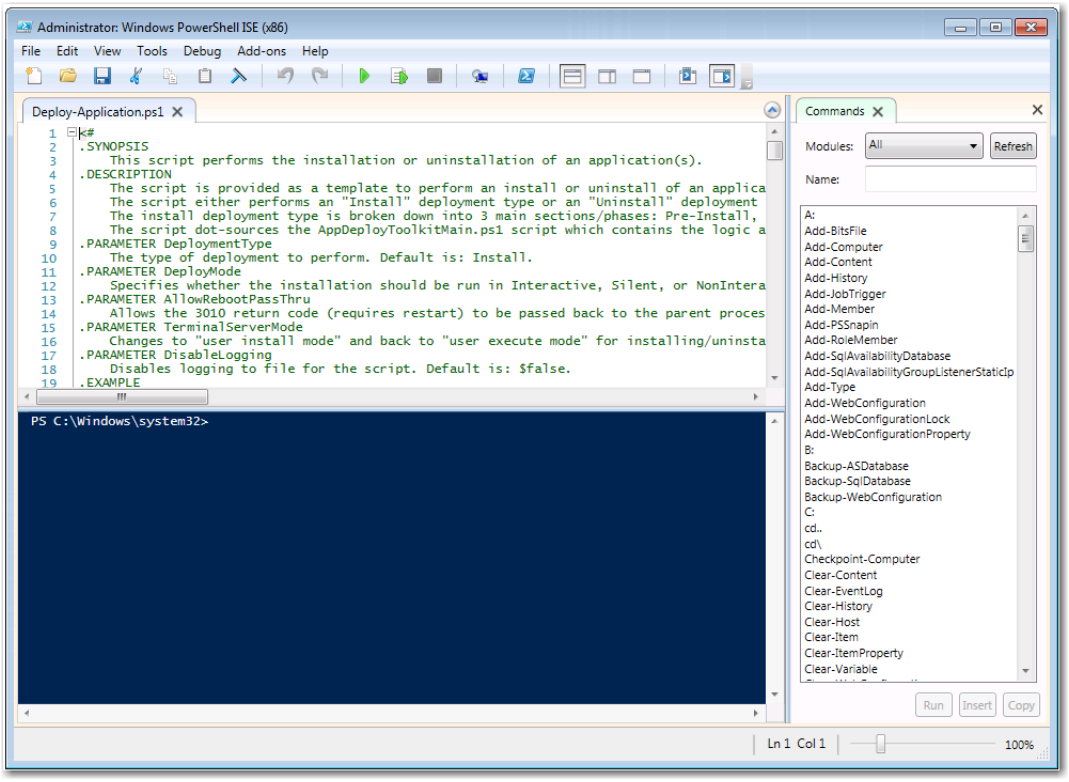


Figure 9: PowerShell Script Template File, Deploy-Application.ps1, in Windows PowerShell ISE

This script is provided as a template to perform an install or uninstall of an application(s). The script either performs an “install” deployment type or an “uninstall” deployment type. The install deployment type is broken down into three main sections: Pre-Install, Install, and Post-Install. The script refers to the `AppDeployToolkitMain.ps1` script which contains the logic and functions required to install or uninstall an application.

Contents of the PowerShellTemplate Directory

In addition to the `Deploy-Application.ps1` template file, the `PowerShellTemplate` directory also includes these additional files:

File	Description
`Deploy-Application.exe` `Deploy-Application.exe.config`	Executable that invokes a PowerShell script.
`Deploy-Application.ps1`	PowerShell script template file which is modified by the Wrap Package Wizard to contain the actual install logic for the specific package, including: • **Installer Path** The variable corresponding is $dirFiles variables (toolkit variable found under, AppDeployToolkit\AppDeployToolkitMain.ps1). • Name The variable corresponding is {appName} • Version The variable corresponding is {appVersion} • Manufacturer The variable corresponding is {appVendor} • Language The variable corresponding is {appLang} • Install Command Line The variable corresponding is {InstallPackage} • Uninstall Command Line The variable corresponding is {UnInstallPackage} For example, if Picasa.msi is the package to be wrapped, it's installer path would be: $dirFiles\Picasa.msi
`AppDeployToolkit\AppDeployToolkitBanner.png`	Banner image which can be customized to brand the PowerShell installer.
`AppDeployToolkit\AppDeployToolkitLogo.ico`	Image used when the PowerShell installer runs.
`AppDeployToolkit\AppDeployToolkitConfig.xml`	Contains configurable preferences for how the toolkit behaves, including language settings.
`AppDeployToolkit\AppDeployToolkitExtensions.ps1`	Script containing your extensions for the framework.

File	Description
AppDeployToolkit\AppDeployToolkitMain.ps1	Main script with all functions.
AppDeployToolkit\AppDeployToolkitHelp.ps1	Help file explaining the functions.
AppDeployToolkit\AppDeployToolkitMain.cs	Additional functions.

Rules for a Custom PowerShell Template

If you are creating a custom PowerShell template file, observe the following rules:

- **Do not rename Deploy-Application.ps1**—The `Deploy-Application.exe` application looks for a file named `Deploy-Application.ps1`. Therefore, if the `.ps1` file name is changed to the package name or anything else, the install/uninstallation will fail.
- **Deploy-Application.ps1 must be in the root folder**—The `Deploy-Application.exe` application expects the `Deploy-Application.ps1` file to be present in the root folder level, the same folder that contains the `Deploy-Application.exe` file.
- **Installer package must be in the Files folder**—The installer package (`.msi` or `.exe`) must be in the `Files` folder.
- **AppDeployToolkit must be in the root folder**—The `AppDeployToolkit`folder (along with all of its contents) must be in the root folder.

Note • *Installation/uninstallation is not obstructed if the `Deploy-Application.exe.config` file is not present in the parent folder level or is even deleted.*

Note • *The `$dirFiles` variable includes the path to the installer package. If you change the value of this variable, installation will fail.*

Customizing the PowerShell Installer

You can customize the banner image that is displayed in the PowerShell installer and the post-installation messages that are displayed.

- Customizing the PowerShell Installer Banner Image
- Customizing Post-Installation Text Messages

Customizing the PowerShell Installer Banner Image

When AdminStudio converts a package to a PowerShell-wrapped package, the following image file is displayed at a banner at the top of the installer:

`\AdminStudio Shared\PowerShellTemplate\AppDeployToolkit\AppDeployToolkitBanner.png`

This image, by default, is the PowerShell App Deployment Toolkit logo.

Figure 10: PowerShell Installer Default Banner

You can customize this banner by replacing the content of the `AppDeployToolkitBanner.png` file.

In the following image, the PowerShell Installer default banner was replaced by a customized banner.

Figure 11: PowerShell Installer with Customized Banner

Customizing Post-Installation Text Messages

You can customize the post-installation messages that are displayed in the PowerShell installer by editing the `Show-InstallationPrompt -Message` value in the `Deploy-Application.ps1` file. By default, the message is: `Thank you for using Flexera!`.

```
##*=============================================
##* POST-INSTALLATION
##*=============================================
[string]$installPhase = 'Post-Installation'

## <Perform Post-Installation tasks here>

## Display a message at the end of the install
If (-not $useDefaultMsi) { Show-InstallationPrompt -Message 'Thank you for using Flexera!' -
ButtonRightText 'OK' -Icon Information -NoWait }
```

Wrapping a Windows Installer (.msi) or Installer Executable (.exe)

You can convert a Windows Installer (`.msi`) or complex installation package (`.exe`) to a PowerShell wrapped package (PowerShell script `.ps1` file) by selecting it in the Application Catalog tree and selecting **Wrap package**.

By wrapping an existing Windows Installer or complex installation package in a PowerShell script, you are able to launch additional deployment tasks via the PowerShell script both before and after installation. The advantages of using wrapping instead of repackaging or transforms to modify an installer include:

- **Package is not modified**—The original package is not modified.

- **Consistent UI experience**—You can provide a consistent UI experience to your users for all of your organization's installers.
- **Powerful PowerShell functionality**—The PowerShell scripting language provides a more powerful set of functionality than can be provided using transforms.

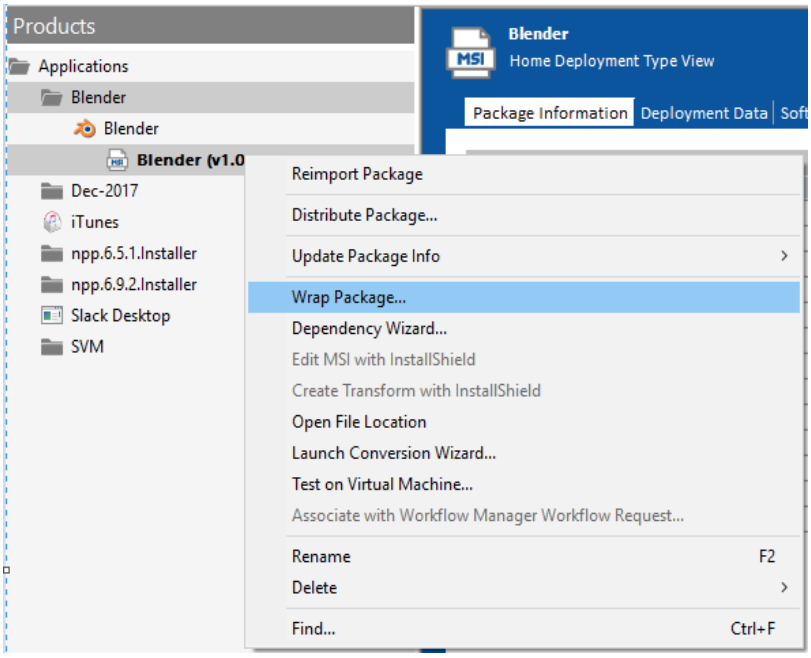
You can convert the following deployment types into a wrapped package:

- Windows Installer packages (`.msi`)
- InstallShield InstallScript `.exe` files
- InstallShield Basic MSI installers that are compressed into a `setup.exe` file
- InstallShield Suite Installer `.exe` files
- Wise Package Studio `.exe` files
- Other executable file types that can be uncompressed by 7-ZIP

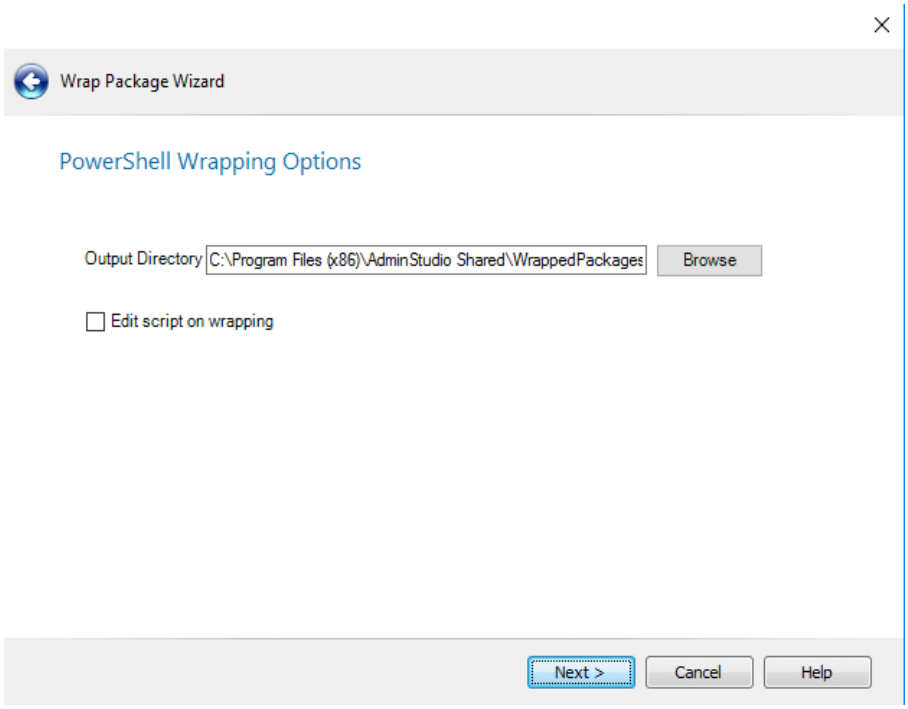
To convert a Windows Installer (`.msi`) or installation package (`.exe`) to a wrapped package, perform the following steps.

Task ***To convert a Windows Installer (.msi) or installation package (.exe) to a wrapped package:***

5. Select the Windows Installer (`.msi`) or installation package (`.exe`) package in the Application Catalog tree and select **Wrap Package** from the context menu.



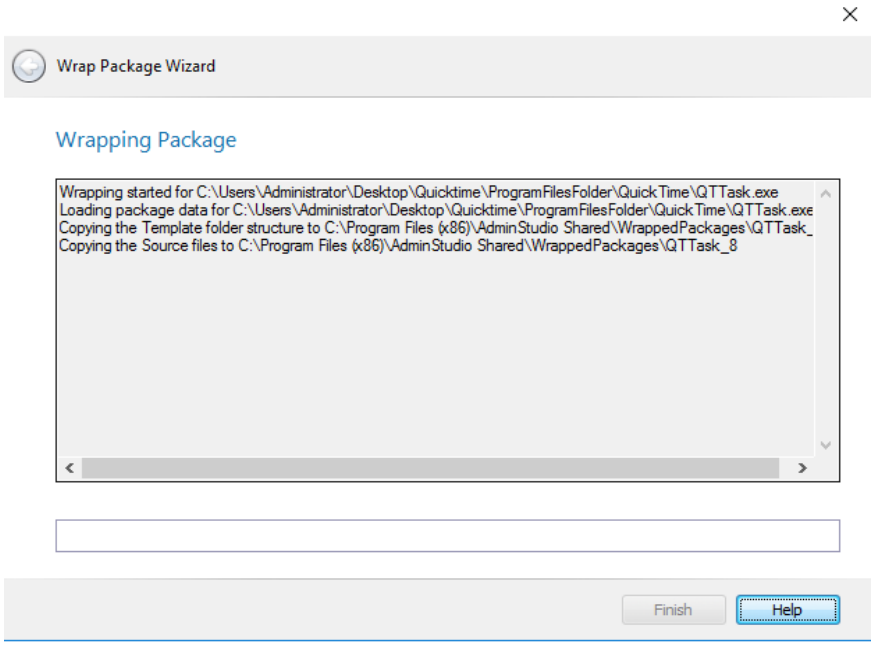
The **Wrap Package** opens.



3. Specify the **Output Directory**, the location where the PowerShell-wrapped package will be stored.

For more information on wrapping msi/exe packages to Wise script wrapped exes, see Wrap MSI/EXE Packages Using the Wrap Package Wizard.

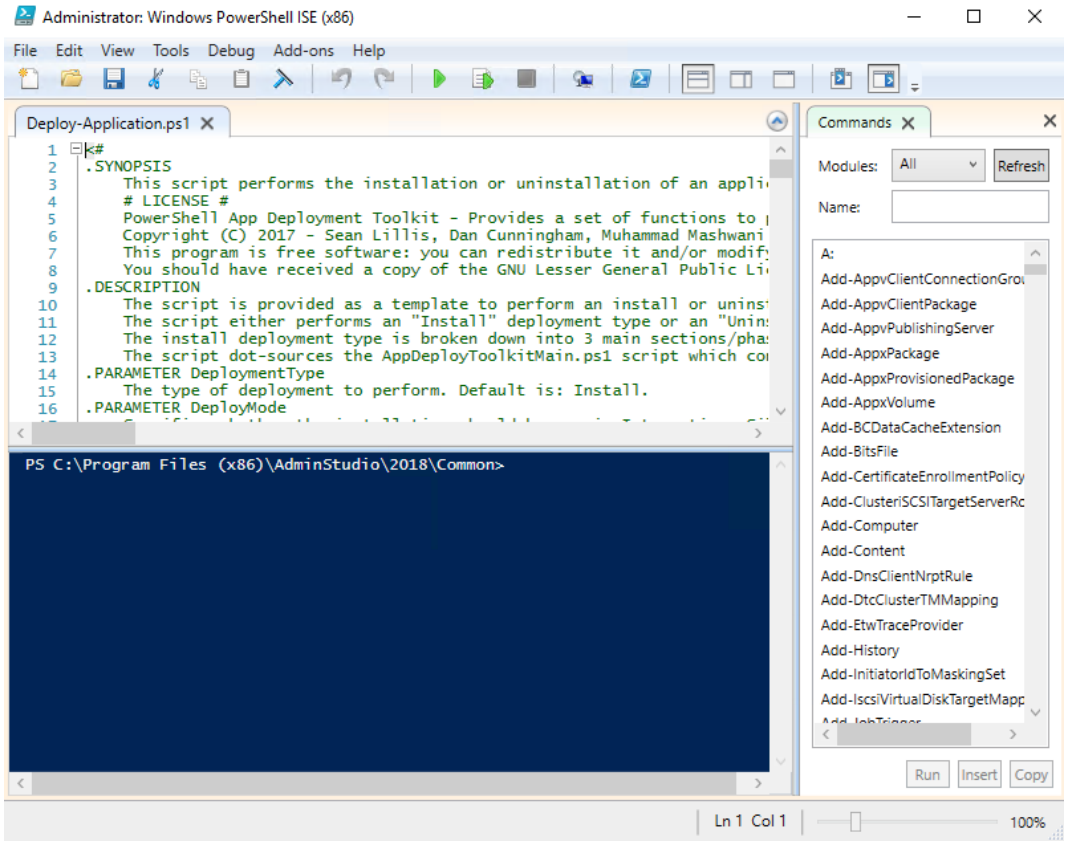
4. Click **Next**. The **Wrapping Package** panel opens and wrapping is initiated. When the process is complete, confirmation messages are listed.



5. Click **Finish** to close the wizard.

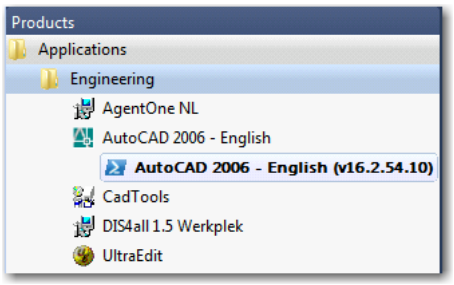
6. If you want to edit the default PowerShell script file in Windows PowerShell ISE after the package has been wrapped, select **Edit script on wrapping** option in the Application Catalog tree.

   If you selected the **Edit script on wrapping** option, the Windows PowerShell ISE application opens and you are prompted to edit the PowerShell script file that you just created.



7. Make any desired changes and save the script file.

8. In the Application Catalog tree, view the package you just wrapped. You will see that the `.msi` or `.exe` deployment type icon for this package has been replaced with a PowerShell deployment type icon.



Important • *All of the OS compatibility, application virtualization compatibility and best practices tests results that were generated by testing the package before it was wrapped are carried over and are viewable on* ***Analyze*** *tab when you select its new PowerShell deployment type icon.*

Note • *Log files for the Wrap Package WizardWrap Package Wizard can be found at:* `C:\Windows\Logs\Software`.

Importing PowerShell Wrapped Packages

You can import existing PowerShell wrapped packages into the Application Catalog. To do so select the **PowerShell wrapped package (.ps1)** option on the **Package Type Selection** panel of the **Import Wizard**.

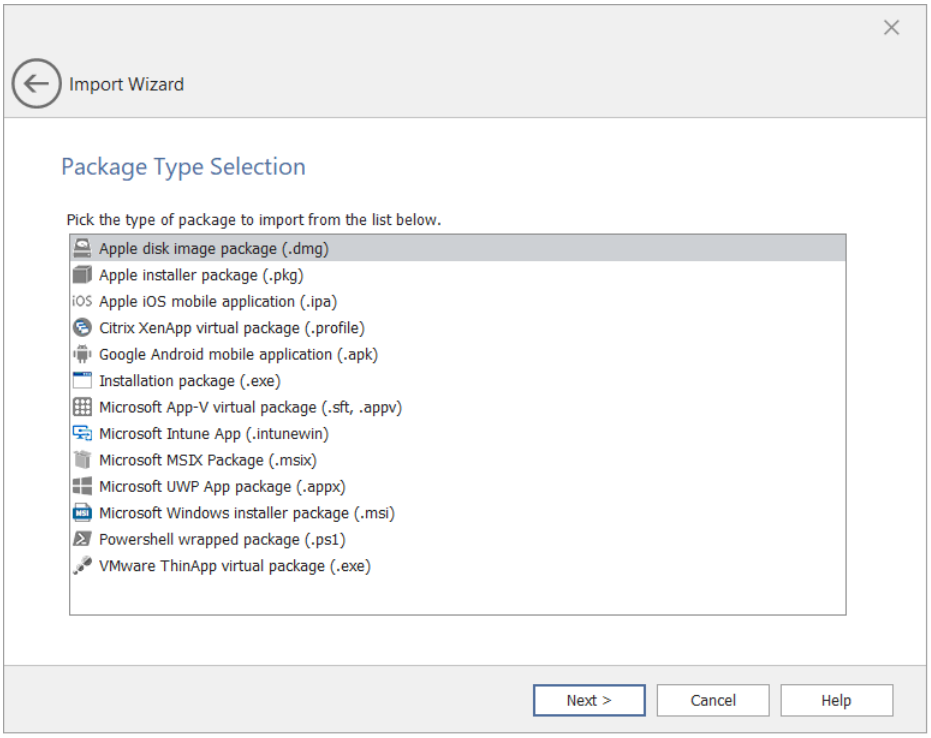


Figure 12: PowerShell Wrapped Package (.ps1) Option on Package Type Selection Panel of Import Wizard

For instructions on importing a PowerShell-wrapped package into the Application Catalog, see the following topics:

- Importing a Single Package File
- Importing a Folder of Multiple Applications

Note • *Also, if the* ***Wrap on Import*** *option is selected on the* ***PowerShell Wrap Options*** *tab of the* ***Options*** *dialog box, Windows Installer packages (*`.msi`*) and installation packages (*`.exe`*) packages will automatically be wrapped (converted to PowerShell script files) during import. For more information, see Setting Wrap Options.*

Viewing a Power Shell Wrapped Package's Bundled Child Applications

You can view the child packages that are bundled within a PowerShell script file on the **Bundled Packages** tab of the **Home Deployment Type** view.

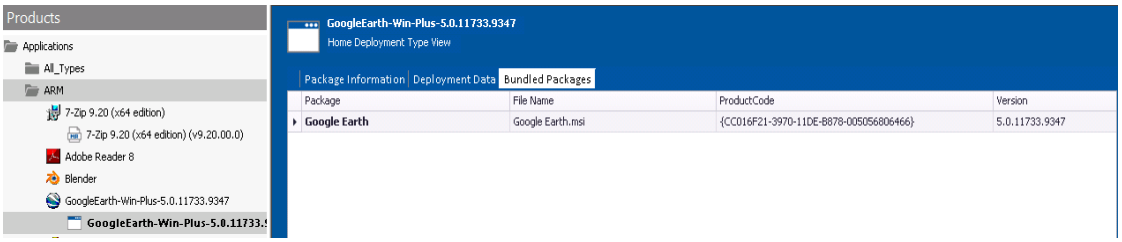


Figure 13: Displaying Child Packages of PowerShell Script File on Bundled Packages Tab

For more information, see Viewing Bundled Packages of Complex Installer Executables.

Testing PowerShell Wrapped Packages

Using Analyze, you can perform OS compatibility, application virtualization compatibility, and best practices testing of the child `.msi` and `.exe` packages that are bundled in a PowerShell script file.

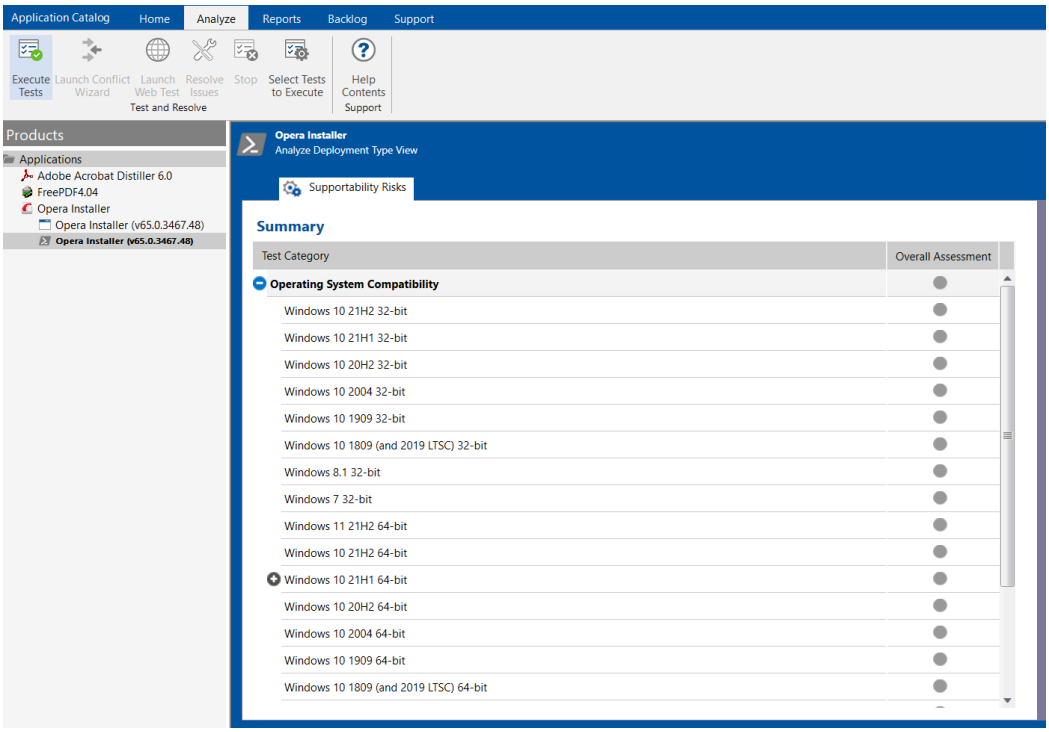


Figure 14: Test Results for a PowerShell Wrapped Package

To test a PowerShell-wrapped package, open the **Analyze** tab, select the application or package in the Application Catalog tree, and click **Execute Tests**.

For more information, see Performing Compatibility, Best Practices, and Risk Assessment Testing and Viewing and Filtering Test Results.

PowerShell Wrapped Packages Report

Reports includes a **PowerShell Wrapped Packages** report which lists details on the PowerShell-wrapped packages in your Application Catalog.

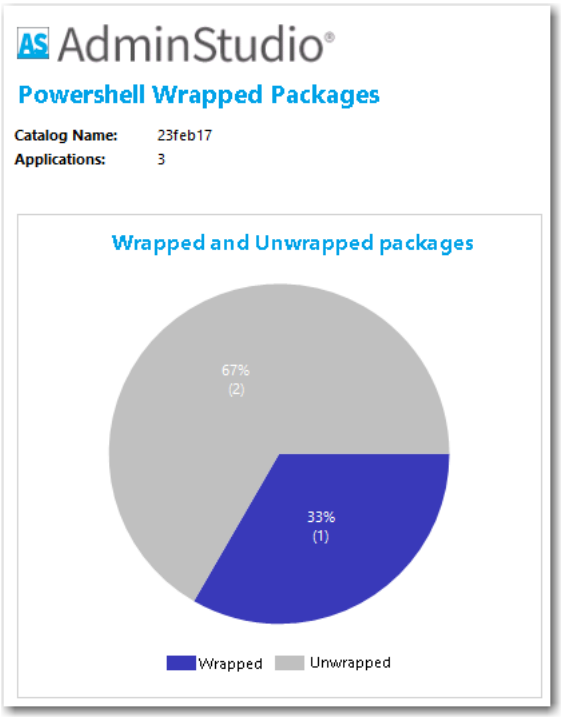


Figure 15: PowerShell Wrapped Packages Summary Report

When you click on one of the sections of the pie chart, a detailed report opens listing packages in that category.

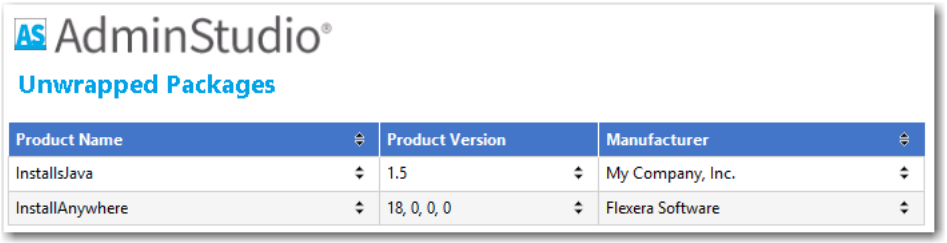


Figure 16: PowerShell Wrapped Package Detail Report

Distributing PowerShell Wrapped Packages

You can distribute PowerShell-wrapped packages to ConfigMgr (Formerly called as System Center Configuration Manager) and Symantec Altiris Client Management Suite using the Distribution Wizard.

Select the application containing the PowerShell-wrapped package deployment type, and then click the **Distribute** button in the toolbar to open the Distribution Wizard, and proceed with distribution.

For more information, see Distributing Applications Using the Distribution Wizard.

Virtualizing PowerShell Wrapped Packages

You can convert a PowerShell-wrapped package to a virtual package using the Conversion Wizard.

For more information, see Using the Conversion Wizard to Perform Virtualization or Repackaging.

Testing PowerShell Wrapped Packages on a Virtual Machine

You can test the installation of a PowerShell wrapped package on a virtual machine using the **Test on Virtual Machine Wizard**. This wizard uses the capability of the Automated Application Converter tool to spin up the selected virtual machine and install the selected package.

For more information, see Using Test on Virtual Machine Wizard.

Editing a PowerShell-Wrapped Package

You can edit the PowerShell-wrapped package script (`.ps1` file) using the Windows PowerShell ISE application. If you selected the **Edit script on wrapping** option on the **PowerShell Wrapping Options** panel of the **Wrap Package Wizard**, the script file opens in the Windows PowerShell ISE application immediately after you close the wizard.

You can also edit the `.ps1` file by right-clicking on the PowerShell-wrapped package node in the Application Catalog tree, and then clicking **Edit Package** on the menu.

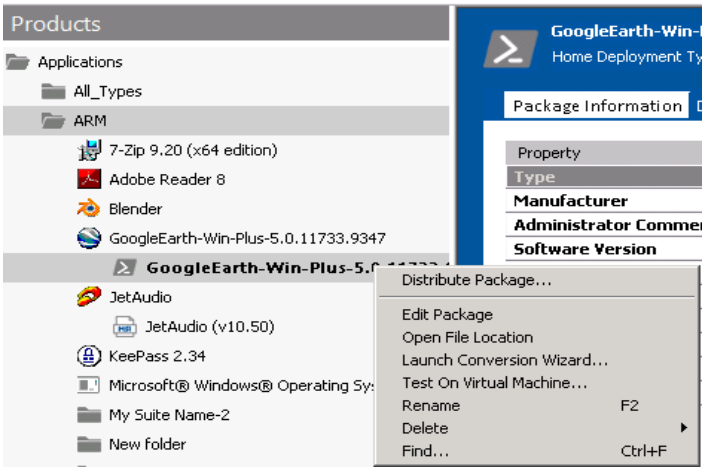


Figure 17: Editing a PowerShell-Wrapped Package

You can also launch the PowerShell editor by clicking the **Edit with PowerShell Editor** button in the ribbon.

Figure 18: Edit with PowerShell Editor Button

Windows PowerShell ISE application opens, where you can edit the PowerShell script file.

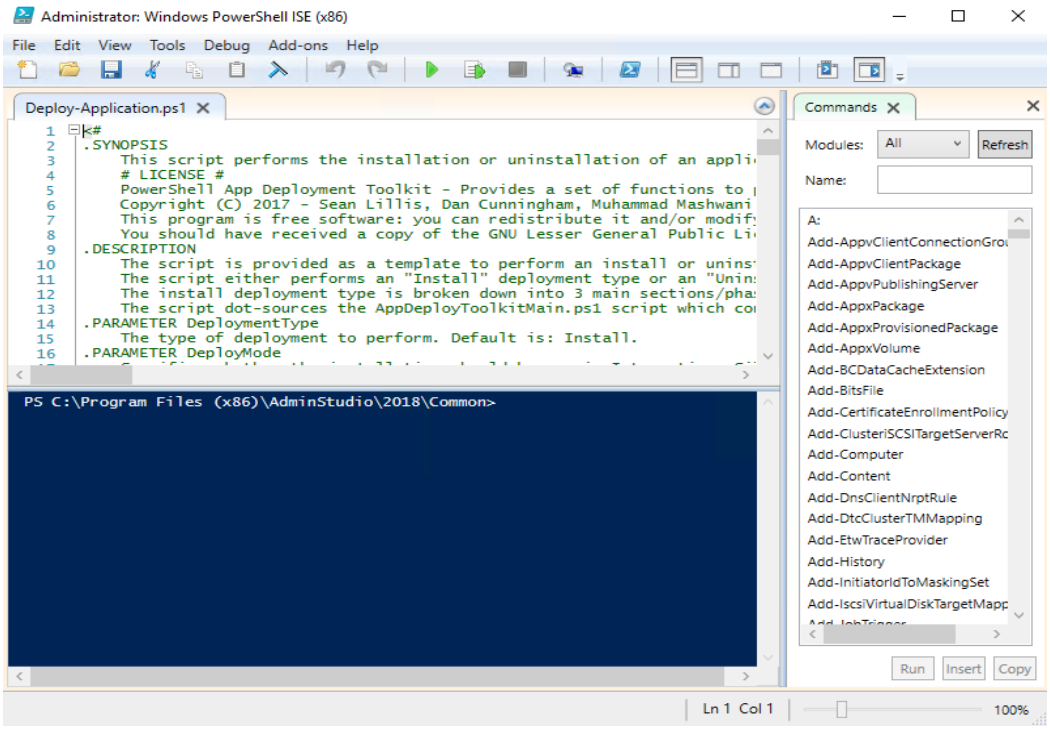


Figure 19: Editing a PowerShell Script File in Windows PowerShell ISE

AdminStudio PowerShell Cmdlet Support for Performing PowerShell Wrapping: Invoke-ASPowerShellWrap

You can use a AdminStudio PowerShell Cmdlet command to perform PowerShell wrapping. You can use the `Invoke-ASPowerShellWrap` command to convert a Windows Installer (`.msi`) or installation package (`.exe`) to a PowerShell wrapped package file (`.ps1`).

The following is an example of the `Invoke-ASPowerShellWrap` command:

```
Invoke-ASPowerShellWrap -PackageId 25 -OutputDirectory "D\WrappedPackagesFeb"
```

For more information, see Invoke-ASWrapPackage.

Updating PowerShell App Deploy Toolkit

The PowerShell App Deployment Toolkit provides a set of functions to perform common application deployment tasks and to interact with the user during a deployment. It simplifies the complex scripting challenges of deploying applications in the enterprise, provides a consistent deployment experience and improves installation success rates.

To update PowerShell App Deploy Toolkit, perform the following steps:

Task ***To update PowerShell App Deploy Toolkit:***

1. Download the desired PowerShell Toolkit version from https://psappdeploytoolkit.com and extract it. The extracted file opens.
2. From the extracted file, open **Deploy-Application.ps1** file, and modify it as mentioned below:
   a. Update variable declaration values by adding the below code:

```
[string]$appVendor = '{appVendor}'
[string]$appName = '{appName}'
[string]$appVersion = '{appVersion}'
[string]$appArch = '{appArch}'
[string]$appLang = '{appLang}'
[string]$appScriptDate = '{appScriptDate}'
```

```
##*=============================================
##* VARIABLE DECLARATION
##*=============================================
## Variables: Application
[string]$appVendor = ''
[string]$appName = ''
[string]$appVersion = ''
[string]$appArch = ''
[string]$appLang = 'EN'
[string]$appRevision = '01'
[string]$appScriptVersion = '1.0.0'
[string]$appScriptDate = 'XX/XX/20XX'
[string]$appScriptAuthor = '<author name>'
##*=============================================
```

```
##*=============================================
##* VARIABLE DECLARATION
##*=============================================
## Variables: Application
[string]$appVendor = '{appVendor}'
[string]$appName = '{appName}'
[string]$appVersion = '{appVersion}'
[string]$appArch = '{appArch}'
[string]$appLang = '{appLang}'
[string]$appRevision = '01'
[string]$appScriptVersion = '1.0.0'
[string]$appScriptDate = '{appScriptDate}'
[string]$appScriptAuthor = '<author name>'
##*=============================================
```

   b. Update the installation values by adding the below code:

```
{InstallPackage}
{InstallPatch}
```

```
##*=============================================
##* INSTALLATION
##*=============================================
[string]$installPhase = 'Installation'

## Handle Zero-Config MSI Installations
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Action
    Execute-MSI @ExecuteDefaultMSISplat; If ($defaul
}

## <Perform Installation tasks here>
```

```
##*=============================================
##* INSTALLATION
##*=============================================
[string]$installPhase = 'Installation'

## Handle Zero-Config MSI Installations
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Action
    Execute-MSI @ExecuteDefaultMSISplat; If ($defau
}

## <Perform Installation tasks here>

{InstallPackage}
{InstallPatch}
```

   c. Update UnInstallation values by adding the below code

```
{UnInstallPackage}
```

```
##*=============================================
##* UNINSTALLATION
##*=============================================
[string]$installPhase = 'Uninstallation'

## Handle Zero-Config MSI Uninstallations
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Action
    Execute-MSI @ExecuteDefaultMSISplat
}

# <Perform Uninstallation tasks here>
```

```
##*=============================================
##* UNINSTALLATION
##*=============================================
[string]$installPhase = 'Uninstallation'

## Handle Zero-Config MSI Uninstallations
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Action
    Execute-MSI @ExecuteDefaultMSISplat
}

# <Perform Uninstallation tasks here>

{UnInstallPackage}
```

d. Update the repair values by adding the below code

{RepairPackage}

```
##*=============================================
##* REPAIR
##*=============================================
[string]$installPhase = 'Repair'

## Handle Zero-Config MSI Repairs
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Actior
Execute-MSI @ExecuteDefaultMSISplat
}
# <Perform Repair tasks here>
```

```
##*=============================================
##* REPAIR
##*=============================================
[string]$installPhase = 'Repair'

## Handle Zero-Config MSI Repairs
If ($useDefaultMsi) {
    [hashtable]$ExecuteDefaultMSISplat =  @{ Action
Execute-MSI @ExecuteDefaultMSISplat
}
# <Perform Repair tasks here>

{RepairPackage}
```

Note • *Repair action type support is available from v3.8.1.*

3. After modifying, change the Template Directory for Wrap Options (Application Manager > Options >Wrap Options) to the folder containing modified Deploy-Application.ps1 file.

Using the Conversion Wizard to Perform Virtualization or Repackaging

Edition • *The Conversion Wizard is included with AdminStudio Professional and Enterprise Editions, and you can use it to perform automated repackaging on a virtual machine.*

You can use the Application Catalog **Conversion Wizard** to perform the following tasks from within Application Manager:

- Convert an App-V 4.x package to App-V 5.0 format
- Convert one or multiple Windows Installer packages, legacy installers, or PowerShell-wrapped packages to virtual packages using either default or customized Automated Application Converter settings
- Perform automated repackaging of one or multiple Windows Installer packages, legacy installers, or PowerShell-wrapped packages on a virtual machine.
- Convert an MSI or EXE package to Microsoft Intune format.

Information about using the Conversion Wizard is presented in the following topics:

- Setting Conversion Wizard Options

- Converting App-V 4.x Packages to App-V 5.0 Format
- Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging
- Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard

Note • *Repackager now support Windows Services for MSIX Packages. A Windows Service installed while converting a legacy package format (MSI/EXE) will be captured and packaged into the MSIX package upon conversion.*

Setting Conversion Wizard Options

Edition •

Before you can perform conversions using the Conversion Wizard, you must first create an Automated Application Converter settings file and set additional default options.

- Creating an Automated Application Converter Settings File
- Specifying the Default Automated Application Converter Settings File
- Setting App-V 5.0 Conversion Options

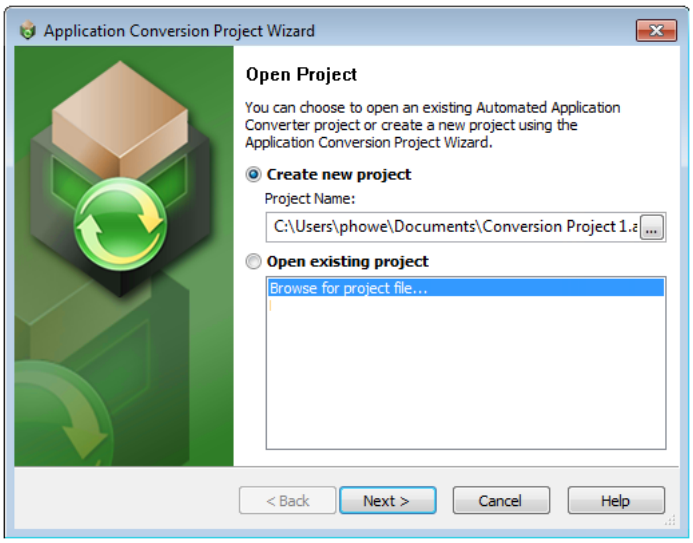
Creating an Automated Application Converter Settings File

In order to use the Conversion Wizard to perform an express conversion of one or multiple Windows Installer packages, PowerShell-wrapped packages, or legacy installers to virtual packages, you must first create an Automated Application Converter settings file that contains virtual machine login information and conversion defaults.

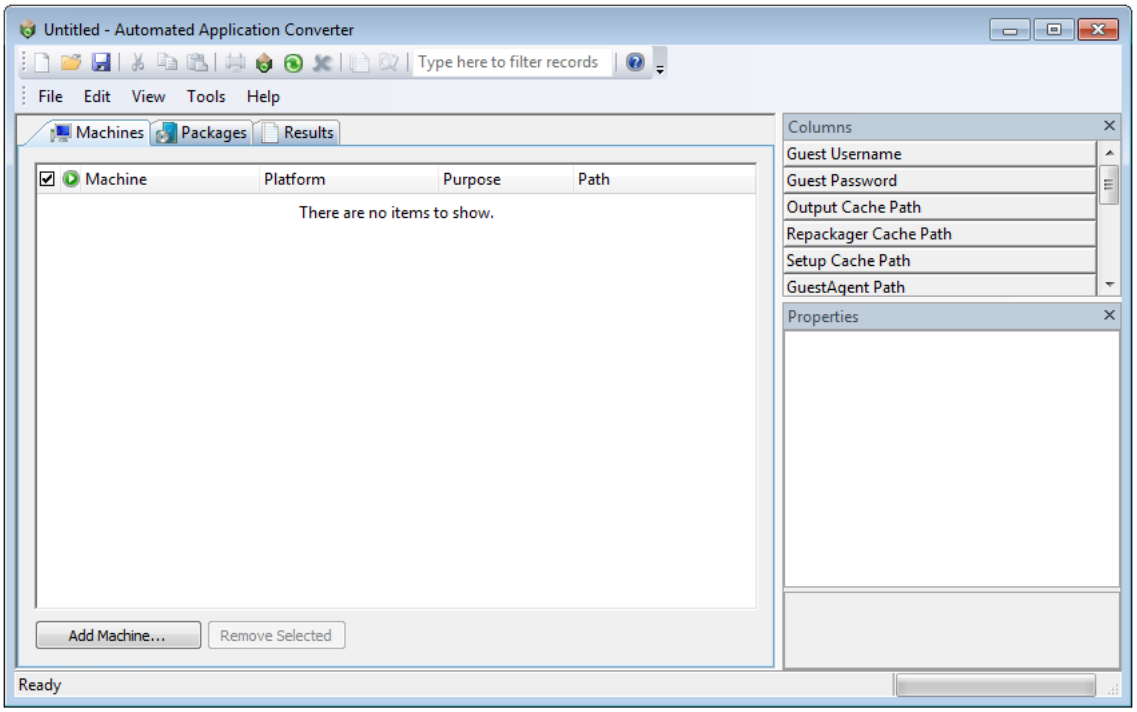
To create an Automated Application Converter settings file, perform the following steps.

Task ***To create an Automated Application Converter settings file:***

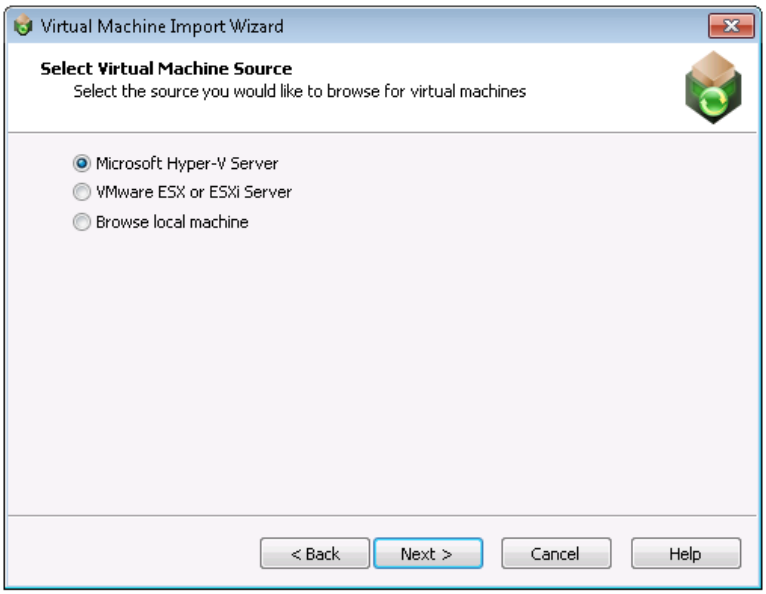
1. Launch Automated Application Converter. The **Open Project** dialog box opens.



2. Click **Cancel**. An untitled project is opened in Automated Application Converter.



3. On the **Machines** tab click **Add Machine**. The **Virtual Machine Import Wizard Welcome** panel opens.
4. Click **Next**. The **Select Virtual Machine Source** panel opens, prompting you to select the type of virtual machine that you are adding.

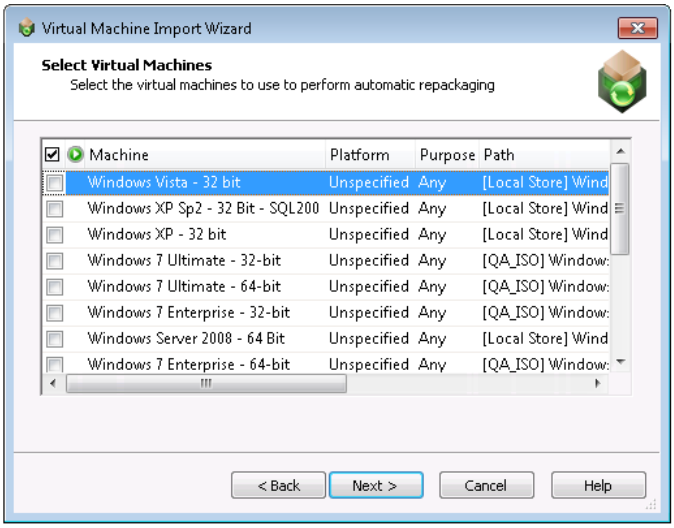


5. Select one of the following options and click **Next**.
   - **Microsoft Hyper-V Server**—Select this option to add a virtual image from a Microsoft Hyper-V Server.
   - **VMware ESX or ESXi Server**—Select this option to add a virtual image from a VMware ESX or ESXi Server.
   - **Browse local machine**—Select this option to add a virtual image from a local installation of VMware Workstation
6. Based upon your selection on the **Select Virtual Machine Source** panel, enter the following information:

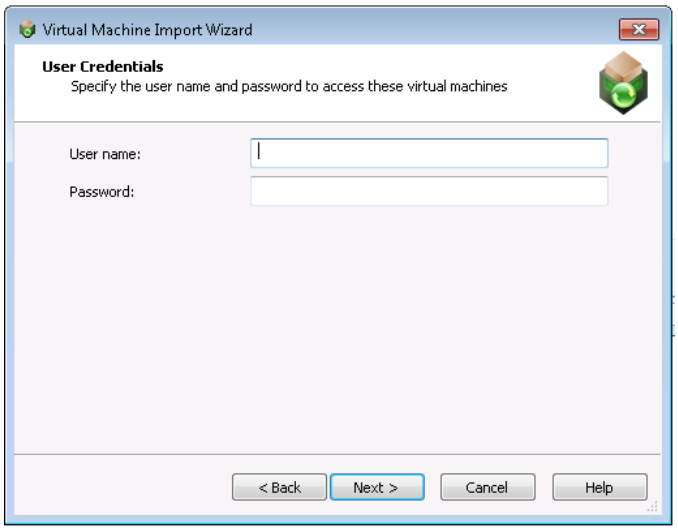
Virtual Machine Source	Steps to Take
Microsoft Hyper-V Server	On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, enter the following information: • **Server Name**—Enter the server name of the Microsoft Hyper-V Server that you want to connect to. • **Authentication**—Select **Windows Authentication** if you want to use the credentials of the logged in user to login to the Hyper-V Server. Select **Server Authentication** if you want to connect to the Hyper-V Server using the specified **User name** and **Password**.
VMware ESX or ESXi Server	On the **Select Virtual Machines from VMware ESX or ESXi Server** panel, enter the following information: • **Server Name**—Enter the name of the VMware ESX or ESXi server. • **User name**—Enter the login ID for the VMware ESX or ESXi server. • **Password**—Enter the password for the VMware ESX or ESXi server.

Virtual Machine Source	Steps to Take
Browse local machine	On the **Select Virtual Machines** panel, do one of the following: ***To add an individual virtual machine:*** **1.** Click **Browse Files**. The **Select Virtual Machine Image File** dialog box opens. **2.** Select the virtual machine image you want to add to the project and click **Open**. ***To add all of the virtual machines in a specific directory:*** **1.** Click **Browse Folders**. The **Browse for Folder** dialog box opens. **2.** Select a directory that contains the virtual machine images that you want to add to your project and click **OK**.

When you have finished this step, the virtual machines will be listed (but not selected) on the **Select Virtual Machines** panel.



7. On the **Select Virtual Machines** panel, select the virtual machine images that you want to use to perform automated repackaging.
8. For each selected image, click in the **Platform** column and identify its platform.
9. Click **Next**. The **User Credentials** panel opens, prompting you to specify the login credentials to use to access the selected virtual machines.



10. Enter the user credentials and click **Next**. The **Virtual Machine Import Wizard Complete** panel opens.
11. Click **Finish** to close the wizard and add the selected virtual machines to your project.
12. On the **File** menu, click **Save** and enter a name for this project file.

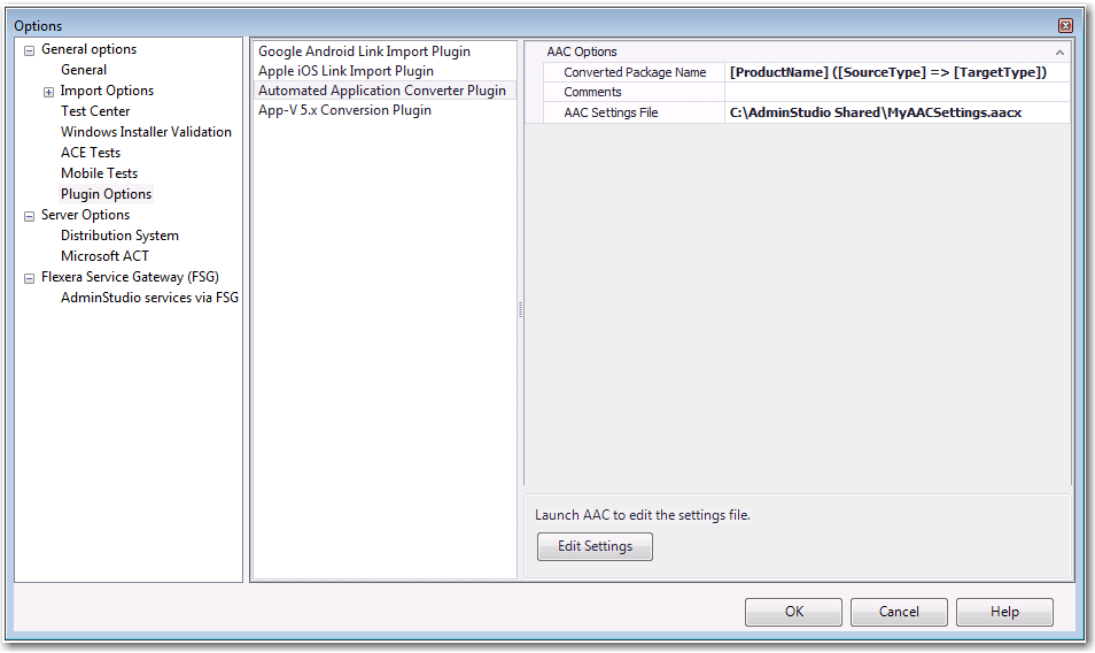
Specifying the Default Automated Application Converter Settings File

Edition •

After you create an Automated Application Converter settings file, as described in Creating an Automated Application Converter Settings File, perform the following steps to specify the default Automated Application Converter settings file.

Task ***To set Automated Application Converter options:***

1. In Application Manager, select **Options** on the Application Catalog menu. The **Options** dialog box opens.
2. Open the **General options > Plugin Options** tab.
3. In the middle pane, select **Automated Application Converter Plugin**. The Automated Application Converter options are displayed in the right pane.



4. In the **Converted Package Name** field, enter a name to differentiate the converted version of the package from the original version. By default, this field will be populated with the original package name `[ProductName]`. For example:
   - `[ProductName]`
   - `[Manufacturer]_[ProductName]`
   - `[ProductName]_v5`
5. In the **Comments** field, enter metadata that you would like to add to each converted package. This text will be displayed in the **Administrator Comments** field on the **Package Information** tab of the **Home Deployment Type View** for each package.
6. In the **AAC Settings File** field, browse to the Automated Application Converter settings file you created in Creating an Automated Application Converter Settings File.
7. Click **OK**.

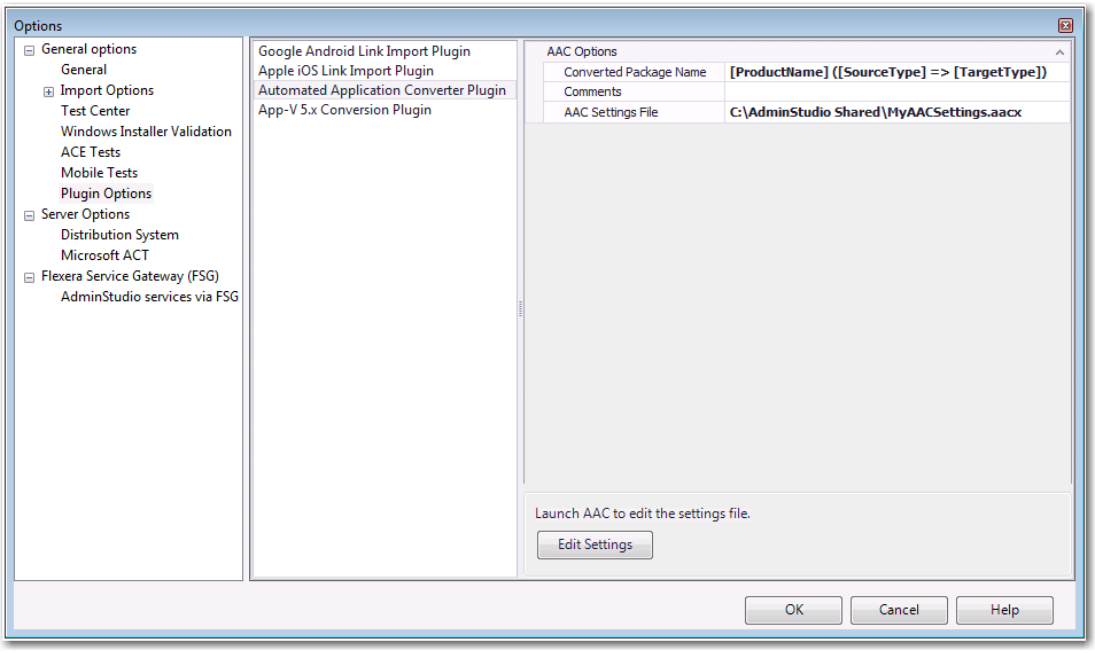
Editing the Default Automated Application Converter Settings File From Application Catalog

You can easily modify the settings in the default Automated Application Converter Settings file—which is specified on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box—by clicking the **Edit Settings** button.

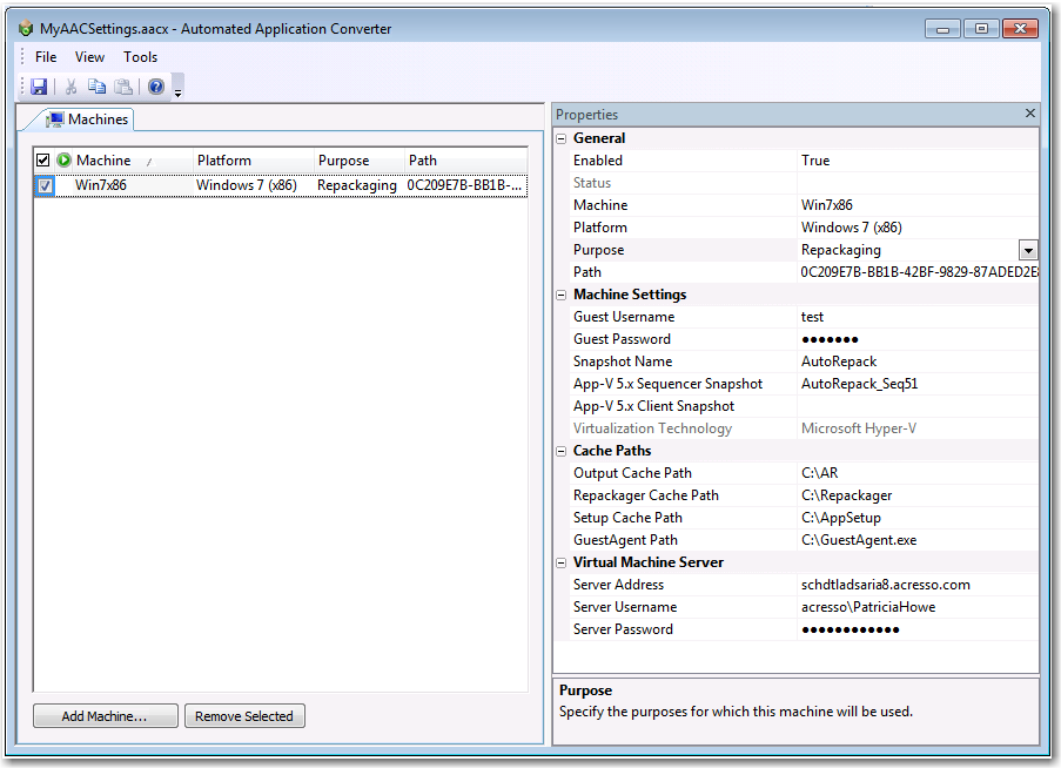
Task ***To edit the default Automated Application Converter settings file from Application Catalog:***

1. In Application Manager, select **Options** on the Application Catalog menu. The **Options** dialog box opens.
2. Open the **General options > Plugin Options** tab.

3. In the middle pane, select **Automated Application Converter Plugin**. The Automated Application Converter options are displayed in the right pane.



4. Click **Edit Settings**. A limited version of Automated Application Converter opens, displaying the **Machines** tab.



5. Edit any of these default conversion settings, as desired.

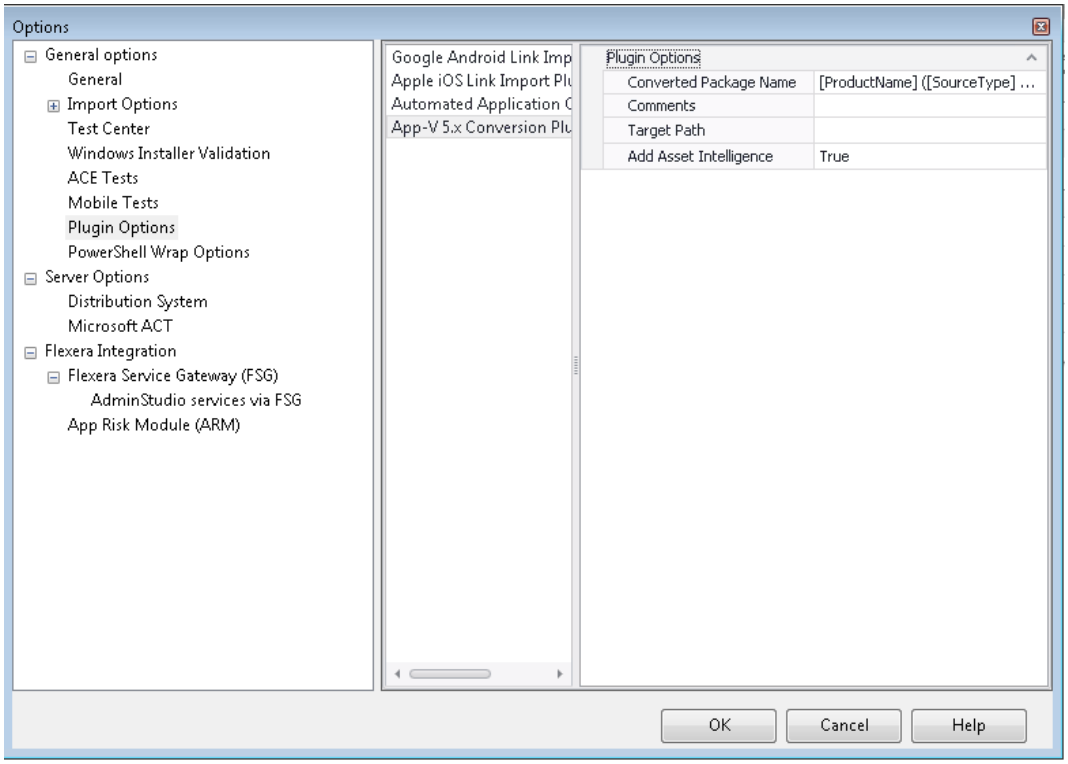
6. Save the settings file and exit Automated Application Converter.

Setting App-V 5.0 Conversion Options

Prior to using the Conversion Wizard to convert an App-V 4.x package to App-V 5.0 format, as described in Converting App-V 4.x Packages to App-V 5.0 Format, you first need to perform the following steps to set App-V 5.0 conversion options in Application Manager.

Task ***To set App-V 5.0 conversion options:***

1. In Application Manager, select **Options** on the Application Catalog menu. The **Options** dialog box opens.
2. Open the **General options > Plugin Options** tab.
3. In the middle pane, select **App-V 5 Conversion Plugin**. The App-V 5.0 conversion options are displayed in the right pane.



4. In the **New Package Name** field, enter a name to differentiate the converted version of the package from the original version. By default, this field will be populated with the original package name `[ProductName]`. For example:
   - `[ProductName]`
   - `[ProductName]_v5`
   - `[Manufacturer]_[ProductName]_v5`
5. In the **Comments** field, enter metadata that you would like to add to each converted package. This text will be displayed in the **Administrator Comments** field on the **Package Information** tab of the **Home Deployment Type View** for each package.
6. In the **Target Path** field, specify the output folder where you want the converted packages to be located.

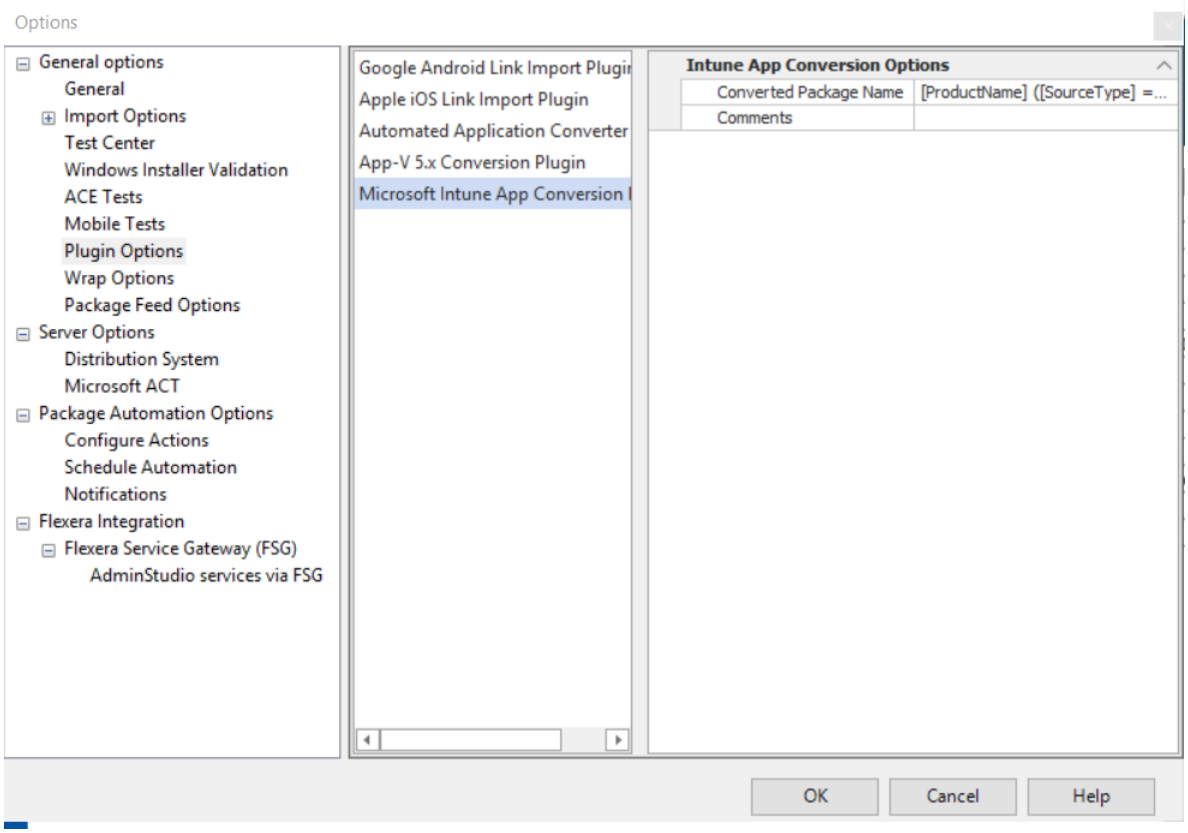
7. Asset intelligence is used to enhance the inventory capabilities of Microsoft System Center 2012 Configuration Manager by extending hardware inventory and adding license management functionality. The asset intelligence features can report application data such as digital PID, MSI product codes, and publisher names for each virtual application registered on a client computer. To add asset intelligence information to a converted App-V 5.x package, set this option to **True**.
8. Click **OK**.

Setting Microsoft Intune Conversion Options

Prior to using the Conversion Wizard to convert an MSI or EXE package to Microsoft Intune format, as described in Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard, you first need to perform the following steps to set Microsoft Intune conversion options in Application Manager.

Task ***To set App-V 5.0 conversion options:***

1. In Application Manager, select **Options** on the Application Catalog menu. The **Options** dialog box opens.
2. Open the **General options > Plugin Options** tab.
3. In the middle pane, select **Microsoft Intune App Conversion Plugin**. The **Microsoft Intune App Conversion Plugin** options are displayed in the right pane.



4. In the **Converted Pacakge Name** field, enter the format of the converted page name to differentiate it from the converted version of the package from the original version. By default, this field will be populated with:

`[ProductName] ([SourceType] => [TargetType])`

5. In the **Comments** field, enter metadata that you would like to add to each converted package. This text will be displayed in the **Administrator Comments** field on the **Package Information** tab of the **Home Deployment Type View** for each package.
6. Click **OK**.

Converting App-V 4.x Packages to App-V 5.0 Format

You can upgrade an App-V 4.x package or group of packages to App-V 5.0 format directly from Application Catalog using the Conversion Wizard.

Important • *To perform this upgrade, the Microsoft Application Virtualization Sequencer Version 5.0 must be installed on the same machine as AdminStudio.*

Important • *If AdminStudio is installed on a Windows 7 (x64) machine, you will need to first set the PowerShell execution policy to "unrestricted" before attempting to use the Conversion Wizard to upgrade an App-V 4.x package to App-V 5.0 format. To do this, execute the following command on an elevated Windows PowerShell (x86) utility:*

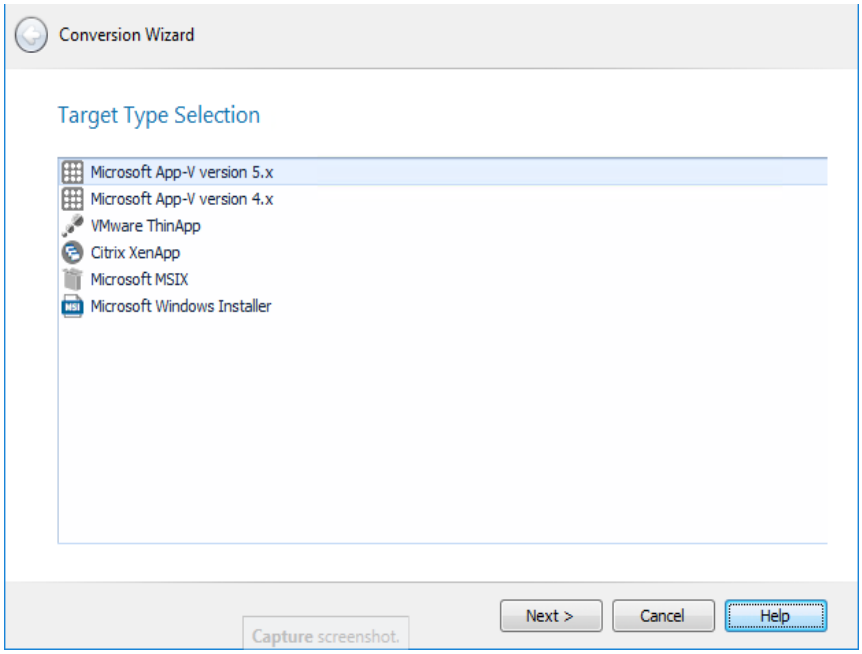
```
Set-ExecutionPolicy Unrestricted
```

Important • *AdminStudio relies on the App-V Sequencer to perform the conversion from 4.x to 5.0. Some elements of highly customized App-V packages are not carried to the 5.0 package during the conversion by the App-V Sequencer. Some of these customizations include OSD scripting and dependencies.*

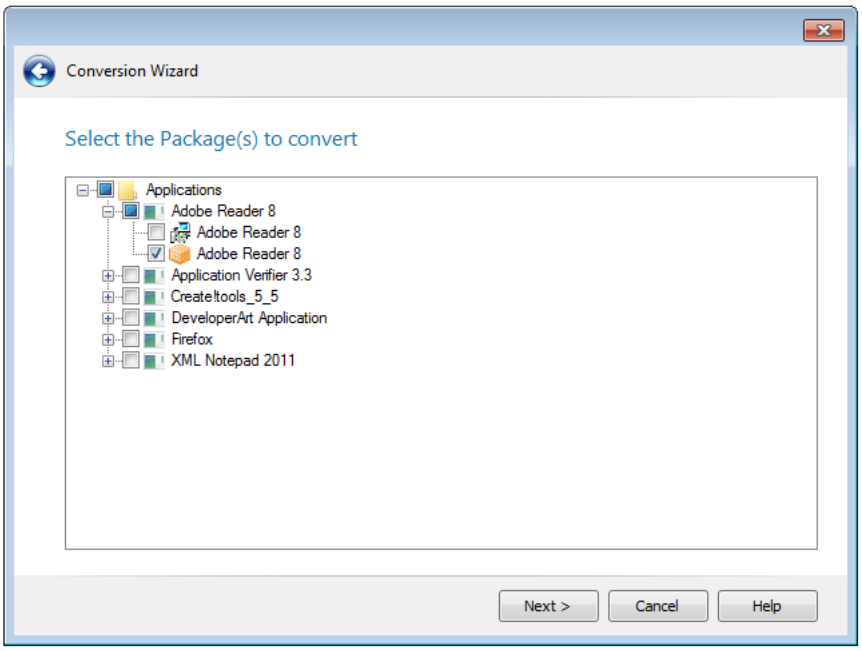
To upgrade App-V packages from version 4.x to 5.0, perform the following steps:

Task ***To upgrade an App-V 4.x package to App-V 5.0 package:***

1. Set App-V 5.0 conversion options, as described in Setting App-V 5.0 Conversion Options.
2. On the Application Manager Home tab, right-click on one of the following in the tree:
   - App-V 4.x package
   - Application containing an App-V 4.x package
   - Group containing one or multiple App-V 4.x packages
- Select **Launch Conversion Wizard** from the shortcut menu. The **Target Type Selection** panel opens.

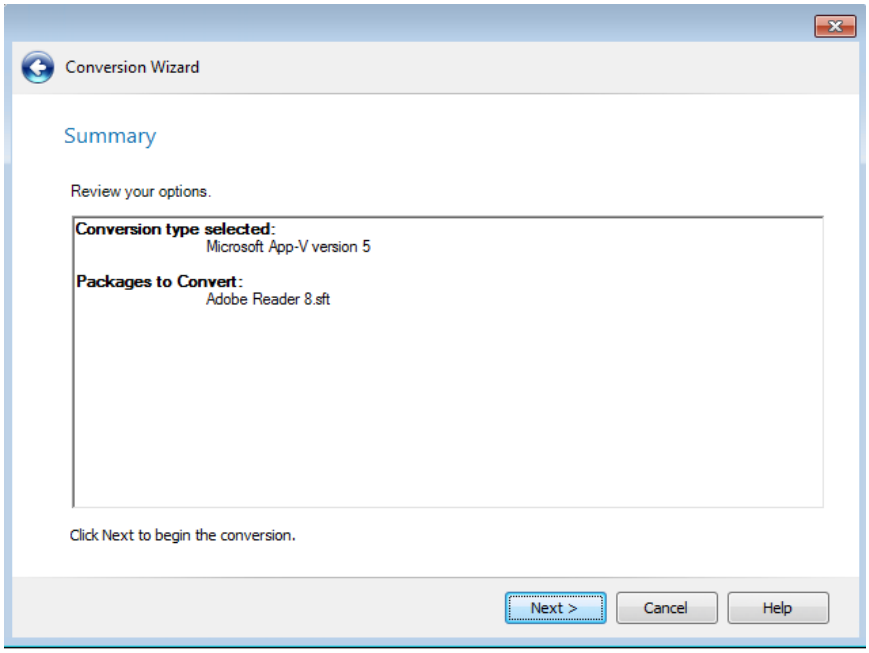


3. Select **Microsoft App-V version 5** and click **Next**. The **Select the Package(s) to convert** panel opens, and the App-V 4.x packages or packages that you had selected when you invoked the Conversion Wizard are selected.

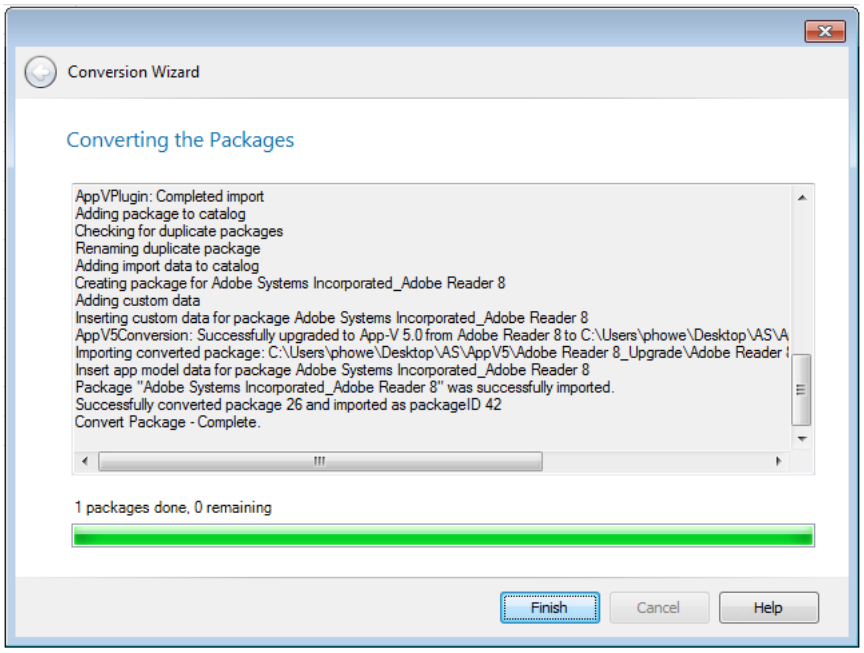


Tip • *In addition to converting App-V 4.x packages to App-V 5.0, you can also use the Conversion Wizard to convert Windows Installer and legacy installers to App-V 5.0 format. See Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging.*

4. Select additional packages, if desired, and click **Next**. The **Summary** panel opens.



5. Click **Next** to begin the conversion. The **Converting the Packages** panel opens.
6. Click **Next**. When conversion is complete, the **Summary** panel opens, confirming that the upgrade has been performed.



7. Click **Finish**. The converted App-V 5.0 package is now listed in the tree, next to the 4.x package.

Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging

Edition • *The Conversion Wizard is included with AdminStudio Professional and Enterprise Editions, and you can use it to perform automated repackaging on a virtual machine.*

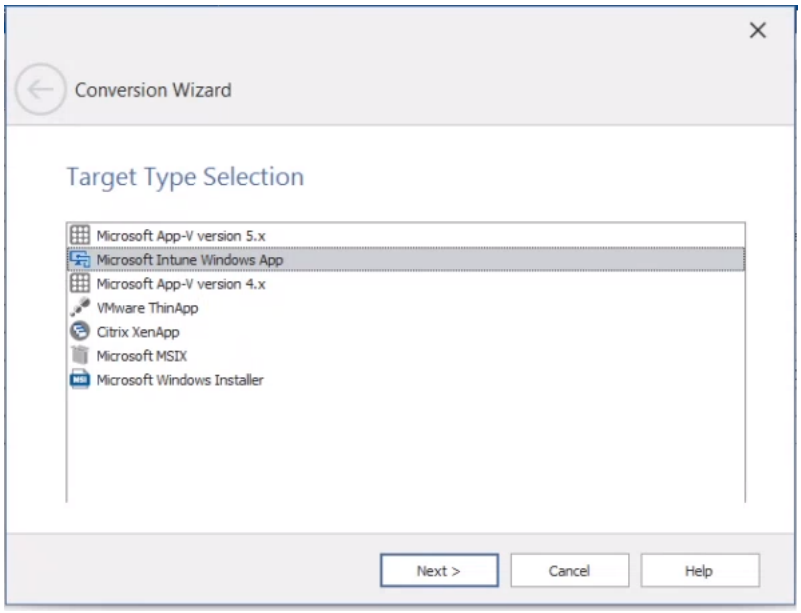
As described in Performing Virtualization and Repackaging Using the Automated Application Converter, you can use the Automated Application Converter tool to convert a single package or a group of packages into Microsoft App-V, VMware ThinApp, Citrix XenApp, MSIX packages and virtual application formats. Using Automated Application Converter to perform this task enables you to make a wide variety of customizations by editing a package's general and App-V-related properties prior to conversion.

However, you can also use the Application Catalog **Conversion Wizard** to quickly convert one or multiple Windows Installer packages, PowerShell-wrapped packages, or legacy installers to virtual packages (of the specified type) or repackaged Windows Installer packages using either the *default* Automated Application Converter settings or using settings that are customized for that run of the Conversion Wizard.

To use the Conversion Wizard to perform virtual package conversion or repackaging, perform the following steps:

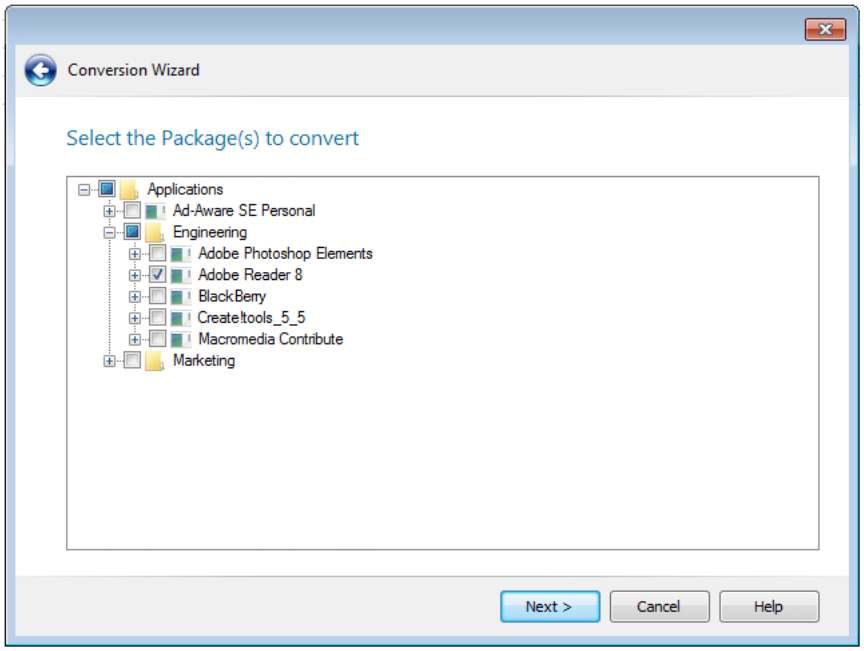
Task ***To use the Conversion Wizard to perform virtual package conversion or repackaging:***

1. On the Application Manager Home tab, right-click on a Windows Installer package, PowerShell-wrapped package, or legacy installer (or a group containing packages of that type) in the tree and select **Launch Conversion Wizard** from the shortcut menu. The **Target Type Selection** panel opens.

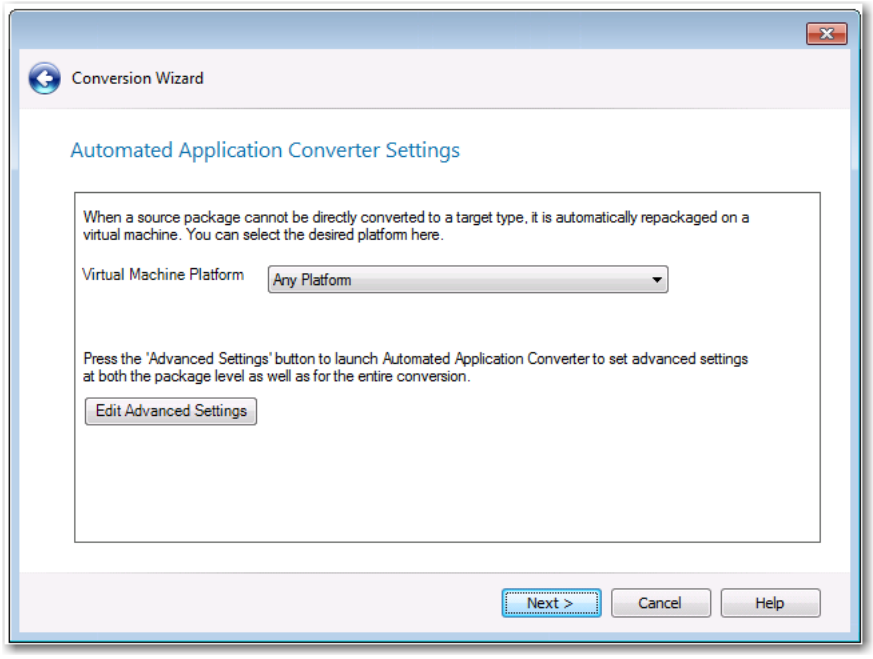


Note • *You can only convert to one virtual package type at a time.*

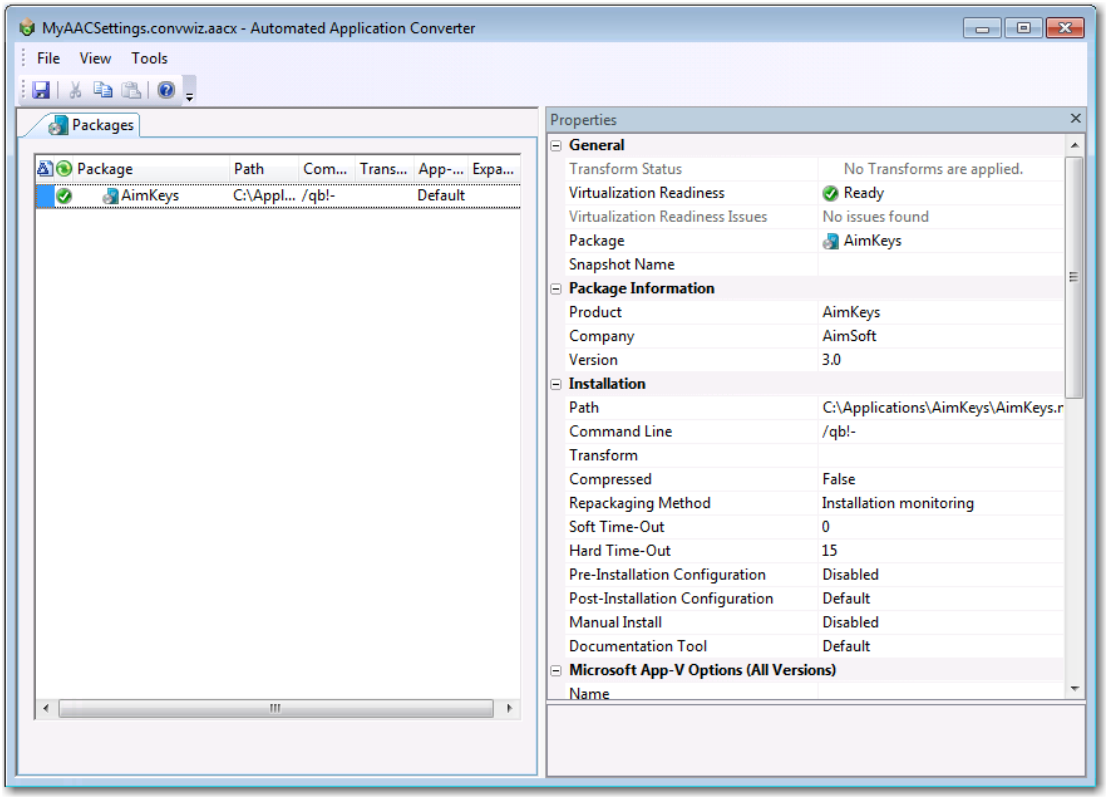
2. Select the desired virtual package conversion type and click **Next**. The **Select the Package(s) to convert** panel opens, and only the package or groups of packages that you had selected when you invoked the Conversion Wizard is selected.



3. Select additional packages, if desired, and click **Next**. The **Automated Application Converter Settings** panel opens.



4. The virtual machine platforms defined in the settings file (that is specified on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box) are listed in the **Virtual Machine Platform** list. Select the platform to use for this run of the Conversion Wizard.
5. If you want to edit additional advanced settings, click the **Edit Advanced Settings** button. A copy of the default conversion settings file is opened, displaying the **Packages** tab of Automated Application Converter.

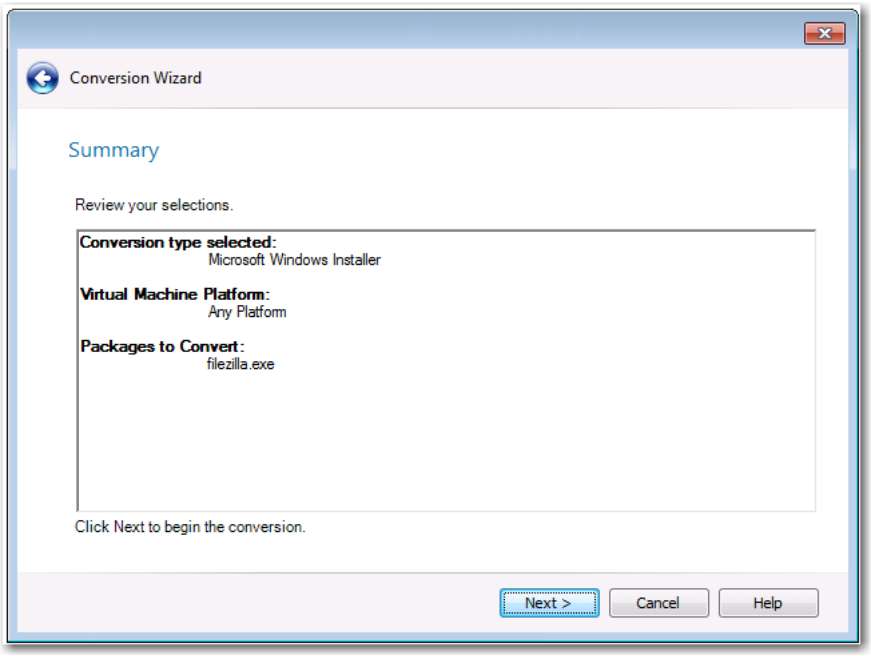


Note • *If the **Target Type Selection** is Microsoft **MSIX** package, you must select either **Certificate File (.pfx)** or **Certificate Store** in the package properties. For more information see Setting MSIX Signing Options.*

6. Modify these settings, if desired, and then save the file and exit Automated Application Converter. You are returned to the **Automated Application Converter Settings** panel of the Conversion Wizard.

Important • *Changes that you make to these settings are only used for this run of the Conversion Wizard. To change the default settings, you need to edit the settings file that is specified on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box.*

7. Click **Next**. The **Summary** panel opens.



8. Click **Next** to begin the conversion. The **Converting the Packages** panel opens.
9. Click **Next**. When conversion is complete, the **Summary** panel opens, confirming that the conversion has been performed.
10. Click **Finish**. The converted package is now listed in the tree; an additional deployment type has been added to the source package's parent application.

Using Test on Virtual Machine Wizard

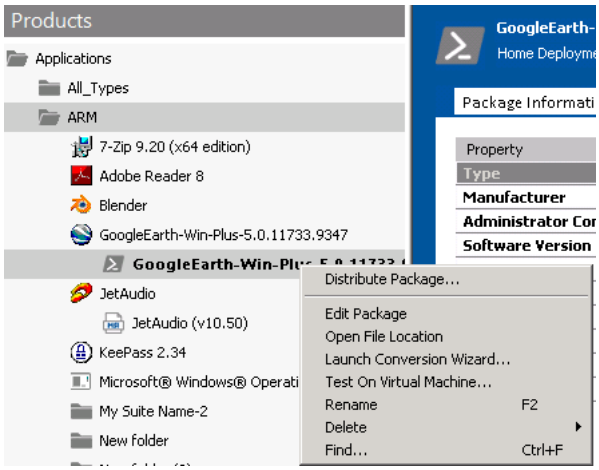
You can use the **Test on Virtual Machine Wizard** to quickly launch a specified virtual machine and install a selected Windows Installer (`.msi`) package, PowerShell-wrapped (`.ps1`) package, App-V package (`.appv`), or installation executable (`.exe`) package (both legacy installers and complex installation executables) for testing. This wizard uses the capability of the Automated Application Converter tool to spin up the selected virtual machine and install the selected package.

Note • *Both legacy installers and complex installer executables (which contain bundled Windows Installer packages) can be tested using the Test on Virtual Machine Wizard.*

To use the Test on Virtual Machine Wizard to test a package, perform the following steps.

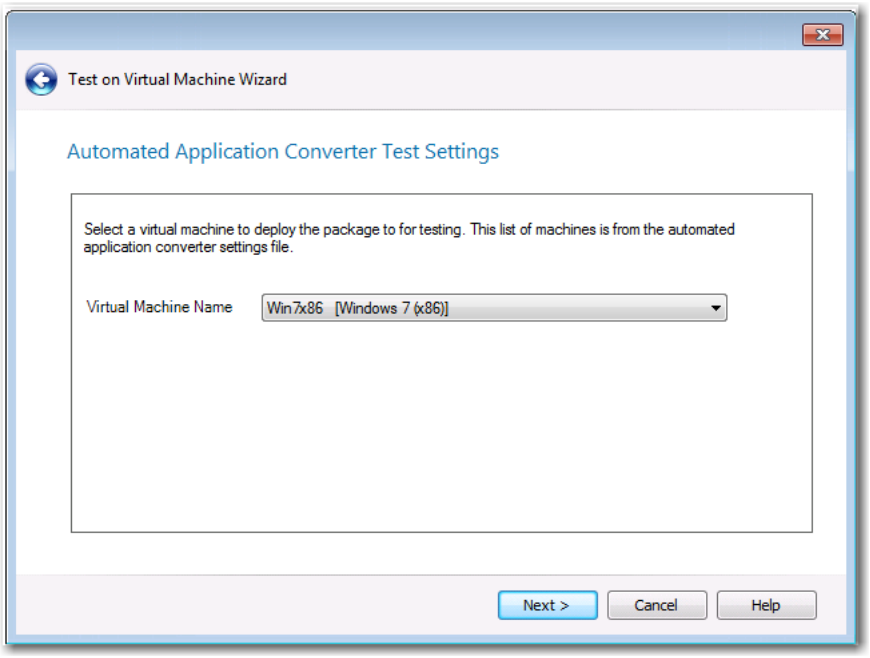
Task ***To use the Test on Virtual Machine Wizard to test a package:***

1. In the Application Manager tree, click the deployment type node of a Windows Installer package (`.msi`), App-V package (`.appv`), PowerShell-wrapped package (`.ps1`), or installer executable package (`.exe`) and select **Test on Virtual Machine** from the shortcut menu.

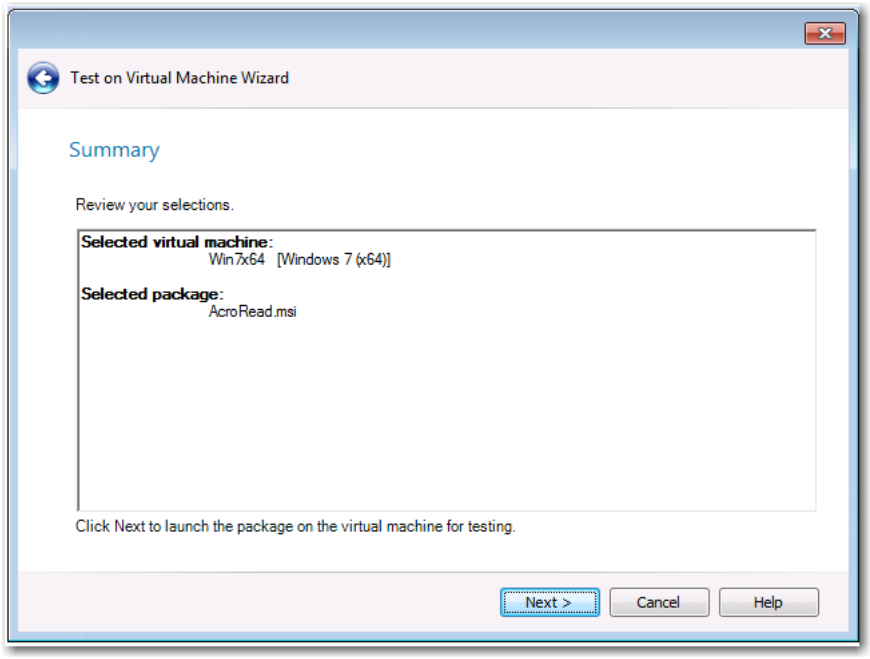


Note • *The **Test on Virtual Machine** selection on the shortcut menu is available on both the **Home** and the **Analyze** tabs of Application Manager.*

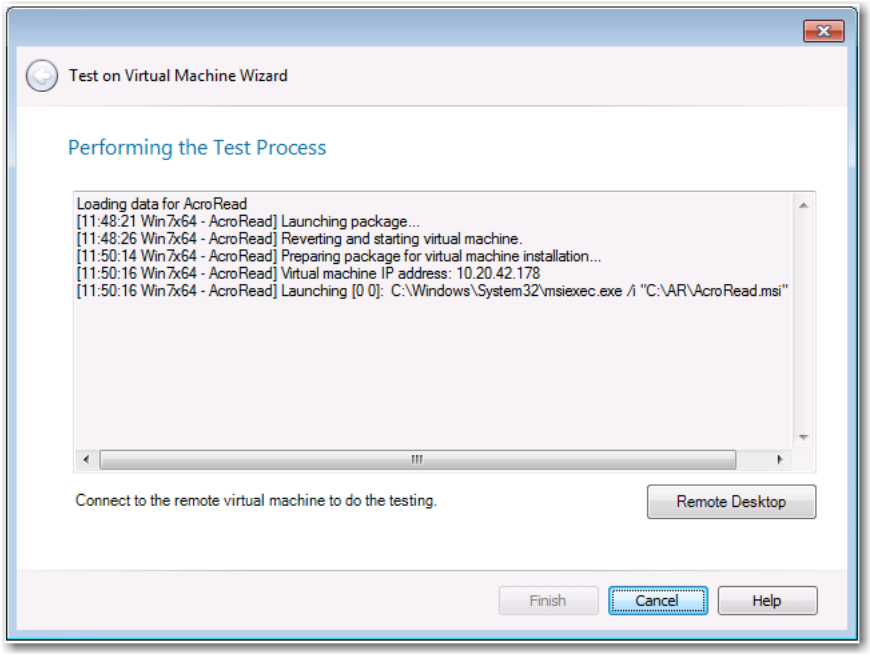
The **Automated Application Converter Test Settings** panel opens, listing the virtual machines defined in the Automated Application Converter settings file that is selected on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box:



2. From the **Virtual Machine Name** list, select the name of the virtual machine that you want to use for testing and click **Next**. The **Summary** panel opens.

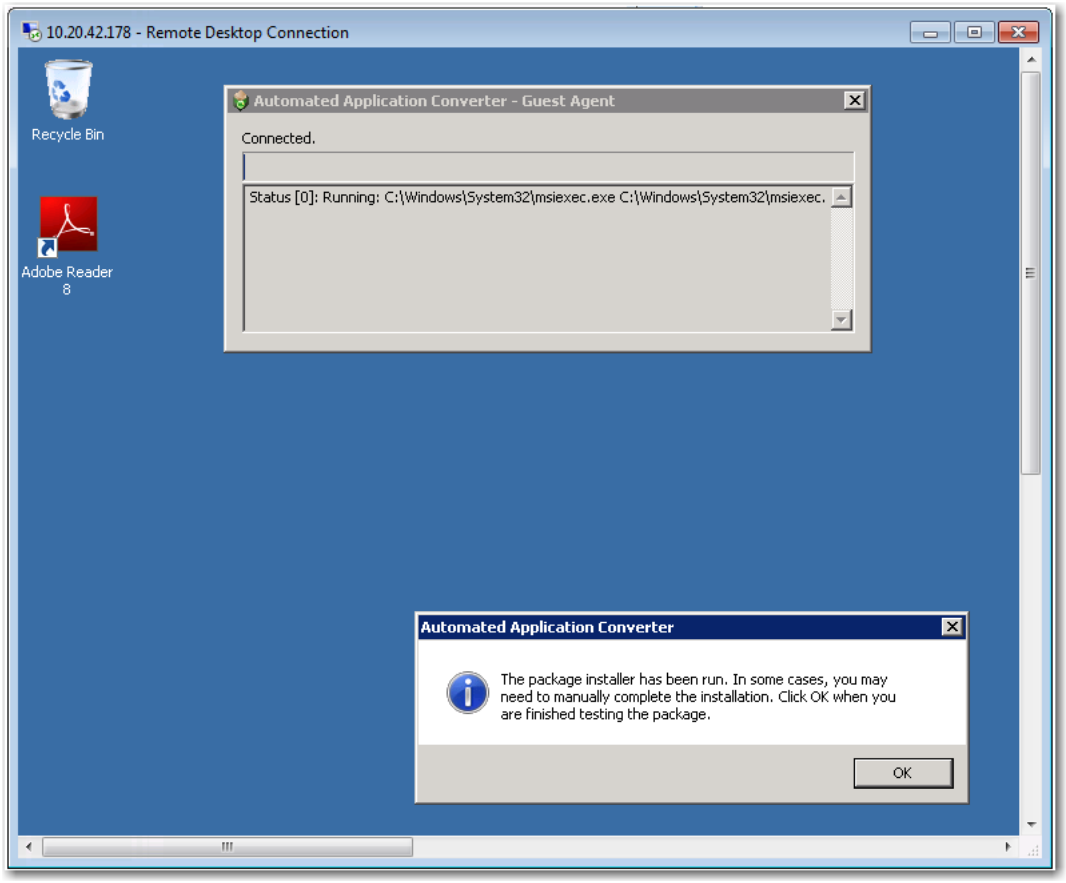


3. Click **Next** to launch the package on the specified virtual machine for testing. The **Performing the Test Process** panel opens, listing progress messages.



When the package has been installed and launched on the virtual machine, the **Remote Desktop** button will become enabled.

4. Click **Remote Desktop** to connect to the virtual machine and perform testing. You may be prompted for login credentials to the virtual machine image. A Remote Desktop session opens displaying the virtual image where this package has been installed.



5. Use the installed shortcuts to launch the package and perform the desired testing.
6. When you have finished testing the package, click **OK** to close the Remote Desktop session and shut down the virtual machine.
7. Return to the **Test on Virtual Machine Wizard** and click **Finish** to close the wizard.

Software Repository Integration into Other AdminStudio Tools

The Software Repository feature is integrated into several other AdminStudio tools:

- InstallShield Editor
- Virtual Package Editor
- Distribution Wizard
- Automated Application Converter

InstallShield Editor

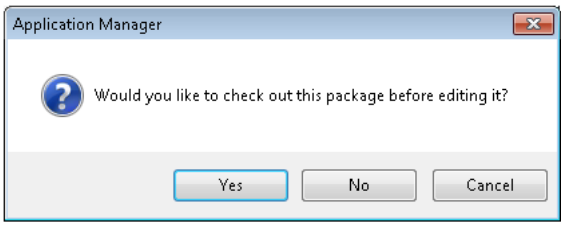
The Software Repository feature is integrated with InstallShield Editor in the following ways:

- You can launch InstallShield Editor from Application Manager.
- From InstallShield Editor, you can browse for packages that are stored in the Software Repository and select them for edit. You can either check a file out for edit or simply get the latest version of the file to edit. You can also undo a check out from InstallShield Editor.
- From the InstallShield Editor, you can add a package to the Software Repository.
- You can add a package to the Software Repository via the InstallShield Editor build process.

For more information on InstallShield Editor's integration with AdminStudio, see InstallShield Editor Integration with Application Catalog and the Software Repository.

Virtual Package Editor

When you launch Virtual Package Editor from Application Manager by right-clicking on an App-V package and then selecting **Edit with Virtual Package Editor** from the shortcut menu, you are prompted to check out the package:



Distribution Wizard

You can launch the Package Distribution Wizard from Application Manager by right-clicking on a package node (not an application node) and then selecting **Distribute Package** from the shortcut menu. When you do this, the package name displayed on the **Package Information** panel is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

- **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package.
- **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.

Automated Application Converter

When publishing packages from Automated Application Converter to a Software-Repository-enabled Application Catalog, the files will be added to the Software Repository.

Taking OS Snapshots

The OS Snapshot Wizard provides a simple way to capture your basic operating system configuration. Using the OS Snapshot Wizard, you can scan your computer's operating system and record the files, INI files, shortcuts, and registry entries present. The Wizard then creates an .osc file representing the system contents. You can import this snapshot file into the an Application Catalog database to identify potential conflicts between Windows Installer-based setups and your operating system.

To provide maximum flexibility during the OS Snapshot process, you can create an exclusion list (similar to the Repackager exclusion list) that identifies files, INI files, shortcuts, and registry entries that the OS Snapshot Wizard should disregard during the scan. Using this list, you can eliminate unnecessary files, shortcuts, or registry entries, and reduce the time it takes to perform the OS Snapshot.

The following topics are included in this section:

- OS Snapshot Best Practices
- Configuring OS Snapshot Analysis Options
- Capturing an OS Snapshot

Caution • *OS Snapshots should only be used for comparison in Application Manager. You should never attempt to convert an OS Snapshot into an MSI package.*

OS Snapshot Best Practices

Before capturing OS Snapshots, consider the following:

Table 7-22 • OS Snapshot Best Practices

Guideline	Description
Capture on a Clean System	For optimal OS Snapshots, you should only capture on a clean system. This ensures an accurate baseline of files necessary for the operating system. This also means that you should never attempt to capture other packages in addition to the operating system. Use Repackager when you need to capture applications.
Exit Other Applications	Shut down all other applications besides OS Snapshot prior to capture. Ideally, this should be done from the Windows Task Manager. This ensures that files are not locked during capture, and unnecessary temporary files are not inadvertently captured.
Only Use OS Snapshot for Import into Application Catalog	Never attempt to convert an OS Snapshot file into a Windows Installer package to install an operating system. AdminStudio does not support this use of OS Snapshots.
OS Snapshots Take Time	Depending on the operating system configuration, OS Snapshot often takes a significant amount of time to capture the base OS state. Consider that many typical OS installations exceed 500MB and contain tens of thousands of files, translating into a lengthy operation of cataloging and converting the files into an OS Snapshot file.

Table 7-22 • OS Snapshot Best Practices (cont.)

Guideline	Description
Take Multiple OS Snapshots	If your environment contains either multiple operating systems, or variations on the same operating system, take snapshots of each OS or variation. You can store all of these in your Application Catalog, allowing you to make comparisons between MSI packages and each OS.
OS Snapshots and Repackaging Are Not the Same	OS Snapshots, as the name implies, is for snapshots of the operating system only. Repackaging is only for traditional installation packages. These operations, although similar, are still very specialized and should only be used for their respective purposes.

Configuring OS Snapshot Analysis Options

Task ***To configure analysis options for OS snapshot captures:***

1. Launch the **OS Snapshot Wizard** by clicking on its icon in the AdminStudio Tools Gallery. The **Welcome** panel opens.
2. Click **Next**. The **Project Information** panel opens.
3. Click **Edit**. The **Analysis Options** dialog box opens.
4. Configure the types of data you want to capture (Files, INI files, Shortcuts, and/or Registry data), and specify if you want to restrict the snapshot to a specific drive.
5. Click **OK** to return to the **Project Information** panel.
6. Click **Start**.

Capturing an OS Snapshot

You use the OS Snapshot Wizard to capture OS Snapshots.

Task ***To capture an OS Snapshot:***

1. Launch the **OS Snapshot Wizard** by clicking on its icon in the AdminStudio Tools Gallery or by making a selection from the **Start** menu. The **Welcome** panel opens.
2. Click **Next**. The **Project Information** panel opens.
3. Provide the OS Snapshot project name and OS Snapshot project folder for the OS Snapshot file.
4. Optionally, click **Edit** to configure analysis options. See Configuring OS Snapshot Analysis Options.
5. Click **Start** to perform the OS Snapshot.
6. On completion of the OS Snapshot, review the results in the **Summary** panel.
7. Click **Finish**.

Note • *The OS Snapshot is stored as an OSC file in the folder defined in the* ***Project Information*** *panel.*

Capturing an OS Snapshot on a Clean Machine

If you would like to capture an OS snapshot on a clean machine, you can choose to either run the OS Snapshot Wizard remotely or to install it on a clean machine.

- **Installing OS Snapshot Wizard with Standalone Repackager**—The OS Snapshot Wizard is installed along with Standalone Repackager. For instructions on how to install OS Snapshot Wizard along with Standalone Repackager on a clean machine, see Installing Repackager on a Clean Machine.
- **Launching OS Snapshot Wizard remotely**—You can launch OS Snapshot Wizard remotely from the clean machine. For instructions, use the same method described in the Launching Repackager Remotely help topic.

Reference

This section contains information on the Application Manager views, dialog boxes and wizards that are accessible when the **Home** tab is selected in the ribbon. This Application Manager functionality is used when managing, connecting to, import data into, or sharing application catalogs.

- Application Manager Interface
- Application Manager Reports Tab
- Products Tree/Home Tab Views
- Merge Module Tree Views
- Environment Tree Views
- Dialog Boxes
- Wizards
- User Permissions in Application Manager
- Database Server Permissions

Application Manager Interface

The Application Manager user interface consists of following areas: the ribbon, the navigation window, the details pane, and the output window. Both the navigation window and the Output window are dockable.

- **Ribbon**—The ribbon provides quick access to Application Manager commands and functionality, and consists of the Application Catalog tab menu and four additional tabs: **Home**, **Analyze**, **Reports**. and **Support**.
- **Shortcut menu**—Additional commands are available on the shortcut menu, which opens when you right-click on a group, application, or package in the Application Manager tree.
- **Navigation window**—The navigation window consists of three tabs: **Products**, **Merge Modules**, and **Environment**.
- **Details pane**—When you select different items in the tabs of the navigation window, the details pane displays corresponding information about that item.

- **Output window**—The output window consists of tabs where output is displayed during different Application Manager processes.

This section details the Application Manager interface and includes the following topics:

- Application Manager Ribbon Interface
- Application Manager Tree and Subnode Icons
- Shortcut Menus
- Output Window

Application Manager Ribbon Interface

Starting with AdminStudio 11.0, Application Manager includes a ribbon interface to provide quick and easy access to Application Manager tasks.

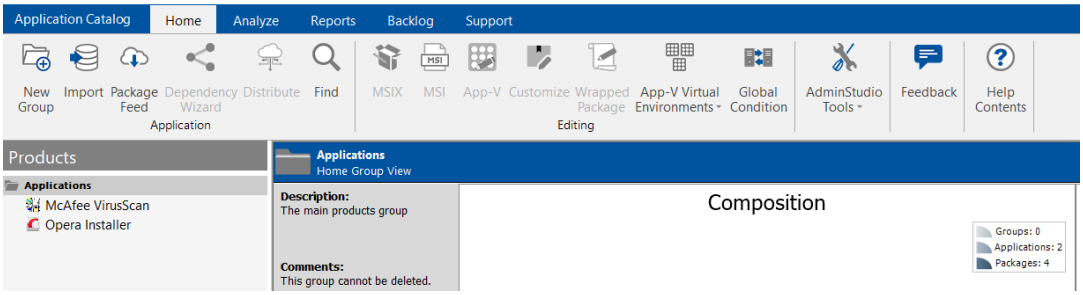


Figure 7-1: Application Manager's Ribbon Interface

The ribbon interface includes the Application Catalog tab menu, along with buttons that are grouped in four additional tabs: **Home**, **Analyze**, **Reports**, and **Support**.

- Application Catalog Tab Menu
- Home Tab of Application Manager Ribbon
- Analyze Tab of Application Manager Ribbon
- Reports Tab of Application Manager Ribbon
- Backlog Tab of Application Manager Ribbon
- Support Tab of Application Manager Ribbon

Application Catalog Tab Menu

The Application Catalog tab menu is opened by clicking the Application Catalog tab:

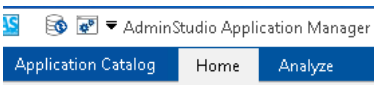


Figure 7-2: Application Catalog Tab and Quick Access Toolbar

The Application Catalog tab menu includes database connection related tasks, as well as commands for setting catalog properties and application options, and exiting from the application.

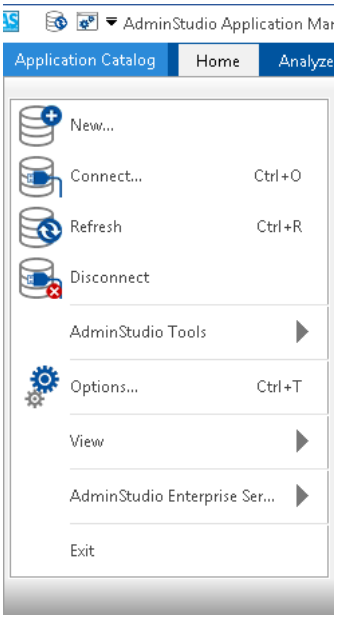


Figure 7-3: Application Catalog Tab Menu

The Application Catalog tab menu, and the Quick Access Toolbar, provide access to the following tasks:

Table 7-23 • Application Catalog Tab Menu & Quick Access Toolbar

Command	Description
New	Opens the **Application Catalog Wizard**, which you can use to create a new Application Catalog.
Connect	Displays the **Connect Application Catalog** dialog box, where you can open an existing Application Catalog.
Refresh	Refreshes the current views. This is particularly useful if multiple people are working on the same Application Catalog. You can also click the Refresh Database control in the Quick Access Toolbar.
Disconnect	Closes the currently open Application Catalog.
Options	Click to open the Options Dialog Box, where you can change various settings including whether to perform testing automatically after package import, default conflict tests to run, and connection information for Configuration Manager and a Microsoft ACT database.
View	Use to toggle the display of the **Status Bar** and **Output Window**.

Table 7-23 • Application Catalog Tab Menu & Quick Access Toolbar

Command	Description
AdminStudio Enterprise Server	Select one of the following options: • **Change AES Password**—Change the password of the current user to log in to the AdminStudio Enterprise Server. • **Log Out**—Log out of the AdminStudio Enterprise Server.
Exit	Click to close Application Manager.
Toggle Output Window	Click this icon in the Quick Access Toolbar to toggle the display of the Output Window.
Customize Toolbar	Click this icon to toggle whether to display the Quick Access Toolbar above or below the ribbon.

Home Tab of Application Manager Ribbon

The **Home** tab includes buttons to import packages into the Application Catalog, edit packages, use Software Repository commands, and distribute packages.

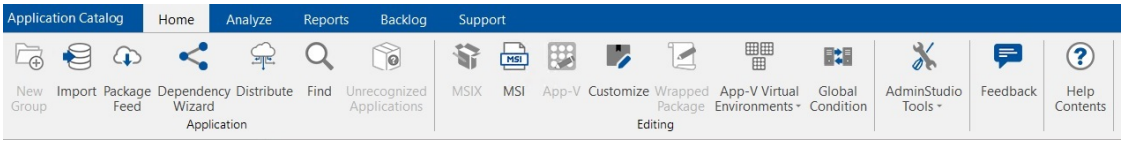


Figure 7-4: Home Tab of Application Manager Ribbon

The **Home** tab of the Application Manager ribbon provides access to the following tasks:

Table 7-24 • Home Tab of Application Manager Ribbon

Group	Button	Description
Application	**New Group**	Create a new group in the tree.
	Import	Launches the Import Wizard, allowing you to import Windows Installer packages, virtual packages (Microsoft App-V, VMware ThinApp, and Citrix).
	Package Feed	Launches the Import Wizard, allowing you to download and import Windows Installer packages, legacy installers and Mac packages from the Package Feed Module.
	Cancel Import	Cancel the import of an application or package.
	Dependency Wizard	Launch the Dependency Wizard. See Specifying Package Dependencies Deployment Data.
	Distribute	Distribute the selected application or package using Distribution Wizard.
	Find	Use to search for data in various tables in the Application Catalog.
	Unrecognized Applications	Used to generate a list of all applications in the Application Catalog that do not have an associated Flexera Identifier. See Managing an Application's Flexera Identifier.
Editing	**MSI**	Open the selected package in InstallShield Editor in Direct Edit Mode.
	MSIX	Open the selected package in the MSIX Editor.
	App-V	Open the selected package in the Virtual Package Editor.
	Customize	Customize the selected package using the Customization wizard. For more information, see Customization Wizard.
	PowerShell	Edit a PowerShell-wrapped package script (`.ps1` file) using the Windows PowerShell ISE application.
	App-V Virtual Environments	Click to open the **App-V Server Connection Groups** or **SCCM Server Environment** dialog box, which list all defined connection groups or server environments.
	Global Condition	Open the Global Conditions dialog box, where you can create new global conditions and edit existing global conditions.

Table 7-24 • Home Tab of Application Manager Ribbon

Group	Button	Description
Repository	**Checkout**	Check selected package out of Software Repository.
	Checkin	Check selected package into the Software Repository.
	Cancel Checkout	Cancel the check-in of a package.
	Version History	Open the **Package Versions** dialog box.
Feedback		Used to provide ideas and / or suggestions.
Support	**Help**	Open the AdminStudio help library.

Note *• Many of these commands can also be accessed through Shortcut Menus.*

Analyze Tab of Application Manager Ribbon

The **Analyze** tab includes buttons to analyze a package's application compatibility and best practices compliance, and to detect and resolve package conflicts.

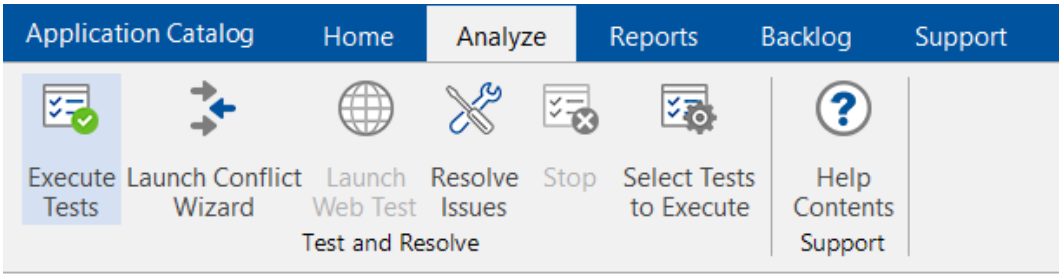


Figure 7-5: Analyze Tab of Application Manager Ribbon

The **Analyze** tab of the Application Manager ribbon provides access to the following tasks:

Table 7-25 • Analyze Tab of Application Manager Ribbon

Group	Button	Description
Test and Resolve	**Execute Tests** Shortcut: F4	Execute all of the tests currently selected on the **Select Tests to Execute** dialog box on the selected package, application, or group. For more information, see Performing Compatibility, Best Practices, and Risk Assessment Testing, and Performing Application Conflict Testing.
	Launch Conflict Wizard Shortcut: F5	Launches the Conflict Wizard to perform application conflict testing. For more information, see Performing Application Conflict Testing.
	Resolve Issues Shortcut: F7	Resolve any automatically resolvable conflicts that have been detected for the selected package. For more information, see Resolving Issues.
	Stop	Stops the text execution or conflict analysis.
	Select Tests to Execute	Opens the **Select Tests to Execute** dialog box, where you select the tests to use during package testing. For more information, see Configuring Testing Options on the Analyze Tab.
Repository	**Checkout**	Check selected package out of Software Repository.
	Checkin	Check selected package into the Software Repository.
	Cancel Checkout	Cancel the checkin of a package.
	Version History	Open the **Package Versions** dialog box.
Support	**Help**	Open the AdminStudio help library.

Reports Tab of Application Manager Ribbon

When you select the **Reports** tab of the Application Manager ribbon, you can view detailed reports on the status of the applications and packages in the Application Catalog.

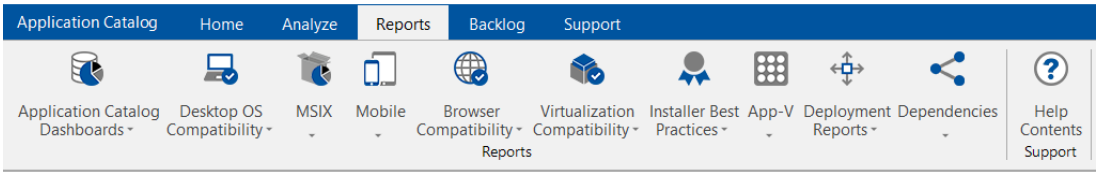


Figure 7-6: Reports Tab of Application Manager Ribbon

For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab.

Backlog Tab of Application Manager Ribbon

The **Backlog** tab helps you manage and execute the requests you get for creating packages for deployment. Backlog Tab is where you will maintain a backlog of all your package requests. A package request consists of details of an application like Product Name, Version, Vendor etc for which you wish to create a package for deployment. A package request also has other properties like Priority and Source to help you better manage your package request backlog. The **Backlog** Tab provides you an option to import bulk package requests at once, typically a list of applications in a .csv format obtained from a inventory system like ConfigMgr or other CMDB systems. You can leverage Package Automation in the **Backlog** Tab to subscribe packages for automation. For more information, see Using Package Automation.

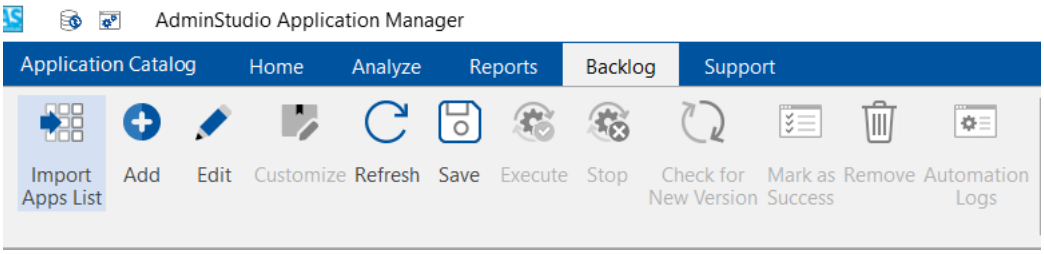


Figure 7-7: Backlog Tab of Application Manager Ribbon

The **Backlog** tab of the Application Manager ribbon provides access to the following tasks:

Table 7-26 • Backlog Tab of Application Manager Ribbon

Button	Description
Import Apps List	Imports a list of inventory packages that are listed in a .CSV file. For more information, see Importing a List of Applications for Automation. ***Note*** • *CSV file formats are supported for importing the packages. These CSV files should have mandatory column for* ***Product Name****. The* ***Product Vendor****,* ***Product Version****, and* ***Priority*** *are optional.*
Add	Add a single package request to the Backlog tab. For more information, see Adding a Single Package Request to the Backlog Tab.
Edit	Modify an existing package request details. For more information, see Editing a Package Request in the Backlog Tab.

Table 7-26 • Backlog Tab of Application Manager Ribbon

Button	Description
Customize	Launch customization wizard for the selected package. For more information, see Customization Wizard.
Refresh	Sync package feed to check any new versions.
Save	Used to save the changes. ***Note*** • *It is a default action when ever you* ***Refresh***, ***Execute***, ***Check for New Version***, *and* ***Mark as Success*** *the package (s).*
Execute	Used to execute the selected inventory packages. This option will be enabled when the package node is selected in the Version in Package Feed column. ***Note*** • *Inventory Package is a file, which contains a list of package information. This file will be in .CSV format.*
Stop	Used to stop the execution process.
Check for New Version	Used to check the latest versions.
Mark as Success	Until it is marked as success, the failed inventory packages can not be executed.
Remove	Used to remove the selected inventory records from the grid.
Automation Logs	Displays last run scheduled logs for both Package Feed and Monitored Directory packages. ***Note*** • *In the event of Application Manager been closed (owing to any reason), then reconfigure the scheduler.*

Support Tab of Application Manager Ribbon

The **Support** tab includes buttons to give you quick access to the AdminStudio help library and information specific to the current release of AdminStudio.

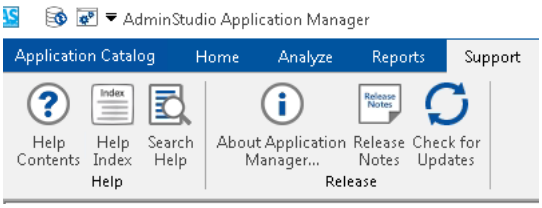


Figure 7-8: Support Tab of Application Manager Ribbon

The **Support** tab of the Application Manager ribbon provides access to the following tasks:

Table 7-27 • Support Tab of Application Manager Ribbon

Group	Button	Description
Help	**Help Contents**	Open the **Contents** tab of the AdminStudio Help Library
	Help Index	Open the **Index** tab of the AdminStudio Help Library.
	Search Help	Open the **Search** tab of the AdminStudio Help Library.
Release	**About Application Manager**	Open the product's **About** dialog box, which contains release information.
	Release Notes	Open an HTML version of the current AdminStudio Release Notes.

Application Manager Tree and Subnode Icons

The icons used in the Application Manager tree and its subnodes are described in this section.

- Application Manager Tree Icons
- Application Manager Subnode Icons

Application Manager Tree Icons

These icons are used on the Application Manager **Products**, **Merge Modules**, and **Environment** tabs in the tree:

Table 7-28 • Application Manager Tree Icons

Name	Icon	Description
Group		A group, which is used to organize your data.

Table 7-28 • Application Manager Tree Icons (cont.)

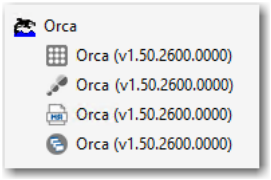
Name	Icon	Description
Application		An application that has been imported into the Application Catalog database. The application can include multiple deployment types—such as a Windows Installer package, a Microsoft App-V package, a VMware ThinApp package, etc. Each deployment type is indicated by a separate subnode of this application node. For most applications, an icon included in the application files is used to represent it in the tree. If an application does not include an icon, this default icon is used. In the following example, the application includes a Microsoft App-V package, a Citrix XenApp package, a VMware ThinApp package, and a Windows Installer package. 
Windows Installer package		Windows Installer package (`.msi`) that has been imported into the Application Catalog
App-V virtual package		Microsoft App-V virtual package (`.sft` or `.appv`) that has been imported into the Application Catalog
XenApp virtual package		Citrix XenApp virtual package (`.profile`) that has been imported into the Application Catalog
ThinApp virtual package		VMware ThinApp virtual package (`.exe`) that has been imported into the Application Catalog
iOS mobile app (local)		Apple iOS mobile app (`.ipa`) that has been imported into the Application Catalog
iOS mobile app (public store)		A link to an iOS mobile app in the Apple Store that has been imported into the Application Catalog.
Microsoft UWP app package or Windows 8 app		Microsoft UWP app package (`.appx`) or Windows 8 app (**.appx**) that has been imported into the Application Catalog.
Windows Store mobile app (public store)		A link to a Windows Store mobile app in the Microsoft Windows Store that has been imported into the Application Catalog.
Google Android mobile app (local)		Google Android mobile app (`.apk`) that has been imported into the Application Catalog

Table 7-28 • Application Manager Tree Icons (cont.)

Name	Icon	Description
Google Android mobile app (public store)		A link to a Google Android mobile app in the Google Play Store that has been imported into the Application Catalog.
Legacy application		A legacy application (.exe) that has been imported into the Application Catalog.
PowerShell-wrapped package		A Windows Installer package or complex installer executable that has been wrapped into a PowerShell script file (.ps1) and imported into the Application Catalog, or a package that has been converted using the Wrap Package Wizard.
MSIX packages		A Microsoft MSIX package (.msix).
Unresolved question overlay		An "!" overlay on an icon indicates that the package has an unresolved question with its associated files (or that one of an application's deployment types has an unresolved question with its associated files). One or more files, transforms, or patch files associated with a package is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog. When a package icon with this overlay is selected, a message appears in the view identifying the files in question and prompting you to take action to resolve the problem. For transforms, it is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog.
Software Repository overlay		This overlay icon indicates that the package is managed within the Software Repository.
Checked out overlay		This overlay icon indicates that the package is managed within the Software Repository and is checked out.
OS Snapshot		On **Environment** tab. An OS snapshot, which is a file representing a particular computer system's contents.
OS Security Patch		On **Environment** tab. Click to open the **Patch** view, which displays information about the selected OS Security Patch.
Enterprise Policy Configuration file		On **Environment** tab. Click to open the **Enterprise Policy View** view, which displays information about the selected Enterprise Policy Configuration.

Table 7-28 • Application Manager Tree Icons (cont.)

Name	Icon	Description
Merge Module		On **Merge Modules** tab. Indicates both the All Merge Modules view, the root node of the merge modules explorer containing a list of all merge modules in the Application Catalog, and an individual Merge Module.
Merge Module managed within the Software Repository		On **Merge Modules** tab. A Merge Module managed within in the Software Repository.

Application Manager Subnode Icons

When you select a subnode of package in the Application Manager tree, you see specialized views that provide information on that package. The subnodes that are available for a package depend upon which tab of the ribbon is selected (**Home** or **Analyze**).

Home Tab Subnodes

The following subnodes are available on the **Home** tab:

Table 7-29 • Application Manager Subnode Icons / Home Tab

Name	Icon	Description
Extended Attributes		Click to display the optional Extended Attributes associated with the package. See Extended Attributes View (Packages) for more information.
App-V History		Click to open a view which lists an entry for each time this App-V package has been saved. ***Note*** • *This information applies to App-V 4.x packages.*
Dependencies		Click to display the Dependencies associated with the package. See Dependencies View for more information. ***Note*** • *This information does not apply to App-V 5 packages.*
Files/Components, Files/Directories		Click to display the Files/Components or Files/Directories view for this package.
INI File Changes		Click to display the INI File Changes View, listing any INI file changes made by the product. ***Note*** • *Windows Installer packages only.*

Table 7-29 • Application Manager Subnode Icons / Home Tab (cont.)

Name	Icon	Description
Registry		Click to display any registry entries created or changed by the package.
Shortcuts		Click to display any shortcuts created by the package.
File Type Associations		App-V packages only. Click to view a list of the file type associations for this App-V package.
Environment Variables		App-V packages only. Click to display the environment variables used in this App-V package.
Merge Modules		Click to display any merge modules included the package.
Catalog History		Click to view a list of logged events for the selected package.
Tables		Click to view the data for a given package in the Application Catalog.
Components		Merge modules on Merge Modules tab. Select to display any components created or changed by the merge module.
Dependency		Merge modules on Merge Modules tab. Select to display any dependencies in the merge module.
Exclusion		Merge modules on Merge Modules tab. Select to display any exclusions in the merge module.
Files		Merge modules on Merge Modules tab. Select to display any files in the merge module
Products		Merge modules on Merge Modules tab. Select to display any products in the Application Catalog that use the merge module.

Analyze Tab Subnodes

The following subnodes are available on the **Analyze** tab:

Table 7-30 • Application Manager Subnode Icons / Analyze Tab

Name	Icon	Description
Patch Impacts		Click to access the Windows Installer package's Patch Impacts Analysis View, which lists patches for which there is patch impact data persisted against the product, and identifies the patch that caused the impact. ***Note*** *• Windows Installer packages only.*
Associated Patches		Click to display the product's Associated Patches View, which displays a list of imported patches that, if installed, would update the selected package. ***Note*** *• Windows Installer packages only.*

Shortcut Menus

Application Manager includes several shortcut menus which can be accessed by right-clicking on nodes on the **Products**, **Merge Modules**, and **Environment** tabs. These menus provide specific functionality in relation to what is clicked and what tab of the ribbon is selected (**Home** or **Analyze**).

- Groups and Applications
- Packages
- Package Request

Note *• This topic lists the shortcut menu commands available on the Application Manager **Products** tab. However, commands that are only available on the **Environment** or **Merge Modules** tabs are also noted.*

Groups and Applications

The shortcut menu that is displayed when a group or application is selected varies depending upon whether the **Home** tab or the **Analyze** tab of the ribbon is selected.

Home Tab of Ribbon

When you right-click on a group or application in the Application Manager tree, the following commands are available through the shortcut menu when the **Home** tab of the ribbon is selected

Table 7-31 • Group/Application Shortcut Menu Commands / Home Tab

Command	Description
Import / Import Package	Select this option to launch the Import Wizard to import Windows Installer and virtual packages into the Application Catalog.
Distribute Group / Distribute Application	Select to publish an application or group of applications to Configuration Manager. For more information, see Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager).
Launch Conversion Wizard	Select to open the Conversion Wizard, where you can upgrade App-V 4.x packages to App-V 5.0 format, or can convert a Windows Installer package to a virtual application. For more information, see Using the Conversion Wizard to Perform Virtualization or Repackaging.
New Group	Creates a new group within the selected group. ***Note*** *• For groups only.*
Rename	Allows you to rename the selected group or application.
Copy	Used in conjunction with the **Paste** command to enable you to share packages with multiple groups. See Copying and Sharing Packages in the Application Catalog. ***Note*** *• For applications only.*
Paste	Used in conjunction with the **Copy** command to enable you to share packages with multiple groups. See Copying and Sharing Packages in the Application Catalog. ***Note*** *• For applications only.*
Delete	Removes the selected group/application from the Application Catalog, including all of its contents.
Find	Select to perform a search for data in the various tables of the Application Catalog. See Searching an Application Catalog for more information.
Properties	Opens the **Properties** dialog box.

Analyze Tab of Ribbon

When you right-click on a group or application in the Application Manager tree, the following commands are available through the shortcut menu when the **Analyze** tab of the ribbon is selected:

Table 7-32 • Group/Application Shortcut Menu Commands / Analyze Tab

Command	Description
Execute Tests	Select to perform application compatibility and best practices testing on the packages in the selected group or application. Tests from the following test groups are run, depending upon the selections made on the **Select Tests to Execute** dialog box: • Operating System Compatibility • Remote Application Publishing Compatibility • Best Practices • Risk Assessment • Application Virtualization Compatibility
Resolve Issues	Select to resolve all detected issues for which an automatic resolution is available.
Launch Conflict Wizard	Select to launch the Conflict Wizard to perform conflict analysis on the packages in the selected group or application. The Conflict Wizard opens directly to the **Target Information** panel.
Launch Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft operating system patches and packages or OS Snapshots in the Application Catalog.

Packages

The shortcut menu that is displayed when a package is selected varies depending upon whether the **Home** tab or the **Analyze** tab of the ribbon is selected.

Home Tab of Ribbon

When you right-click on a package in the Application Manager tree, the following commands are available through the shortcut menu when the **Home** tab of the ribbon is selected:

Table 7-33 • Package Shortcut Menu Commands / Home Tab

Command	Description
Reimport Package	Select to reimport the selected package into the Application Catalog.
Customize	Launch the Customization Wizard. For more information, see Customization Wizard.
Convert/Repackage	Launch the Conversion Wizard.
Distribute Package	Launch the Distribution Wizard.

Table 7-33 • Package Shortcut Menu Commands / Home Tab (cont.)

Command	Description
Edit MSI with InstallShield	Open this package in InstallShield Editor, where you can directly edit the package. ***Note*** • *Supported for Windows Installer packages only.*
Create Transform with InstallShield	Create a transform file for the selected package and open it in InstallShield Editor. ***Note*** • *Supported for Windows Installer packages only.*
Associate with Workflow Manager Workflow Request	Launches the **Associate with Workflow Manager Workflow Request** dialog box, from which you can pick a package in Workflow Manager with which to associate the extended attribute data for the selected product. ***Note*** • *Supported for Windows Installer packages only.*
Rename	Rename the selected package.
Delete	When an application is selected, selecting **Delete** deletes the application and all of its deployment types. When a Windows Installer or virtual package is selected, you have the following options: • **Package**—Removes the product from the current Application. • **Package from all Applications**—Removes the package from all Applications and removes it from the Application Catalog. • **All Extended Attributes**—Removes all extended attributes from the selected package. • **History Log Information**—Removes all change history data for the selected package. • **Package Association**—Removes association with Windows Installer package. (Virtual packages only.) When a patch is selected on the **Environment** tab, you have the following options: • **Patch** • **Patch from all Groups** • **Patch Impact Data** When a merge module is selected on the **Merge Modules** tab, clicking **Delete** deletes the merge module from the Application Catalog.
Find in Package	Select to perform a search for data in the various tables of the Application Catalog. See Searching an Application Catalog for more information.

Table 7-33 • Package Shortcut Menu Commands / Home Tab (cont.)

Command	Description
Wrap	Launches the Wrap Package Wizard. For more information, see Wrap MSI/EXE Packages Using the Wrap Package Wizard.
Open Package File Location	Select to open the location where the package files are located.
Test on Virtual Machine	Launch the Test on Virtual Machine Wizard. For more information, see Test on Virtual Machine Wizard.
Update Package Info	Select to update the package information.

Analyze Tab of Ribbon

When you right-click on a package in the Application Manager tree, the following commands are available through the shortcut menu when the **Analyze** tab of the ribbon is selected:

Table 7-34 • Package Shortcut Menu Commands / Analyze Tab

Command	Description
Execute Tests	Select to perform application compatibility and best practices testing on the selected package. Tests from the following test groups are run, depending upon the selections made on the **Select Tests to Execute** dialog box: • Operating System Compatibility • Best Practices • Risk Assessment • Application Virtualization Compatibility • Remote Application Publishing Compatibility
Resolve Issues	Select to resolve all detected issues for which an automatic resolution is available.
Launch Conflict Wizard	Launches the Conflict Wizard directly to the **Target Information** panel.
Launch Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft patches and packages and OS Snapshots in your Application Catalog.
Generate Report	Select to generate a Patch Report for that patch. ***Note*** *• Available only when an OS security patch is selected on the* ***Environment*** *tab.*

Package Request

Important • *On* ***Version in Catalog*** *and* ***Version in Package Feed*** *columns, context menu options are not available.*

When you right-click on a package request in the **Backlog** tab, the following options are available through the context menu:

Table 7-35 • Package Shortcut Menu Commands / Backlog Tab

Command	Description
Execute	Select **Execute** to take the latest version of the package in Package Feed Module through all the pre-configured actions for package automation. For more information, see Matching the Applications. ***Note*** • *This option will be enabled when the package node is selected in the Version in Package Feed column.*
Customize	Launches the Customization Wizard. For more information, see Customization Wizard.
Edit	Select Edit to modify values of the selected package request. For more information, see Editing a Package Request in the Backlog Tab.
Mark as Success	Select **Mark as Success** to change the package request status to **Success**.
Clear Matching	Select **Clear Matching** to clear the current matches in **Version in Catalog** and **Version in Package Feed Module** columns for the selected package request.
Remove	Select to remove the selected package request from the **Backlog** Tab.
Add	Select to add a new package request to the **Backlog** tab. ***Note*** • *This option can be seen only on the empty grid.*

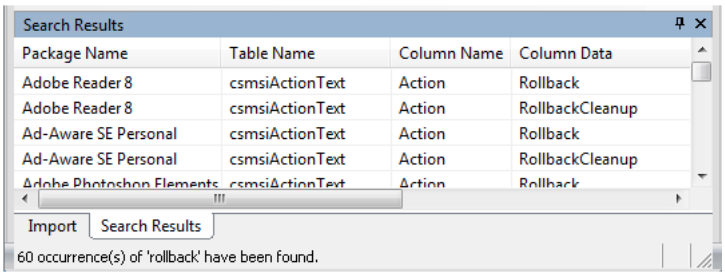
Output Window

When processes are performed on items in the Application Catalog or when a search is performed, the output messages and results of those wizards are displayed in the various tabs of the Output Window.

Table 7-36 • Output Window Tabs

Tab	Description
Output Tab	When testing is performed, messages are displayed in this tab.

Table 7-36 • Output Window Tabs (cont.)

Tab	Description
Patch Impact Tab	After the Patch Impact Analysis Wizard is run to identify conflicts between a Microsoft patch and a Windows Installer package or OS Snapshot, all of the impacts that were generated are listed in a table format. Each table row lists the following information: • **Patch**—Name of the patch that caused the conflict. • **Product**—Name of the impacted product. • **Type**—The type of impact. • **Description**—Description of the conflict that was found between the patch and the product. The following is an example of a conflict description: `Package 'WindowsXP v5.00' uses version '6.0.9589' of file 'SystemFolder\expsrv.dll' but Patch 'Windows2000-KB837001-x86-ENU' uses version '6.0.72.9589' of the same file` If you double-click on a row in this window, the conflicting file on the target product that is being impacted by the patch will be displayed and highlighted in the Tables View. For further information on the impacts generated by the Patch Impact Analysis, you may want to generate a Patch Report, or view the Product tab's Patch Impacts Analysis View, Dependencies View, and Associated Patches View. See Analyzing the Impact of Installing a Microsoft Operating System Patch for more information.
Search Results Tab	When Find is used to search for data in Application Catalog tables, the data that is found is displayed in this tab, in the following format: 

Application Manager Reports Tab

Edition • *The **Reports** tab is included with AdminStudio Enterprise Edition.*

On the **Reports** tab, AdminStudio provides a wide array of reports containing Application Manager summary information on the Windows Installer and App-V packages in your Application Catalog, giving you insight into the readiness of those packages for distribution and for conversion to virtual packages.

These reports include test results from operating system compatibility testing, best practices testing, application conflict testing, and remote application publishing compatibility. They also include information about the App-V packages in your Application Catalog, as well as ConfigMgr (Formerly called as System Center Configuration Manager) deployment information.

For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab.

Products Tree/Home Tab Views

The following views are displayed when the Application Manager **Products** tree is selected and the **Home** tab is selected in the ribbon.

- Group View
- Application View
- Home Deployment Type View
- Home Deployment Type View Subnode Views

Group View

The Group view, which is displayed on the right side of Application Manager whenever a group is selected, consists of pie charts that summarize the following information in the selected group:

- **Composition**—Displays the number of subgroups, applications, and packages in the selected group.
- **Packages**—Displays the number of packages in each of the following categories: installers (Windows Installer packages and legacy installers), virtual packages, mobile apps, and other.
- **Deployments**—Displays the number of packages that are deployed
- **Virtualizable**—Displays the number of virtual packages.

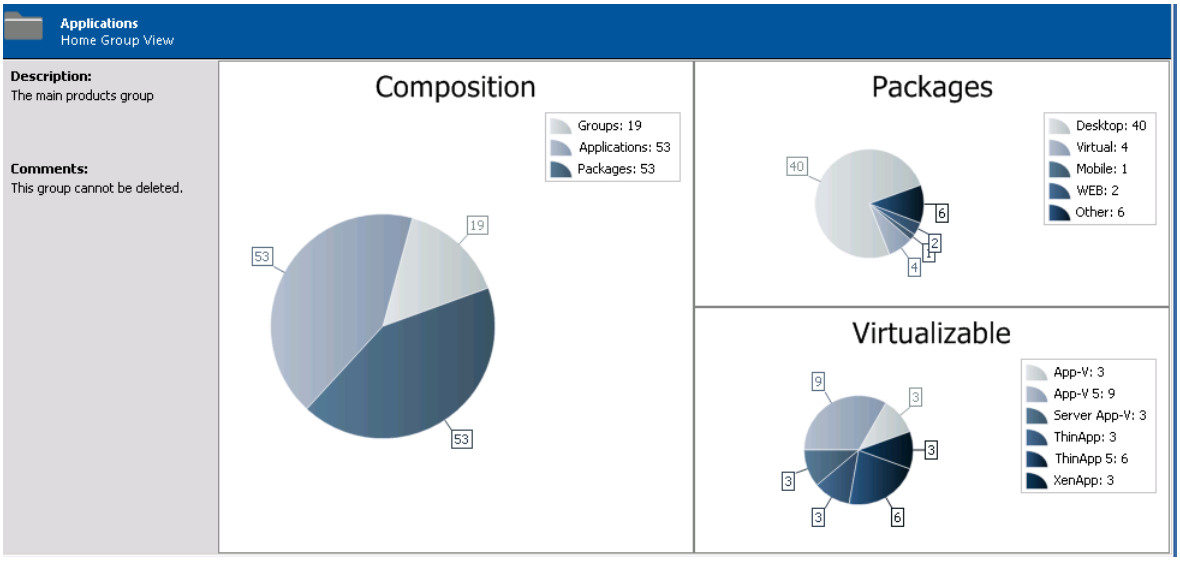


Figure 7-9: Home Tab / Group View

If you select an application or package in the Group View, properties for that application or package are displayed in the right pane.

Application View

When the **Home** tab is selected in the Application Manager ribbon and an application is selected in the tree, the **Application View** opens, which provides summary information about the application, deployment data for each of its deployment types, dependencies/supersedences information, and ConfigMgr (Formerly called as System Center Configuration Manager) and/or Citrix XenApp deployment status. Much of this information is used during deployment to ConfigMgr (Formerly called as System Center Configuration Manager).

The **Application View** presents this information on the following tabs:

- General Information Tab
- References Tab
- Deployment Status Tab
- App Portal Information Tab
- XenApp Deployment Types Tab
- Altiris Deployment Types Tab
- Workspace ONE Deployment Types Tab
- Extended Attributes Tab

General Information Tab

The **General Information** tab of the **Application View** lists summary information about the application that AdminStudio gathered during package import. Click in the **Value** column to edit a value.

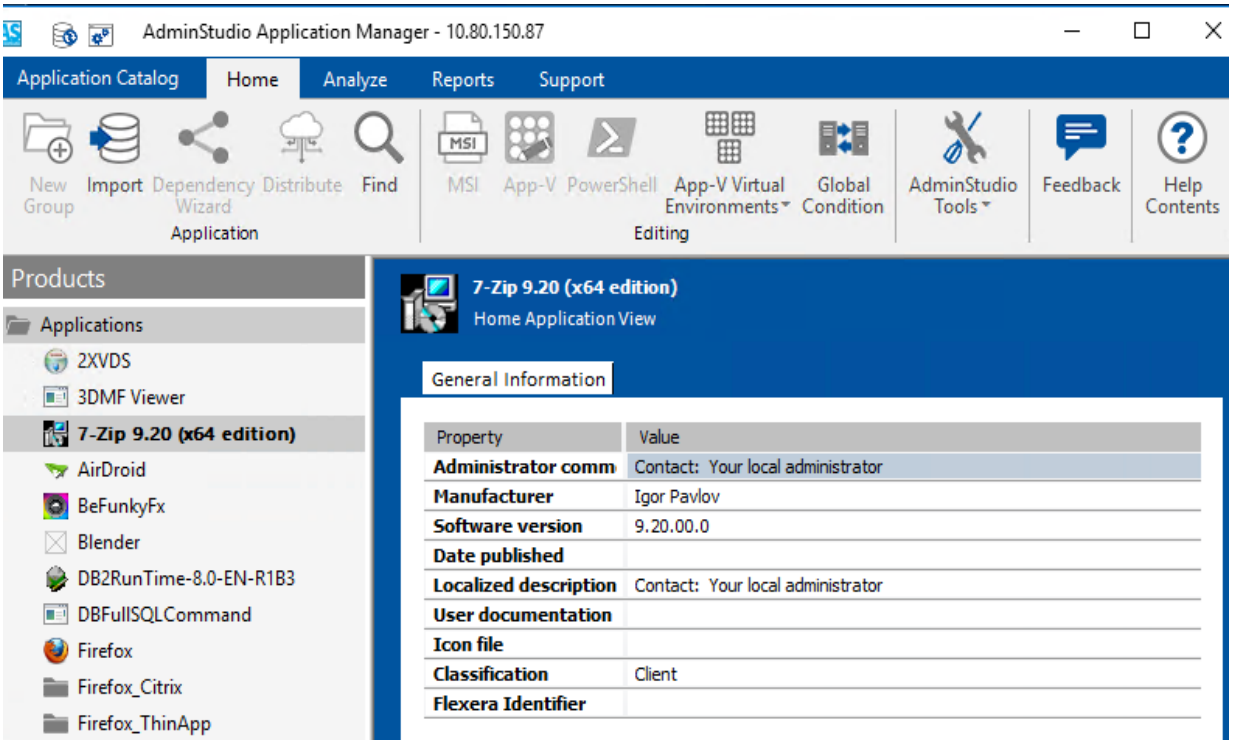


Figure 7-10: Application View / General Information Tab

The **General Information** tab of the **Application View** includes the following properties:

Table 7-37 • General Information Tab of Application View

Property	Description
Administrator comments	Comments related to this application, possibly regarding support for this application.
Manufacturer	Manufacturer of the application, as discovered from its deployment types.
Software version	Version of the application, as discovered from its deployment types.
Date published	The purpose of this field is to display the date the application was published to System Center 2012 Configuration Manager. When you create an application in Application Manager (usually by importing a package), this field is left blank. • If you do not enter a value in this field, when you publish the application to System Center 2012 Configuration Manager, this field will be automatically updated to display the published date. • If you enter a value in this field, and then publish the application to System Center 2012 Configuration Manager, the date that you entered will be listed as the published date in Configuration Manager.
Localized description	Description of this application written in the language of the target user.
User documentation	Location of documentation provided with this application.

Table 7-37 • General Information Tab of Application View

Property	Description
Icon file	Icon used in the Application Manager tree to represent this application. Click the browse button to open the **Change Icon** dialog box to specify a different icon by selecting an `.exe`, `.dll`, or `.ico` file.
Classification	Identifies whether this is a **Client** or **Server** application, or whether the application classification is **Unknown**. By default, this property is set to **Client** for all applications.
Flexera Identifier	A unique identifier assigned to applications by the FlexNet Manager Suite and stored in its libraries. The FlexNet Manager Suite Application Recognition Library uniquely identifies over 110,000 applications (including multiple versions and editions) from over 14,000 publishers. The Flexera Identifier key is used to link application information from Application Manager with application information in App Portal and FlexNet Manager Suite. ***Note*** *• A single Flexera Identifier represents an application and all of its deployment types.* ***Note*** *• This field is only populated with a Flexera Identifier if you have entered valid Flexera Service Gateway connection information on the Application Manager Options dialog box. For a description of the possible informational messages that could appear in this field, see Flexera Identifier Messages.*

Flexera Identifier Messages

When packages are imported into the Application Catalog, or when you click the **FlexNet Manager Platform** button on the **Flexera Service Gateway (FSG)** tab of the Application Manager Options dialog box, AdminStudio queries FlexNet Manager Suite for the Flexera Identifier of the imported applications, and messages are returned.

Information on an individual application's Flexera Identifier is displayed in the **Flexera Identifier** field on the **General Information** tab of the **Application View**.

Information on messages relating to an application's Flexera Identifier and communicating with FlexNet Manager Suite are described in the following sections:

- Flexera Identifier Field Messages
- Flexera Service Gateway (FSG) Tab of Options Dialog Box / Import Wizard Messages

Flexera Identifier Field Messages

The following messages are displayed in the **Flexera Identifier** field on the **General Information** tab of the **Application View** for an individual application.

Table 7-38 • Flexera Identifier Messages on Application View

Message	Description
[Blank]	Connection information to the Flexera Service Gateway is not entered on the **Flexera Service Gateway (FSG)** tab of the Application Manager Options dialog box.
`arl://MGS-APP-`*nnnnnnnnnn*	Indicates that a matching Flexera Identifier was found in the FlexNet Manager Suite Application Recognition Library (ARL).
`Flexera Identifier not found`	Indicates that a Flexera Identifier was not found in the FlexNet Manager Suite ARL.
`Multiple applications detected`	Indicates that because a Windows Installer package has multiple software tags, AdminStudio did not query FlexNet Manager Suite for a Flexera Identifier.

Flexera Service Gateway (FSG) Tab of Options Dialog Box / Import Wizard Messages

The following messages related to obtaining application Flexera Identifiers are listed on the **Flexera Service Gateway (FSG)** tab of the **Options** dialog box and in the messages displayed on the **Running the Import** panel of the Import Wizard:

Table 7-39 • Flexera Identifier Messages on Options Dialog Box / Import Wizard

Message	Description
`Flexera Identifier: arl://MGS-APP-00000034807`	Appears in the **Running the Import** panel of the Import Wizard when the Flexera Identifier for the imported package is successfully found.
`Error while fetching Flexera Identifier`	Appears in the **Running the Import** panel of the Import Wizard when one of the following occurs: • FlexNet Manager Suite is not registered with the Flexera Service Gateway. • FlexNet Manager Suite web service is not available. • Logged-in user does not have access to the FlexNet Manager Suite web service.
`Flexera Identifier has not been assigned for this application`	Appears in the **Running the Import** panel of the Import Wizard when a Flexera Identifier was not found in the FlexNet Manager Suite ARL for that application.

Table 7-39 • Flexera Identifier Messages on Options Dialog Box / Import Wizard

Message	Description
`Not synchronized with FlexNet Manager Platform`	Is displayed on the **Options** dialog box after you have upgraded an existing Application Catalog that was created using a version of AdminStudio prior to 11.5 SP2.

Note *• When you click the* ***FlexNet Manager Platform*** *button on the* ***Flexera Service Gateway (FSG)*** *tab of the Application Manager Options dialog box, AdminStudio first searches the ARL for an application's first Windows Installer package. If a Flexera Identifier is found, that ID is used. If a Flexera Identifier is not found for the Windows Installer package, AdminStudio then searches the ARL for the application's first App-V package, and if that is also not found, it searches for the application's first* **.** *exe file. If that is also not found, AdminStudio searches the ARL for the application's deployment type that was imported first.*

References Tab

On the **References** tab of the **Application View**, you can view a list of packages that are dependent upon this application or that supersede this application.

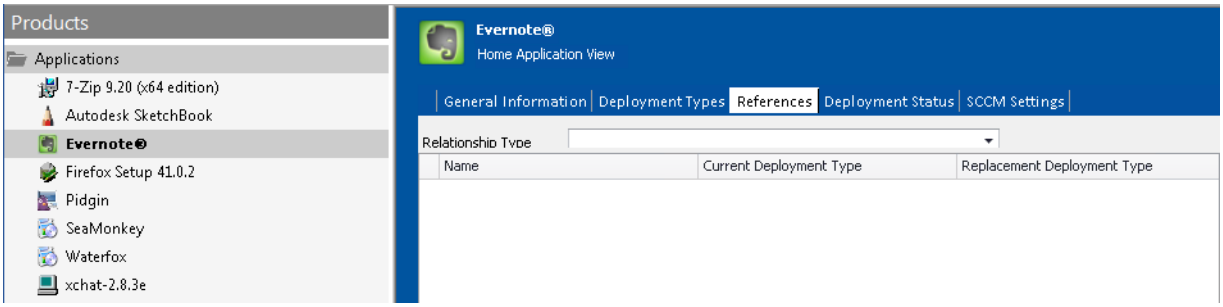


Figure 7-11: Application View / References Tab

These dependencies are defined on the **Dependencies** and **Supersedence** subtabs of the **Deployment Data** tab of the **Home Deployment Type View** for a selected package. If another package has specified that it is dependent upon or supersedent to this package, that package will be listed here.

For more information, see Specifying Package Dependencies Deployment Data and Specifying Package Supersedences Deployment Data.

The **References** tab of the **Application View** includes the following properties:

Table 7-40 • Application View / References Tab

Property	Description
Relationship Type	Select one of the following: • **Applications that supersede this application**—Select to view applications that contain a package which supersedes a package in this application. If both packages were installed on a target machine, the supersedent package would be used. • **Applications that depend on this application**—Select to view applications that contain a package which is dependent upon a package in this application. In order to run properly, the dependent package requires that the package that it is dependent upon be installed on the same target machine
Name	Name of application that contains a package that is dependent upon or supersedent to this application.
Current Deployment Type	Name of the deployment type of this application that is involved in this supersedence or dependency relationship.
Referring Deployment Type	Name of the deployment type of the referring application that is involved in this supersedence or dependency relationship.

Deployment Status Tab

The **Deployment Status** tab of the **Application View** lists data from ConfigMgr (Formerly called as System Center Configuration Manager) that is specific to this application, not to its deployment types. The data is read from the active ConfigMgr (Formerly called as System Center Configuration Manager) server that has been specified on the **Server Options > Distribution System** tab of the Application Manager Options dialog box.

Note *• If Application Manager is unable to establish an active link to the ConfigMgr (Formerly called as System Center Configuration Manager) server, then a message indicating that there is no active connection will be displayed.*

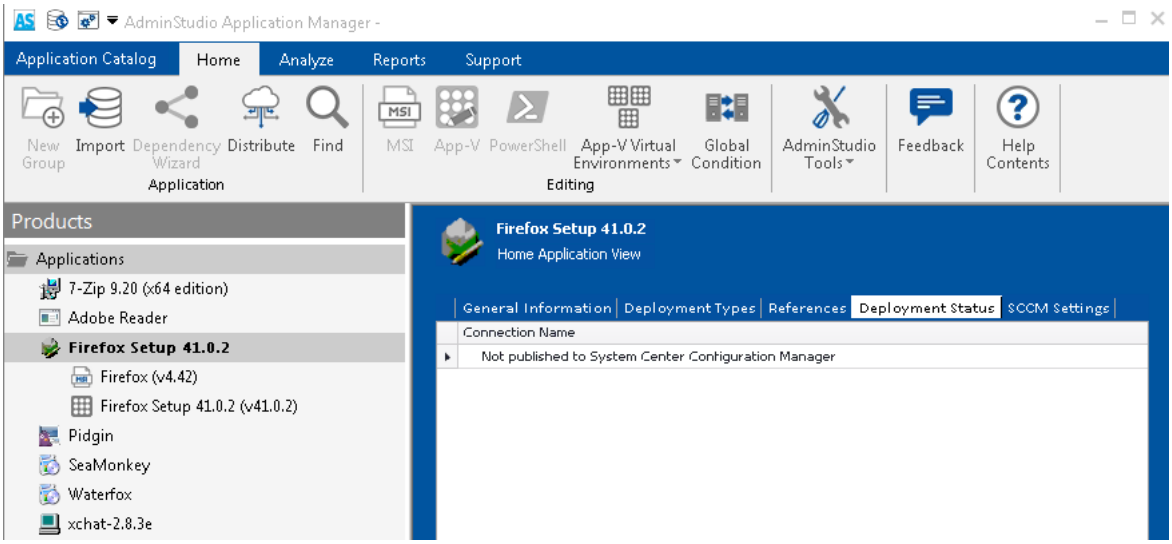


Figure 7-12: Application View / Deployment Status Tab

The **Deployment Status** tab of the **Application View** includes the following properties:

Table 7-41 • Application View / Deployment Status Tab

Property	Description
Server	Name of server where this application has been published.
Site Code	Site code of the ConfigMgr server where this application has been published.

Table 7-41 • Application View / Deployment Status Tab

Property	Description
Status	Indicates the status of the application on the System Center 2012 Configuration Manager server as **Active** or **Retired**. You can retire or reinstate an application in System Center 2012 Configuration Manager by changing this value, without even being required to republish the application. To change an application's status on System Center 2012 Configuration Manager select one of the following options: • **Retire**—Select this option to make this application unavailable for distribution by System Center 2012 Configuration Manager. • **Active**—Select this option to reinstate this application, making a formerly retired application once again available for distribution by System Center 2012 Configuration Manager. ***Note*** *• When you retire an application, it is no longer available for deployment but the application and any deployments of the application are not deleted. Existing copies of this application that have been installed on client computers will not be removed. If an application that has no deployments is retired, it will be deleted from the Configuration Manager console after 60 days. However, any installed copies of the application are not removed.* ***Note*** *• If the application has not been published to System Center 2012 Configuration Manager, the following message will be displayed:* `Not published to ConfigMgr (Formerly called as System Center Configuration Manager)`
Is Deployable	Indicates whether the application is ready to be deployed by ConfigMgr (Formerly called as System Center Configuration Manager). Values are **True** or **False**.
Is Deployed	Indicates whether the application has been deployed by ConfigMgr (Formerly called as System Center Configuration Manager). Values are **True** or **False**.
Deployments	Number of times this application has been deployed by ConfigMgr (Formerly called as System Center Configuration Manager).
Devices With App	Number of machines that this application has been successfully deployed to by ConfigMgr (Formerly called as System Center Configuration Manager).
Devices With Failure	Number of machines that ConfigMgr (Formerly called as System Center Configuration Manager) attempted to deploy this application to but was unsuccessful.
Users With App	Number of users this application has been successfully deployed to by ConfigMgr (Formerly called as System Center Configuration Manager).

Table 7-41 • Application View / Deployment Status Tab

Property	Description
Users With Failure	Number of users that ConfigMgr (Formerly called as System Center Configuration Manager) attempted to deploy this application to but was unsuccessful.
Revision	Revision of application.

ConfigMgr Settings Tab

The **ConfigMgr Settings** tab of the **Application View** lists an application's ConfigMgr (Formerly called as System Center Configuration Manager) settings. The **ConfigMgr Settings** tab is only displayed if you have set up a connection to a ConfigMgr (Formerly called as System Center Configuration Manager) server.

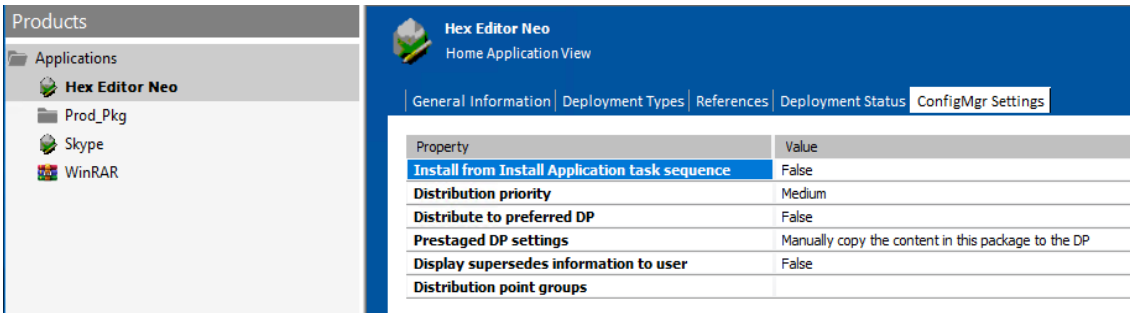


Figure 7-13: SCCM Settings Tab of Application View

The **SCCM Settings** tab includes the following properties:

Table 7-42 • SCCM Settings Tab

Properties	Description
Install from Install Application task sequence	Select **True** to deploy this application when deploying an operating system, as part of an Install Application task sequence. Select **False** to install this application manually.
Distribution priority	When you are sending multiple packages to a distribution point, those packages are sent in priority order, with higher priority packages being sent first. Use this property to specify a package's priority. The following options are available: • **High** • **Medium** • **Low**
Distribute to preferred DP	To enable on-demand content distribution to preferred distribution points, select **True**. When enabled, the content is distributed to all preferred distribution points in the list when a client requests the content for the package and the content is not available on any preferred distribution points.

Table 7-42 • SCCM Settings Tab

Properties	Description
Prestaged DP settings	Select one of the following options to specify how you want to distribute content to prestaged distribution points: • **Automatically download content when packages are assigned to DP**—Select to ignore the prestage settings and distribute content to the distribution point. • **Download only content changes to the DP**—Select to prestage the initial content to the distribution point, and then distribute content changes to the distribution point. • **Manually copy the content in this package to the DP**—Select to always prestage content on the distribution point. (Default)
Display supersedes information to user	Set this option to **True** to allow users to see deployments for this application and all applications that it supersedes in the Application Catalog. This may result in the user installing multiple applications on the same device, if the requirements for these applications are met.
Distribution point groups	Specify the System Center 2012 Configuration Manager distribution point groups to which this application's content will be distributed. ***Note*** • *If AdminStudio is integrated with App Portal, this is a required field. If no distribution point group is entered, the App Portal administrator will be required to manually enter this information in ConfigMgr (Formerly called as System Center Configuration Manager) before App Portal will be able to distribute this application.*

App Portal Information Tab

If AdminStudio and App Portal are integrated (by being connected to the same Flexera Service Gateway), the **App Portal Information** tab is displayed on the **Application View**.

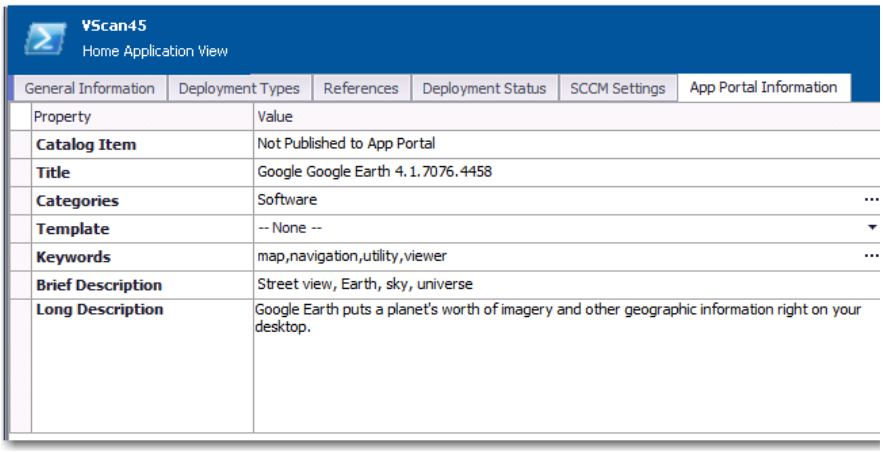


Figure 7-14: App Portal Information Tab

On this tab, you can control whether or not an App Portal catalog item is automatically created when you publish this application to System Center 2012 Configuration Manager or Symantec Altiris Server, and you can also specify the following App Portal catalog item information:

- Catalog Item Category
- Catalog Item Template
- Search Keywords
- Brief Description and Long Description

The **App Portal Information tab** includes the following properties:

Table 7-43 • App Portal Information Tab‘

Property	Description
Catalog Item	If published to App Portal, its catalog item number is listed. If it is not published to App Portal, the following is listed: `Not Published to App Portal`
Title	(Read only) This field is a concatenation of the application’s **Manufacturer** and **Software version** fields on the **General Information** tab with the application name in the Application Manager tree. This field simulates the same concatenation that App Portal will perform in order to fill in the **Title** field on the **General > Title & Description** tab of the App Portal **Catalog Item Properties** dialog box when this application is published to an App Portal-linked distribution system. ***Note*** *• The value in the App Portal* ***Title*** *field will identify the application in the App Portal storefront, as shown in Brief Description.*

Table 7-43 • App Portal Information Tab‘

Property	Description
Categories	(Required) Click the browse button in this field to open the **Categories** dialog box, where you can both indicate that you want to automatically create an App Portal catalog item when this application is published (by selecting the **Notify Flexera App Portal on publish of current Application** option), and specify the App Portal category or categories the catalog item will appear in. After a category is selected, it will be listed in this field. ***Note*** *• For more information, see Categories Dialog Box.*
Template	(Optional) In App Portal, you can use templates to automatically assign a defined set of properties to a catalog item. All of the templates defined in your installation of App Portal are listed in this field. Select a template from the list to apply it to the catalog item that will be created when this application is published to an App Portal-linked distribution system. ***Tip*** *• For catalog items that require a complex set of properties, it would be beneficial to create an App Portal template that contains all of those settings and properties. Then, whenever a new catalog item is created, properties and settings can be automatically loaded by selecting that template.*
Keywords	(Optional) In App Portal, end users can search for a catalog item by performing a search on the **Browse Home** tab. To assist in that search, keywords can be assigned to a catalog item. To assign keywords to the catalog item that will be created when this application is published to an App Portal-linked distribution system, click the browse button to open the **Keywords** dialog box, where you can create and assign keywords. ***Note*** *• For more information, see Keywords Dialog Box and Edit Keywords Dialog Box. Also see Specifying Catalog Item Keywords.*
Brief Description **Long Description**	(Optional) When you view an App Portal catalog item on the **Browse Home** tab, the catalog item’s **Title**, **Brief Description**, and **Full Description** properties are displayed. In these fields, enter the text that you want to use for the App Portal **Brief Description** and **Long (Full) Description** for this application. ***Note*** *• For an example of how the Brief and Long (Full) description is displayed in App Portal, see Creating a General Catalog Item in the App Portal Help Library.*

XenApp Deployment Types Tab

The **XenApp Deployment Types** tab of the **Application View** lists Citrix XenApp data for all of the application's deployment types. It contains the same information that is displayed on the XenApp Deployment Data Tab for each of its associated deployment types (packages).

Compressed View

The compressed view of the **XenApp Deployment Types** tab of the **Application View** lists the application's associated packages. You can view XenApp data for the application's App-V 4.x and Citrix XenApp packages.

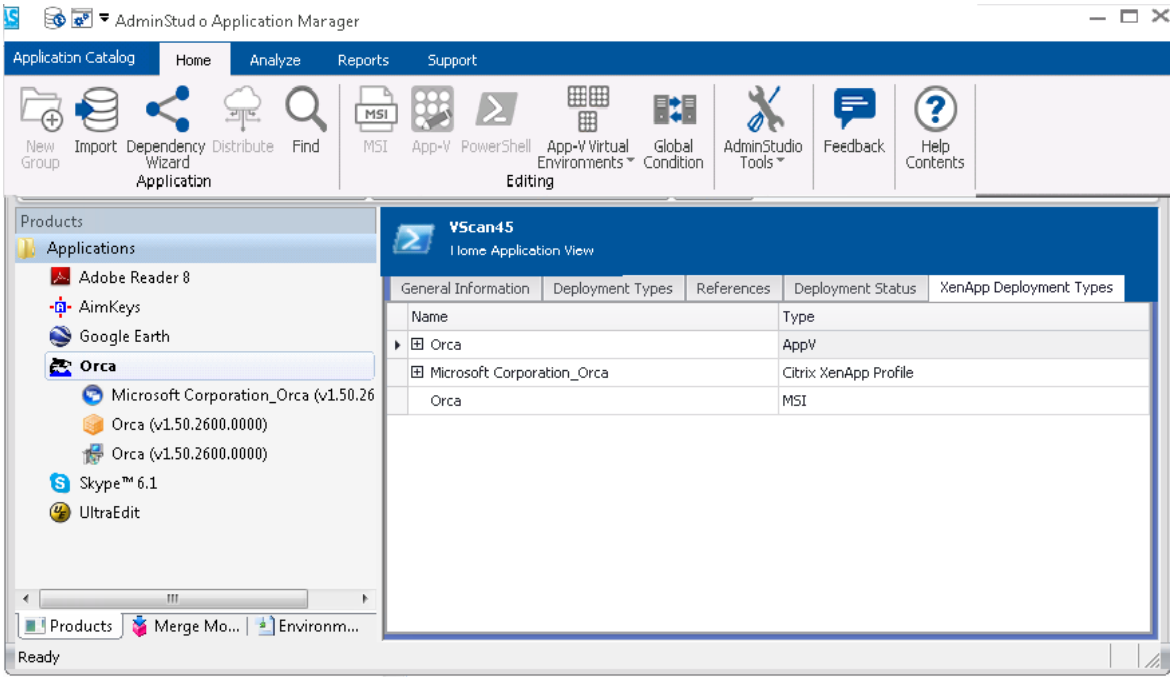


Figure 7-15: Application View / XenApp Deployment Types Tab (Compressed)

The compressed view of the **XenApp Deployment Types** tab of the **Application View** includes the following properties:

Table 7-44 • Application View / XenApp Deployment Types Tab (Compressed)

Property	Description
Name	Name of the package (deployment type). ***Note*** • *You can only view deployment data for an application's App-V 4.x and XenApp packages. The application's other deployment types will be listed on the* ***XenApp Deployment Types*** *tab, but no XenApp deployment information is displayed.*
Type	Indication of the package's deployment type: **MSI**, **AppV**, **Citrix**, **ThinApp**, or **Symantec**.

Expanded View

When you click the plus sign next to a package name, it expands to list the same deployment information that is displayed on the **XenApp Deployment Data** tab of the **Home Deployment Type View** for the selected package.

For a description of the properties displayed on these subtabs for each of an application's packages, see XenApp Deployment Data Tab.

Altiris Deployment Types Tab

The **Altiris Deployment Types** tab of the **Application View** list Altiris data for all of the application's deployment types. It contains the same information that is displayed on the Altiris Deployment Data Tab for each of its associated deployment types (packages).

Compressed View

The compressed view of the **Altiris Deployment Types** tab of the **Application View** lists the application's associated packages. You can view Altiris data for the application's Windows Installer, VMware ThinApp and legacy installer packages.

The compressed view of the **Altiris Deployment Types** tab of the **Application View** includes the following properties:

Table 7-45 • Application View / Altiris Deployment Types Tab (Compressed)

Property	Description
Name	Name of the package (deployment type). ***Note*** *• You can only view deployment data for an application's Windows Installer, VMware ThinApp and legacy installer packages. The application's other deployment types will be listed on the* ***Altiris Deployment Types*** *tab, but no Altiris deployment information is displayed.*
Type	Indication of the package's deployment type, such as: **MSI**, **AppV**, **Citrix XenApp Profile**, **ThinApp**, or **Symantec**.

Expanded View

When you click the plus sign next to a package name, it expands to list the same deployment information that is displayed on the **Altiris Deployment Data** tab of the **Home Deployment Type View** for the selected package.

For a description of the properties displayed on these subtabs for each of an application's packages, see Altiris Deployment Data Tab.

Workspace ONE Deployment Types Tab

The **Workspace ONE Deployment Types** tab of the **Application View** lists Workspace ONE data for all of the application's Apple iOS (local and public store), MSI, EXE, and Google Android (local and public store) packages. It contains the same information that is displayed on the Workspace ONE Deployment Data Tab for each of its associated iOS, MSI, EXE, and Android packages.

Compressed View

The compressed view of the **Workspace ONE Deployment Types** tab of the **Application View** lists the application's associated packages. You can view Workspace ONE data for the application's Apple iOS (local and public store), MSI, EXE, and Google Android (local and public store) packages.

The compressed view of the **Workspace ONE Deployment Types** tab of the **Application View** includes the following properties:

Table 7-46 • Application View / Workspace ONE Deployment Types Tab (Compressed)

Property	Description
Name	Name of the package (deployment type). ***Note*** *• You can only view deployment data for an application's Apple iOS (local and public store), MSI, EXE, and Google Android (local and public store) packages. If an application contains an iOS or Android package as well as a package of another deployment type, those other deployment types will be listed on the* ***Workspace ONE Deployment Types*** *tab, but no Workspace ONE deployment information will be displayed.*
Type	Indication of the package's deployment type, such as: **iOS App**, **iOS Public App**, **Android App**, **Android Public App**, **MSI**, **EXE**, **AppV**, **Citrix XenApp Profile**, **ThinApp**, or **Symantec**.

Expanded View

When you click the plus sign next to a package name, it expands to list the same deployment information that is displayed on the **Workspace ONE Deployment Data** tab of the **Home Deployment Type View** for the selected package.

For a description of the properties displayed on these subtabs for each of an application's packages, see Workspace ONE Deployment Data Tab.

Extended Attributes Tab

If you want to record custom data for applications, you can define custom extended attributes and display those attributes on the **Extended Attributes** tab of the **Application View**.

To enable the **Extended Attributes** tab of the **Application View**, you need to open a provided sample `ApplicationExtendedAttributes.SQL` script file, edit that script file to define your application attributes, and then run that SQL script on your Application Catalog.

Important *• For information on enabling the Extended Attributes tab and defining attributes, see Enabling Application Extended Attributes.*

After you edit and run that SQL script, the extended attributes that you have defined are listed on the **Extended Attributes** tab.

Important • *The **Extended Attributes** tab will only be visible for applications imported into the Application Catalog after the `ApplicationExtendedAttributes.SQL` script is run.*

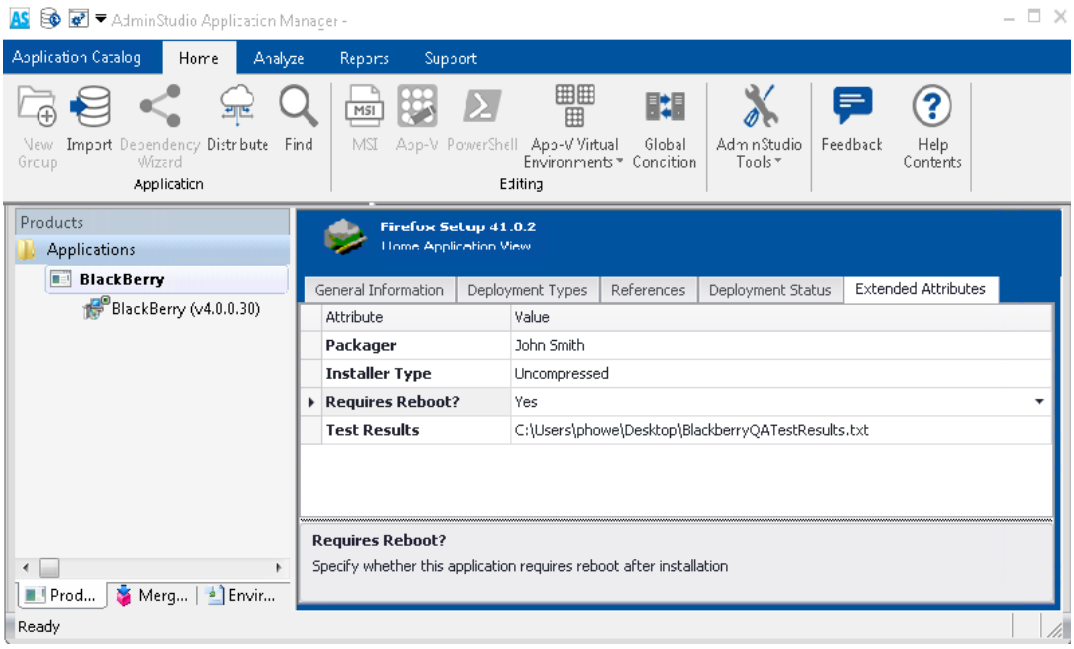


Figure 7-16: Application View / Extended Attributes Tab

Home Deployment Type View

When the **Home** tab is selected in the Application Manager ribbon and a package is selected in the tree, the **Home Deployment Type View** opens, which provides summary information about the package, its deployment data, and software ID tag information. The **Home Deployment Type View** presents this information on the following tabs:

- Package Information Tab
- Deployment Data Tab
- Bundled Packages Tab
- PKG Installer Choices Tab
- App-V Deployment Data Tab
- Casper Deployment Data Tab
- XenApp Deployment Data Tab
- Software Identification Tag Tab
- Altiris Deployment Data Tab
- Workspace ONE Deployment Data Tab
- Microsoft Intune Deployment Data Tab

Package Information Tab

When you select a package in the tree and select the **Package Information** tab in the **Home Deployment Type View**, details about that package.

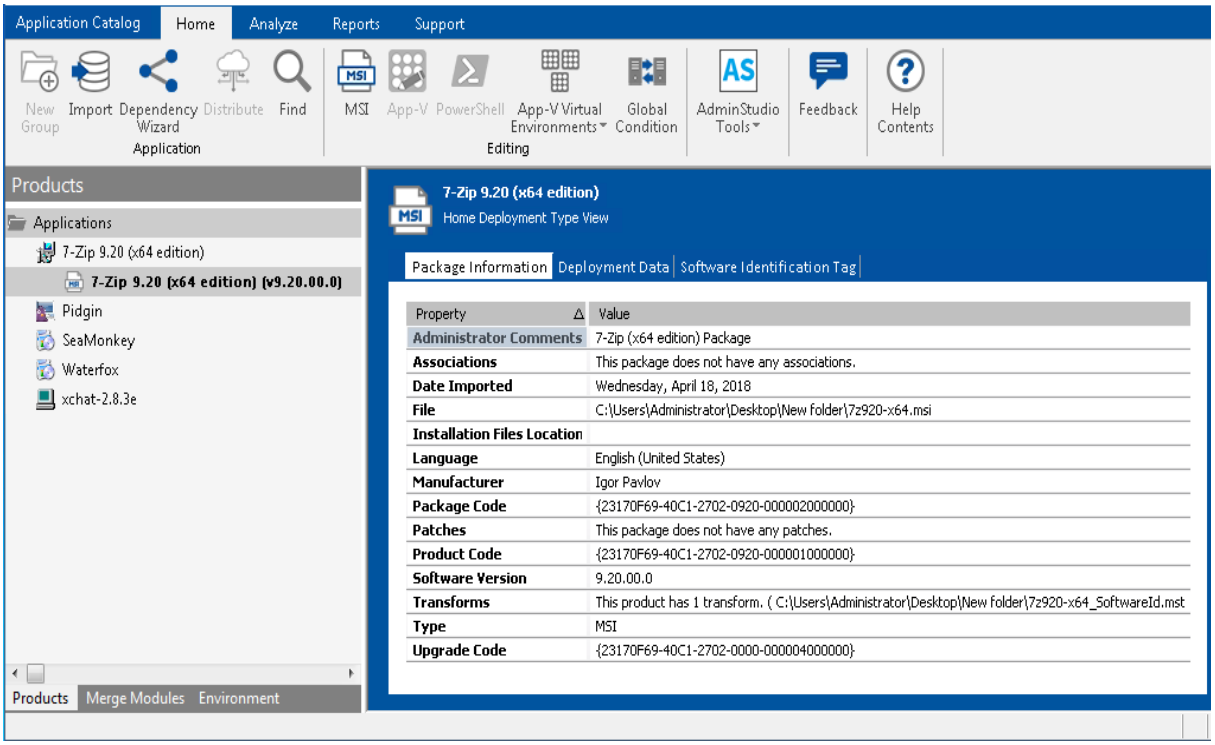


Figure 7-17: Home Deployment Type View / Package Information Tab

The **Package Information** tab of the **Home Deployment Type View** includes the following properties.

Table 7-47 • Home Deployment Type View / Package Information Tab

Field	Description
Type	Identifies the package's deployment type as one of the following: • Android App • Android Public App • Apple Mac Public App • Citrix XenApp Profile • iOS App • iOS Public App • Legacy Installer • Mac Installer Package • Microsoft App-V • MSI • MSIX • Universal Windows App - Desktop • Universal Windows App - Mobile • VMware ThinApp 4.x • Windows Store Public App • Windows RT App
Manufacturer	Manufacturer of the application, as discovered from its deployment types.
Administrator Comments	Comments related to this application, possibly regarding support for this application.
Software Version	Version of this package.
Language	Identifies the language of the intended target user of this package.
File	Identifies the location of this package. It can be either a hard-coded path or a UNC path. Ability to update the package location with the ellipses ... button. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository`
Date Imported	The date and time the package was imported.

Table 7-47 • Home Deployment Type View / Package Information Tab (cont.)

Field	Description
IsMobileApp	If this property is set to **True**, the Windows 8 **.appx** package is a desktop app. If it is set to **True**, it is a mobile app. ***Note*** *• Displayed for Windows 8 RT (runtime) apps only.*
Displayed Product Name	Lists the property that is mapped to the **File Name** property in Casper. ***Note*** *• Displayed for macOS DMG and PKG packages only.*
Genre(s)	Categories assigned to this application in the public store. ***Note*** *• Displayed for public store applications only.*
Domain **User Name** **Password**	If login credentials were entered on the panel of the Import Wizard, those credentials are listed here.
Package Id	Unique identifier that is associated with this App-V package. ***Note*** *• Displayed for App-V packages only.*
Version Id	Unique identifier that is associated with this version (revision) of the App-V package. ***Note*** *• Displayed for App-V packages only.*
Package Version	Package version number of the App-V package. ***Note*** *• Displayed for App-V packages only.*
Supported OS	Operating systems that this App-V package supports. ***Note*** *• Displayed for App-V packages only.*

Table 7-47 • Home Deployment Type View / Package Information Tab (cont.)

Field	Description
Primary Feature Block Size	Size of the App-V package's primary feature block, feature block 1. Feature block 1 must contain the core functionality of the App-V application that is necessary to launch the application. At application launch, all of the files in feature block 1 are streamed to the client in one unit. ***Note*** *• Displayed for App-V packages only.*
Total Size	Total size of this App-V application, including all feature blocks. ***Note*** *• Displayed for App-V packages only.*
Server URL Location	For App-V 4.x packages, the location on the App-V server from which this package can be streamed. ***Note*** *• Displayed for App-V 4.x packages only.*
Compressed	Indicates whether this App-V package is compressed. ***Note*** *• Displayed for App-V packages only.*
Client Version	Minimum version number of the App-V client that is required to use the App-V package. ***Note*** *• Displayed for App-V packages only.*
Package Code	The globally unique identifier (GUID) for the setup package. ***Note*** *• Displayed for Windows Installer, XenApp, ThinApp.*
Product Code	A string that uniquely identifies the product. ***Note*** *• Displayed for Windows Installer, XenApp, ThinApp.*
Upgrade Code	A string used to upgrade the application. The upgrade code for a package groups that package into a specific product family. ***Note*** *• Displayed for Windows Installer packages only.*

Table 7-47 • Home Deployment Type View / Package Information Tab (cont.)

Field	Description
Original File	When a user selects a Windows Installer package (`.msi`) along with one or more patch files (`.msp`) to import, AdminStudio first performs an administrative installation to merge the `.msi` and `.msp` files into one `.msi` file, and then imports the merged `.msi` file into the Application Catalog. In this instance, the **Original File** field lists the name and path of the original Windows Installer package that the patches were applied to, while the **File** field lists the name and path of the merged `.msi` file that was imported. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository` ***Note*** *• When a Windows Installer package that was imported without a patch is selected, the entry of the* ***Original File*** *and* ***File*** *fields is identical.*
Transforms	Lists the number of transforms associated with a Windows Installer package, and the transform file locations. ***Note*** *• Displayed for Windows Installer packages only.*
Patches	Lists the number of patches associated with a Windows Installer package, and the patch file locations. ***Note*** *• Displayed for Windows Installer packages only.*
Applications	Lists this package's associated applications. ***Note*** *• Displayed for Windows Installer, XenApp, ThinApp, and legacy packages.*

Table 7-47 • Home Deployment Type View / Package Information Tab (cont.)

Field	Description
Associations	Lists this package's associated packages: • **For virtual packages**, it lists the packages associated source Windows Installer package. • **For Windows Installer packages**, it lists its associated virtual packages. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package. In order for packages to be listed here, the package has to have already been imported into the Application Catalog and must be associated with the selected package. For more information, see Associating a Virtual Package with its Source Windows Installer Package. ***Note*** • *Displayed for Windows Installer, Citrix XenApp, and VMware ThinApp.*
Conflict Results	Date and time that conflict testing was last performed on this package. ***Note*** • *Displayed for Windows Installer packages only.*

Viewing Package Version History from the Home Deployment Type View

If you are connected to a Software Repository-enabled Application Catalog and you select a package that has more than one version, the **Version History** button in the **Content** tab of the Application Manager ribbon is enabled. If you click on this link, the **Package Versions** dialog box opens, listing all of the versions of the selected package.

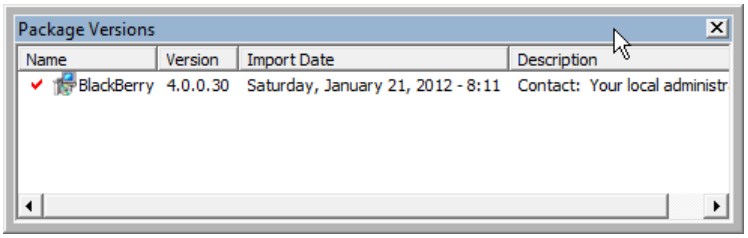


Figure 7-18: Package Versions Dialog Box

Note • *If the package and/or transforms are no longer in their original import directory, you can locate the file(s) from the provided hyperlink. You are also informed if the last modified date for the package in the Application Catalog does not match the last modified date of the package in its external location. You are given the opportunity to reimport the package to keep it synchronized in the Application Catalog.*

Deployment Data Tab

Note • *The **Deployment Data** tab does not apply to ThinApp applications, XenApp profiles.*

When a package is imported into the **Home** tab, Application Manager mines package elements for deployment data such as detection methods, dependencies, and requirements. You can view and modify this data and add new data by editing the properties on the subtabs of the **Deployment Data** tab and by using the easy-to-use wizards provided on the **Detection Methods**, **Requirements**, **Dependencies**, and **Supersedence** subtabs. AdminStudio displays deployment data for all of an application's packages (deployment types) in a multi-tabbed, organized format that is easy to navigate through and to update.

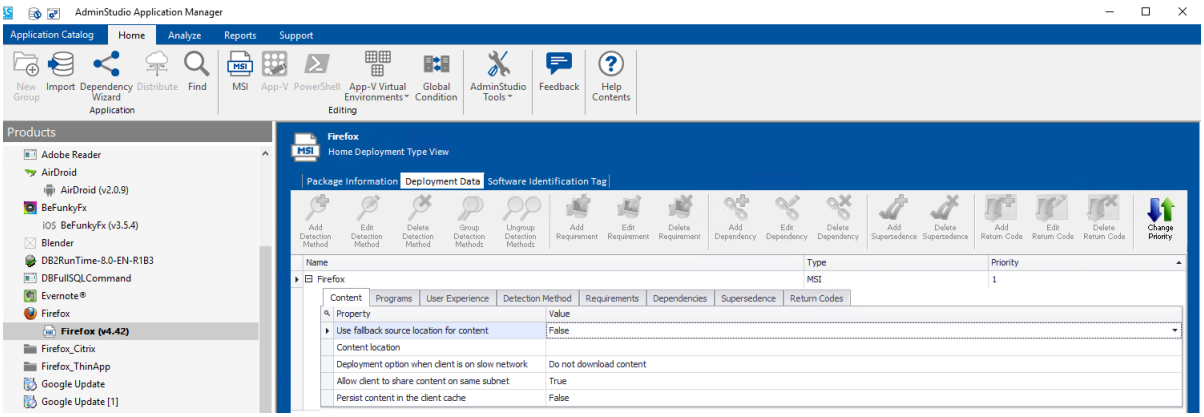


Figure 7-19: Home Deployment Type View / Deployment Data Tab

This deployment data is used by ConfigMgr (Formerly called as System Center Configuration Manager) when deploying packages. The data displayed on the **Deployment Data** tab of the **Home Deployment Type View** corresponds to the application model data stored for applications and packages in Microsoft System Center 2012 Configuration Manager. When packages are published from the Application Catalog to ConfigMgr (Formerly called as System Center Configuration Manager), this data is also published, which helps to ensure successful deployment.

The **Deployment Data** tab of the **Home Deployment Type View** has the following subtabs:

- Deployment Data Tab / Content Subtab
- Deployment Data Tab / Programs Subtab
- Deployment Data Tab / User Experience Subtab
- Deployment Data Tab / Detection Method Subtab
- Deployment Data Tab / Requirements Subtab
- Deployment Data Tab / Dependencies Subtab
- Deployment Data Tab / Supersedence Subtab
- Deployment Data Tab / Return Codes Subtab
- Deployment Data Tab / Detection Method AppX, MSIX Subtab
- Deployment Data Tab / Framework Subtab
- Deployment Data Tab / Virtual Environments Subtab

- Deployment Data Tab / Install Behavior Subtab
- Deployment Data Tab / Publishing Subtab
- XenApp Deployment Data Tab
- XenApp Deployment Data Tab / XenApp Information Subtab
- XenApp Deployment Data Tab / Advanced Settings Subtab

Deployment Data Tab / Content Subtab

Note • *The **Deployment Data** tab does not apply to ThinApp applications, XenApp profiles.*

The **Content** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists general information about package contents.

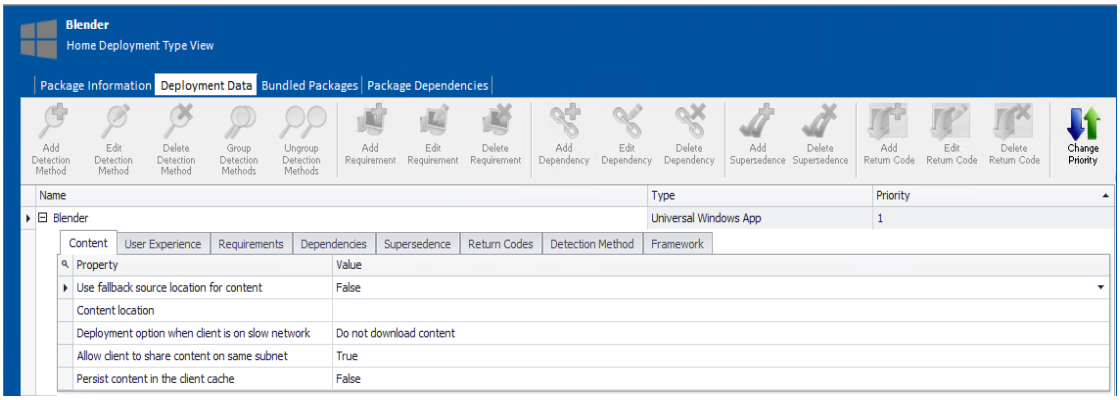


Figure 7-20: Deployment Data Tab / Content Subtab

The **Content** subtab of the **Deployment Data** tab includes the following deployment properties:

Table 7-48 • Deployment Data Tab / Content Subtab

Property	Description
Use fallback source location for content	To enable clients to “fall back” to using an unprotected distribution point if the package is not available on a protected (preferred) distribution point, set this option to **True**. By default, this option is set to **False**.

Table 7-48 • Deployment Data Tab / Content Subtab

Property	Description
Content location	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
Deployment option when client is on fast (LAN) network ***Note*** *• App-V packages only.*	Select one of the following options to specify how the client should download content when on a fast network: • **Download content from distribution point and run locally**—Select this option to download the content from the distribution point and run it locally. • **Stream content from distribution point**—Select this option for App-V packages to stream content from the distribution point.
Deployment option when client is on slow network	Select one of the following options to specify whether the client should download content when on a slow network: • **Do not download content**—When the client is connected within a slow or unreliable network boundary, do not download content. Select this option to save network bandwidth. (Default) • **Download content from distribution point and run locally**—Select this option if, when the client is connected within a slow or unreliable network boundary, you want it to download the content from the distribution point and run it locally. • **Stream content from distribution point**—Select this option to stream content from the distribution point. ***Note*** *• The **Stream content from distribution point** option is only available for App-V packages.*
Enable peer-to-peer content distribution ***Note*** *• App-V packages only.*	Select this option to reduce load on the network by allowing clients to download content from other clients on the network that have already downloaded and cached the content. This option utilizes Windows BrancheCache and can be used on computers running Windows Vista SP2 and later.

Table 7-48 • Deployment Data Tab / Content Subtab

Property	Description
Allow client to share content on same subnet ***Note*** *• MSI and EXE packages only.*	To reduce the load on the network by allowing clients to download content from other local clients on the network that have already downloaded and cached the content, select **True**.
Persist content in the client cache	To retain content in the cache on the client computer indefinitely even if it has already been run, select **True**. ***Note*** *• Setting this property to **True** will reduce the available cache space. This might cause a large deployment to fail at a later point if there is insufficient space available in the cache.*
Load content to App-V cache ***Note*** *• App-V packages only.*	Entire package (instead of just Feature Block 1) is loaded completely into the App-V cache prior to launch.
Uninstall content settings ***Note*** *• MSI and EXE packages only.*	Select one of the following options to define the uninstallation behavior: • **Same as install content** - Select this option when the installation content and Uninstallation content are same.(Default) • **No uninstall content** - Select this option when there is no uninstallation content • **Different from install content** - Select this option when uninstallation content and installation content are different.
Uninstall content location ***Note*** *• MSI and EXE packages only.*	Specify the network path that contains the uninstallation content.

Deployment Data Tab / Programs Subtab

Note *• The **Programs** subtab is only displayed for Windows Installer and legacy installer packages.*

The **Programs** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists command line parameters for package installation and uninstallation.

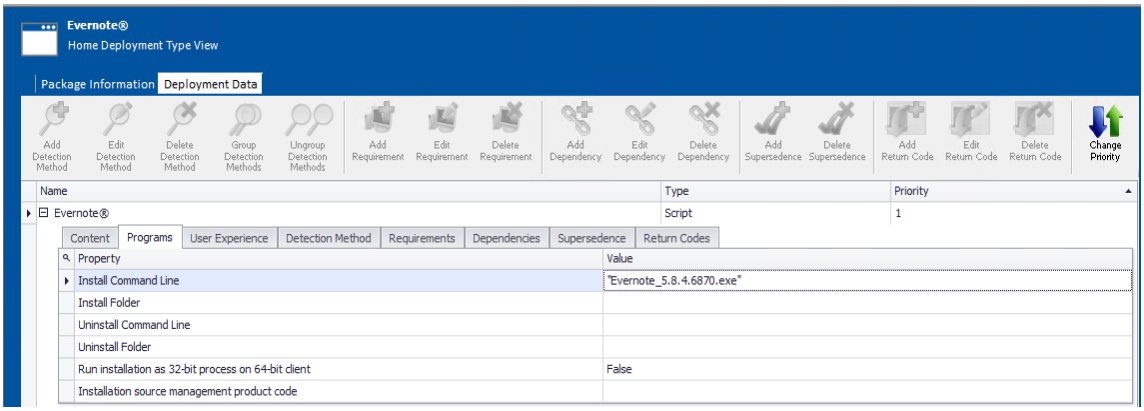


Figure 7-21: Deployment Data Tab / Programs Subtab

The **Programs** subtab of the **Deployment Data** tab includes the following deployment properties:

Table 7-49 • Deployment Data Tab / Programs Subtab

Property	Description
Install command line	Specify the command line that Configuration Manager will use to install this package on a client machine, including any required installation parameters.
Install folder	Specify the folder that contains the installation program for the deployment type. This folder can be an absolute path on the client or a path to the distribution point folder that contains the installation files. This field is optional.
Uninstall command line	Specify the command line that Configuration Manager will use to uninstall this package from a client machine, including any required parameters.
Uninstall folder	Specify the folder that contains the uninstall program for the deployment type. This folder can be an absolute path on the client or a path relative to the distribution point folder that contains the package. This field is optional.
Run installation as 32-bit process on 64-bit client	Select **True** to run the installation of this deployment type as a 32-bit process on a 64-bit client. To run it as a 64-bit process on a 64-bit client, select **False**.
Installation source management product code	To enable installation source management, enter the Windows Installer product code. ***Note*** • *In ConfigMgr (Formerly called as System Center Configuration Manager), installation source management enables a Windows Installer file to automatically be updated or repaired from content source files on an available distribution point.*

Table 7-49 • Deployment Data Tab / Programs Subtab

Property	Description
Repair command line	Specify the command line that Configuration Manager will use to repair this package from a client machine, including any required parameters.
Repair folder	Specify the folder that contains the repair program for the deployment type. This folder can be an absolute path on the client or a path relative to the distribution point folder that contains the package. This field is optional.

Deployment Data Tab / User Experience Subtab

Note *• The **User Experience** subtab is only displayed for Windows Installer packages, legacy installer packages, AppX, MSIX and Windows Store mobile apps.*

The **User Experience** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists parameters relating to the user experience during installation.

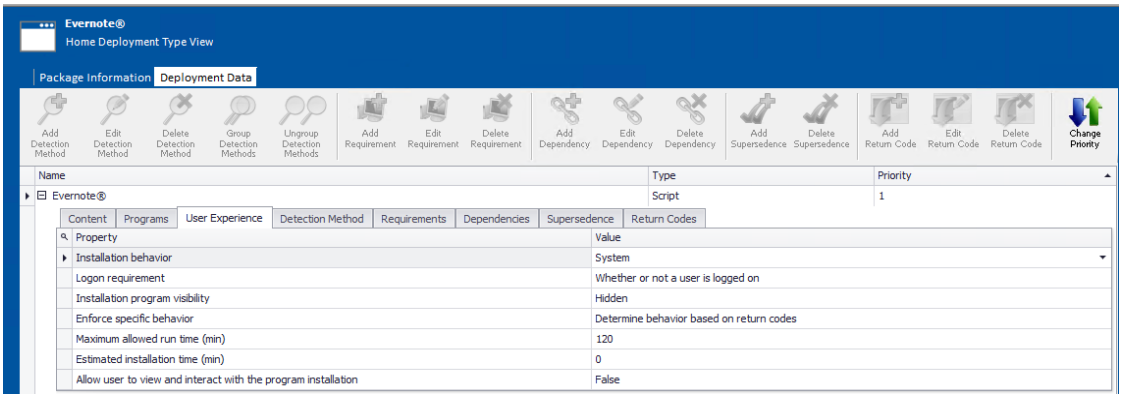


Figure 7-22: Deployment Data Tab / User Experience Subtab

The **User Experience** subtab of the **Deployment Data** tab includes the following deployment properties:

Table 7-50 • Deployment Data Tab / User Experience Subtab

Property	Description
Installation behavior	Select one of the following options: • **User**—The application installs for only the user who it is deployed to. • **System**—The application installs only once and is available to all users. • **Any**—If the application is deployed to a device, then it will install for all users. If the application is deployed to a user, then it will install for only that user.

Table 7-50 • Deployment Data Tab / User Experience Subtab

Property	Description
Logon requirement	Select one of the following options to specify the login requirements for installing this application: • **Only when a user is logged on** • **Whether or not a user is logged on** • **Only when no user is logged on** ***Note*** *• If you have set the* ***Installation behavior*** *property to* ***User****, this option will default to* ***Only when a user is logged on*** *and cannot be changed.*
Installation program visibility	Select one of the following options to specify the mode in which the deployment type will run on client devices: • **Maximized**—The deployment type runs maximized on client devices. Users will see all installation activity. • **Normal**—The deployment type runs in the normal mode based on system and program defaults. This is the default mode. • **Minimized**—The deployment type runs minimized on client devices. Users might see installation activity in the notification area or task bar. • **Hidden**—The deployment type runs hidden on client devices and users will see no installation activity.
Enforce specific behavior	Select one of the following options to enable Configuration Manager to enforce specific OS reboot behavior regardless of the application's intended behavior: • **Determine behavior based on return codes**—Handle reboots based on the codes configured on the Return Codes tab. • **No specific action**—No reboot required after installation. • **The software installation program might force a device restart**—Configuration Manager will not control reboot; the actual installation might force a reboot without warning. • **Configuration Manager client will force a mandatory device restart**—Configuration Manager will force a device reboot—either by notifying the user or without notification.

Table 7-50 • Deployment Data Tab / User Experience Subtab

Property	Description
Maximum allowed run time (min)	Specifies the maximum time (in minutes) that the program is expected to run on the client computer. This setting can be specified as a whole number greater than zero. The default setting is 120 minutes. This value is used for two purposes: • To monitor results from the deployment type. • To determine if a deployment type will be installed when maintenance windows have been defined on client devices.
Estimated installation time (min)	Specify the estimated time that the deployment type will take to install.
Allow user to view and interact with program installation	Set this property to **True** to enable the user to view and interact with the program installation in order to configure installation options. If it is set to **False**, the program installation is hidden from the user. ***Note*** *• This property can be set to* ***True*** *only when the* ***Login requirement*** *property is set to* ***Only when a user is logged on****.*
Provision this application for all users on the device	Set this property to **True** to give access to the windows app packages for all users on the device. If **false**, the access will be available only for the user it is deployed to. ***Note*** *• This property is applicable for AppX and MSIX packages.*

Deployment Data Tab / Detection Method Subtab

Note *• The* ***Detection Method*** *subtab is only displayed for Windows Installer and legacy installer packages.*

The **Detection Method** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists methods to detect whether this package is already installed on the target system.

- **Adding a detection method**—To add a detection method to the **Detection Method** subtab, click the **Add Detection Method** button in the ribbon toolbar to open the Detection Method Wizard.
- **Editing or deleting a detection method**—To modify an existing detection method, select the requirement and click **Edit Detection Method** in the ribbon toolbar. You can use **Delete Detection Method** to delete a detection method from the list.
- **Grouping detection method clauses together**—To group detection method clauses together, select the clauses and then click the **Group Detection Methods** button. When clauses are grouped, a left parentheses **(** is listed in the **Group Start** column before the first clause, and a right parentheses **)** is listed in the **Group End** column after the last clause. Use the **Connector** column to select an operator (**And** or **Or**) which specifies how to join multiple clauses.

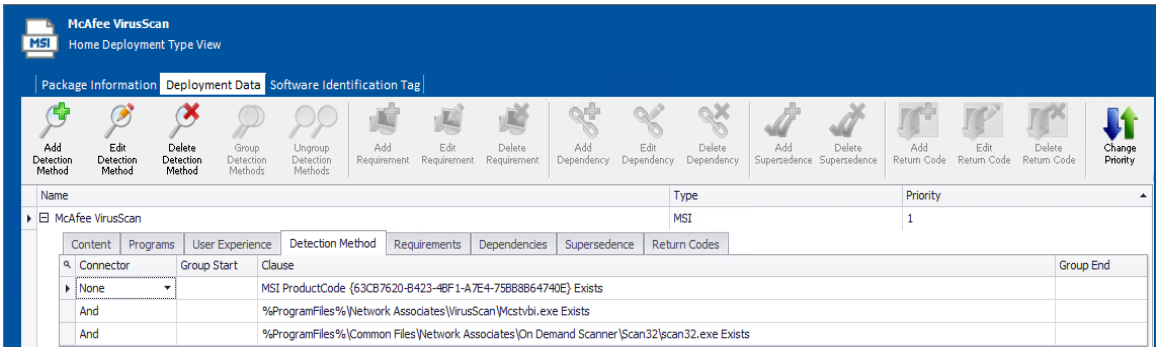


Figure 7-23: Deployment Data Tab / Detection Methods Subtab

The **Detection Method** subtab of the **Deployment Data** tab includes the following deployment properties:

Table 7-51 • Deployment Data Tab / Detection Method Subtab

Property	Description
Connector	Indicator of how the listed detection method clauses are connected. Options are: **None**, **And**, or **Or**.
Group Start	If this detection method clause is at the beginning of a clause group, a left parentheses **(** appears in this field.
Clause	List of defined clauses.
Group End	If this detection method clause is at the end of a clause group, a right parentheses **)** appears in this field.

Deployment Data Tab / Requirements Subtab

Note • *The **Requirements** subtab is displayed for Windows Installer, Windows Store, App-V, Apple iOS, MSI, Google Android, and legacy installer packages.*

You can use the **Requirements** subtab of the **Deployment Data** tab to add user or device requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package.

To add a requirement to the **Requirements** subtab, click the **Add Requirement** button the ribbon toolbar to open the Requirement Wizard. You can set device requirements, custom device requirements, and user and group requirements.

To modify an existing requirement, select the requirement and click **Edit Requirement** in the ribbon toolbar. You can use **Delete Requirement** to delete a requirement from the list.

The **Requirements** subtab of the **Deployment Data** tab includes the following properties:

Table 7-52 • Deployment Data Tab / Requirements Subtab

Property	Description
Condition	Lists the condition of the defined requirement.
Operator	Operator used in defined requirement.
Value	Value or values in defined requirement.
Category	Category type of defined requirement. Options are: • **Custom requirements** • **Device requirements** • **User requirements**
Drive	Drive specified in defined requirement.

Deployment Data Tab / Dependencies Subtab

Note *• The **Dependencies** subtab is displayed for Windows Installer, Windows Store, App-V, and legacy installer packages.*

You can use the **Dependencies** subtab to view or edit a list of other packages in the Application Catalog that must also be deployed by ConfigMgr (Formerly called as System Center Configuration Manager) with this package onto the target machine in order for this package to successfully operate.

To add a dependency to the **Dependencies** subtab, click **Add Dependency** in the ribbon toolbar to open the Dependency Wizard.

To modify an existing dependency, select the dependency and click **Edit Dependency** in the ribbon toolbar. You can use **Delete Dependency** to delete a dependency from the list.

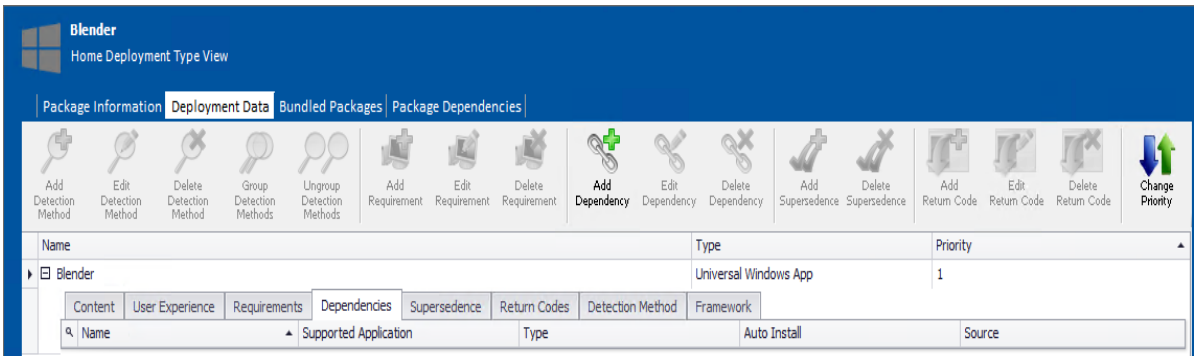


Figure 7-24: Deployment Data Tab / Dependencies Subtab

The **Dependencies** subtab of the **Deployment Data** tab includes the following deployment properties:

Table 7-53 • Deployment Data Tab / Dependencies Subtab

Property	Description
Name	Group name of dependencies.
Supported Application	Dependent application.
Type	Deployment type of dependent application.
Auto Install	Auto install setting.
Source	Location of dependent application.

Deployment Data Tab / Supersedence Subtab

Note *• The **Supersedence** subtab is displayed for Windows Installer, App-V, Apple iOS, MSI, Windows Store, Google Android, and legacy installer packages.*

You can use the **Supersedence** subtab to view or edit a list of other packages that this package would supersede if installed on the same target machine (meaning that the package on the target system would need to be uninstalled prior to installing this package).

- **Adding a supersedent application**—To add a supersedent application to the **Supersedence** subtab, click **Add Supersedence** in the ribbon toolbar to open the Supersedence Wizard.
- **Editing or deleting a supersedent application**—To modify an existing supersedence, select the supersedence and click **Edit Supersedence** in the ribbon toolbar. You can use **Delete Supersedence** to delete a supersedent application from the list.

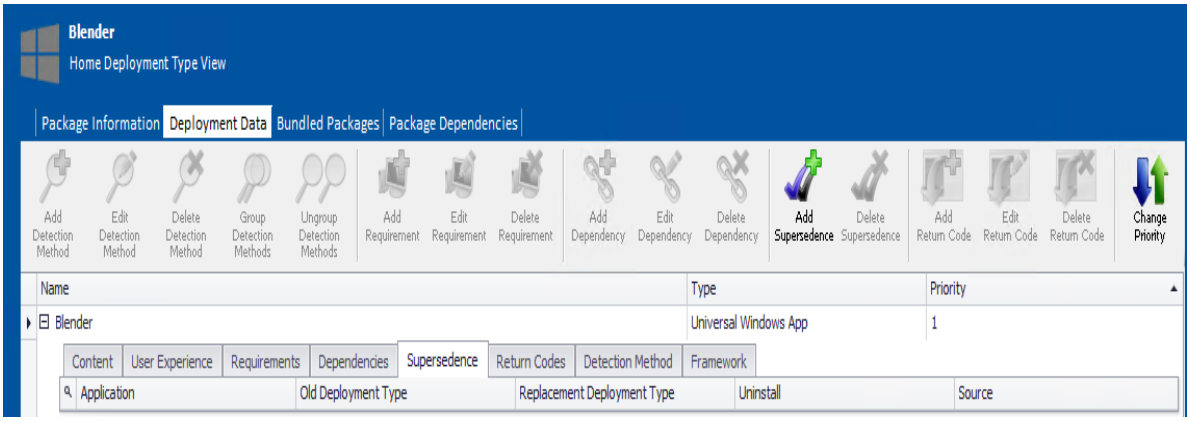


Figure 7-25: Deployment Data Tab / Supersedence Subtab

The **Supersedence** subtab of the **Deployment Data** tab includes the following properties:

Table 7-54 • Deployment Data Tab / Supersedence Subtab

Property	Description
Application	Name of supersedent application.
Old Deployment Type	Name of original deployment type.
Replacement Deployment Type	Name of replacement deployment type.
Uninstall	Uninstall setting.
Source	Location of application.

Deployment Data Tab / Return Codes Subtab

Note *• The **Return Codes** subtab is displayed for Windows Installer, App-V, Windows Store, and legacy installer packages.*

You can view and edit a MSI and EXE package's return codes on the **Return Codes** subtab of the **Deployment Types** tab. Return codes are used to indicate whether a restart is required, whether an installation is complete, and to customize the text shown to users when a specific code is returned.

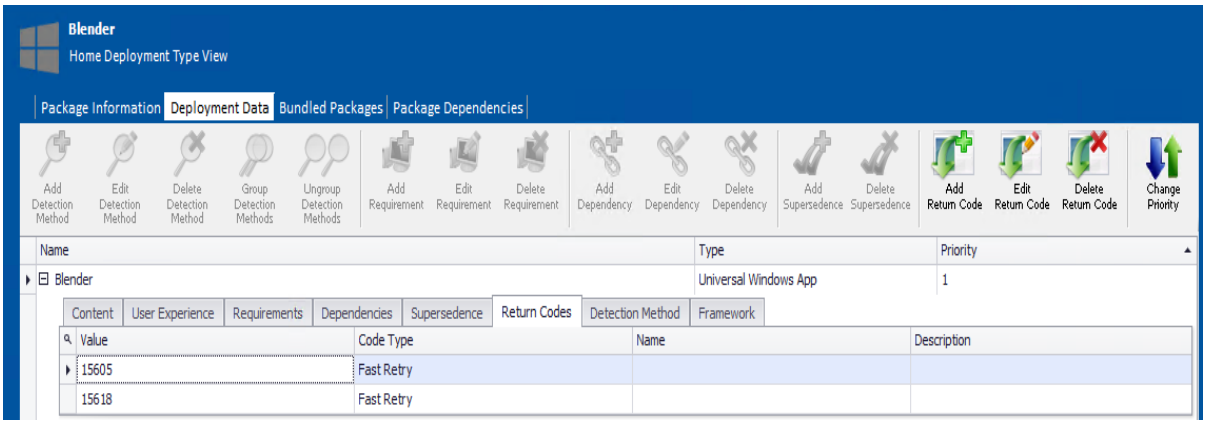


Figure 7-26: Return Codes Subtab of Deployment Types tab

The following return codes are populated by default during package import:

- **0**—Success (no reboot)
- **1707**—Success (no reboot)
- **3010**—Soft Reboot
- **1641**—Hard Reboot
- **1618**—Fast Retry

On the **Return Codes** tab, you can add, edit, and delete return codes.

- **Adding a return code**—Click **Add Return Code** in the ribbon and define a new return code on the **Add Return Code** dialog box.
- **Editing a return code**—Select a return code, click **Edit Return Code** in the ribbon, and edit the details of the return code on the **Edit Return Code** dialog box. However, the **Return Code Value** field cannot be edited.
- **Deleting a return code**—Select a return code, click **Delete Return Code** in the ribbon, and confirm the deletion.

Deployment Data Tab / Detection Method AppX, MSIX Subtab

Note • *The **Detection Method AppX,MSIX** subtab is displayed for Windows Store mobile apps and MSIX packages.*

The defined detection methods for Microsoft MSIX packages(.msix), Microsoft UWP app packages (.appx) and Windows 8 packages (.appx) are listed on the **Detection Method AppX** subtab of the **Deployment Types** tab on the **Home Deployment Type View**. Detection methods are used to detect whether this package is already installed on the target system.

The **Detection Method AppX,MSIX** subtab of the **Deployment Data** tab includes the following properties:

Table 7-55 • Deployment Data Tab / Detection Method AppX, MSIX Subtab

Property	Description
Name	Name of the detection method.
Publisher	Publisher of the detection method.
Version	Version of the detection method.
Resource Id	Resource ID of the detection method.
Processor architecture	Type of processor architecture of the detection method.

Deployment Data Tab / Framework Subtab

Note • *The **Framework** subtab is displayed for Windows Store mobile apps.*

When you have a Windows Store mobile app selected in the Application Manager tree, the **Framework** subtab of the **Deployment Data** tab of the **Home Deployment Type View** is displayed, and displays any customizations that may have been added to this Windows Store mobile app.

Windows Store mobile app developers can use the application framework to customize a mobile app. With the framework, they can create a task or an extension to customize the application. They can extend existing functions within the application or embed new functionality with custom business logic.

If the selected Windows Store mobile app has any application framework customizations, they will be listed on the **Framework** subtab.

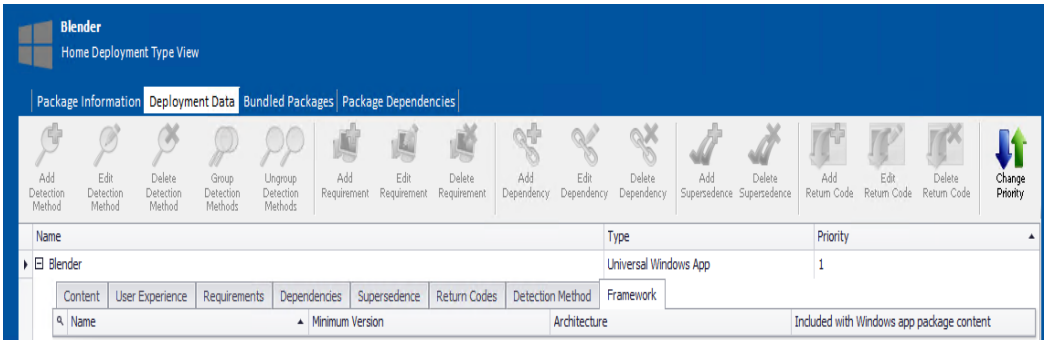


Figure 7-27: Framework Subtab of Deployment Types tab

The **Framework** subtab of the **Deployment Data** tab includes a list of application framework customizations that have been included with this Windows Store mobile app. For each item, the following information is listed:

Table 7-56 • Deployment Data Tab / Framework Subtab

Property	Description
Name	Name of framework item.
Minimum Version	Minimum version of framework item.
Architecture	Type of architecture of framework item.
Included with Windows app package	Indicates whether the framework item is included with the Windows Store mobile app.

Deployment Data Tab / Virtual Environments Subtab

In Application Manager, you can create App-V virtual environments for App-V 5.0 packages. App-V virtual environments in Microsoft System Center 2012 Configuration Manager enable deployed virtual applications to share the same file system and registry on client computers. This means that unlike standard virtual applications, these applications can share data with each other.

Tip • *Using virtual environments to group dependent packages together in App-V 5.0 is similar to the Dynamic Suite Composition feature used with App-V 4.x packages.*

Virtual environments are created or modified on client computers when the application is installed or when clients next evaluate their installed applications. You can order these applications so that when multiple applications attempt to modify the same file system or registry value on a client computer, the application with the highest order takes precedence.

For information on creating a virtual environment, see Creating an App-V Server Virtual Environment.

You can view an App-V 5.0 package's virtual environments on the **Virtual Environments** subtab of the **Deployment Types** tab.

Deployment Data Tab / Install Behavior Subtab

Note • *The **Install Behavior** subtab is only displayed for Windows Installer and legacy installer packages.*

The **Install Behavior** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists methods to add an executable file that must be closed on the target system, before installing the current application.

- **Adding a Install Behavior**—To add an executable file which must be closed to the **Install Behavior** subtab, click the **Add Install Behavior** button in the ribbon toolbar to open the **Install Behavior** wizard.
- **Editing or deleting a Install Behavior** —To modify an existing executable file, select the file and click **Edit Install Behavior** in the ribbon toolbar. You can use **Delete Install Behavior** to delete an executable file from the list.

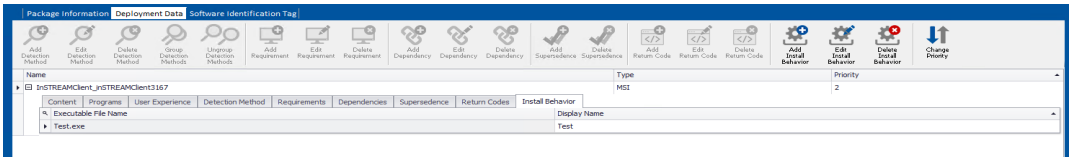


Figure 7-28: Deployment Data Tab / Install Behavior Subtab

The **Install Behavior** subtab of the **Deployment Data** tab includes the following properties:

Table 7-57 • Deployment Data Tab / Install Behavior Subtab

Property	Description
Executable File Name	Name of an executable file with the file extension(.exe or .msi).
Display Name	Name of an executable file to be displayed.

Deployment Data Tab / Publishing Subtab

Note • *The **Publishing** subtab is only displayed for App-V and SFT packages.*

The **Publishing** subtab of the **Deployment Data** tab of the **Home Deployment Type View** allows admin to select and unselect the shortcut components of the application to be deployed. Selected shortcuts will be created on the target system after successful deployment.

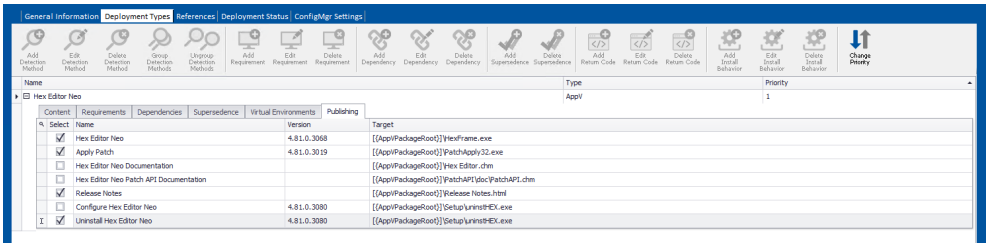


Figure 7-29: Deployment Data Tab / Publishing Subtab

The **Publishing** subtab of the **Deployment Data** tab includes the following properties:

Table 7-58 • Deployment Data Tab / Install Behavior Subtab

Property	Description
Select	Select or Unselect checkbox.
Name	Name of the shortcut components.
Version	Version of the shortcut components.
Target	Associated executable files.

Bundled Packages Tab

The **Bundled Packages** tab of the **Home Deployment Type View** lists the child packages that are bundled in Windows `.exe` installers, PowerShell wrapped package `.ps1` files, .msix packages, and macOS `.dmg` and `.pkg` installers.

- Bundled Packages of Complex Installer Executable Files
- Bundled Packages of macOS .pkg and .dmg Files

Bundled Packages of Complex Installer Executable Files

You can import complex installer executable files that contain bundled Windows Installer packages into the Application Catalog. There are multiple installation executable types that can contain embedded Windows Installer packages, including the following:

- InstallShield InstallScript `.exe` files
- InstallShield Basic MSI installers that are compressed into a `setup.exe` file
- InstallShield Suite Installer `.exe` files
- Wise Package Studio `.exe` files
- PowerShell wrapped package `.ps1` files
- Other executable file types that can be uncompressed by 7-ZIP

After these complex installer executables have been imported, you can view a list of the child `.msi` packages bundled within them on the **Bundled Packages** tab of the **Home Deployment Type View**.

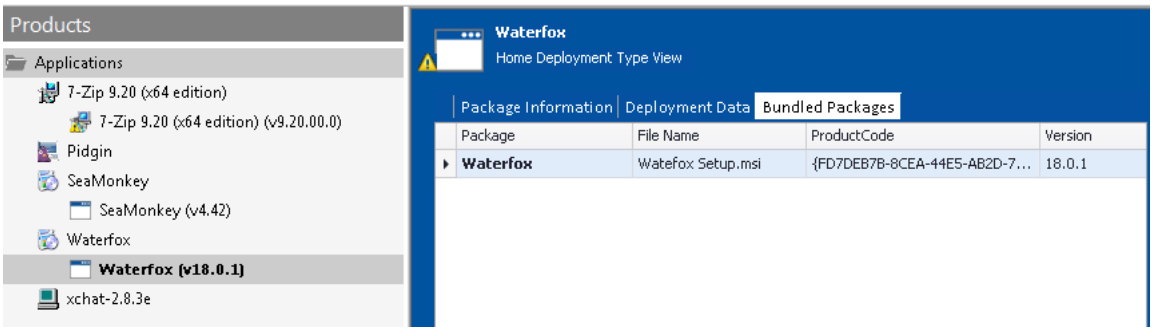


Figure 7-30: Bundled Packages Tab of Home Deployment Type View

When inspecting these child `.msi` packages, Application Manager extracts the information about each package, such as product name and version number. This makes it much more likely that Application Manager will be able to assign a Flexera Identifier to these applications.

You can perform operating system compatibility, application virtualization compatibility, and best practices testing on these bundled packages, and the test results will be combined. For more information, see Viewing Combined Test Results of Bundled Packages.

Note • *AdminStudio will only inspect complex installer (.exe or .ps1) files one level deep. If a complex installer file contains another complex installer file bundled within it, that child file will not be inspected.*

Bundled Packages of macOS .pkg and .dmg Files

If an Apple installer package (`.pkg`) or disk image (`.dmg`) contains child packages bundled within it, those child packages will be listed on the **Bundled Packages** tab of the **Home Deployment Type View**.

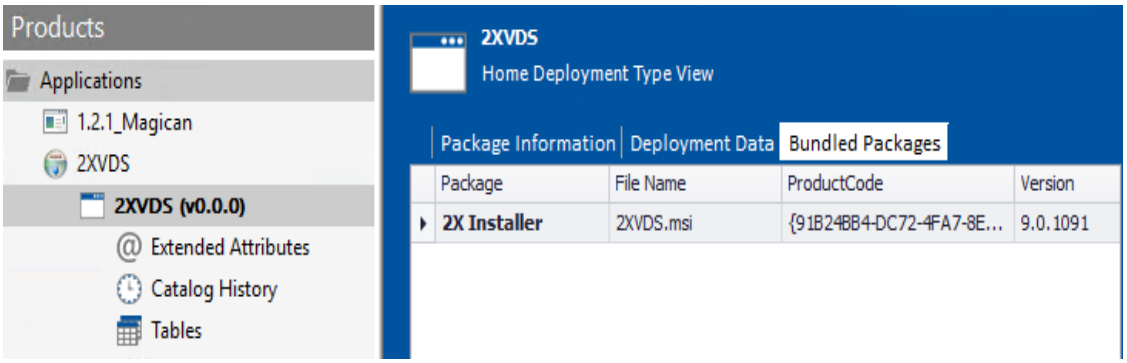


Figure 8: Bundled Packages Tab of Home Deployment Type View for Apple Installer Package (.pkg)

When the imported `.dmg` or `.pkg` file is tested, each of these child packages is tested, and the combined test results are listed in Analyze, as described in Viewing Combined Test Results of Child Applications of PKG and DMG Installers.

Note • *AdminStudio will only inspect macOS package files one level deep. If a .dmg or .pkg package contains another .dmg or .pkg package bundled within it, that child package will not be inspected.*

PKG Installer Choices Tab

Just as a Windows Installer package can be customized by adding a transform file, an Apple installer package (.pkg) can be customized by editing an XML file that is embedded within it. The settings defined in the embedded XML file are displayed on the **PKG Installer Choices** tab of the package's **Home Deployment Type View**.

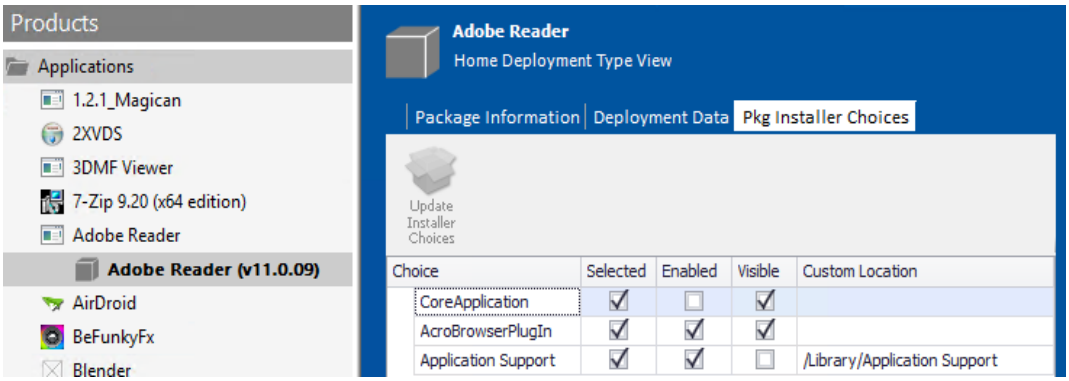


Figure 9: PKG Installer Choices Tab of Home Deployment Type View for Mac PKG Installer

The **PKG Installer Choices** tab lists all settings that have been defined in the embedded XML settings file by the application manufacturer. To customize this installer (such as to prepare it for silent installation by Casper), you can make changes to the settings on this tab and then click **Update Installer Choices**. AdminStudio will then save your changes in the package's embedded settings file.

For each installer **Choice** listed on the **PKG Installer Choices** tab, the following options are available:

Table 8 • PKG Installer Choices Tab

Option	Description
Visible	This option can be either selected or not selected: • **Selected**—This choice setting will be displayed in the installer. • **Not selected**—This choice setting will not be displayed in the installer.
Selected	This option can be either selected or not selected: • **Selected**—If this choice setting is displayed in the installer, its check box will be checked. • **Not selected**—If this choice setting is displayed in the installer, its check box will not be checked.
Enabled	This option can be either selected or not selected: • **Selected**—If this choice setting is displayed in the installer, it will be enabled. • **Not selected**—If this choice setting is displayed in the installer, it will be disabled.
Custom Location	If this choice setting explicitly permits the user to specify a user-defined installation path, the path entered in this field would populate the user-defined installation path when it is displayed in the installer.

Note • *Modifying the installer choices of an Apple installer package does not affect the digital signature of the package.*

App-V Deployment Data Tab

Note • *Because Microsoft App-V server only supports App-V 5.0 packages, the* ***App-V Deployment Data*** *subtab is only displayed for App-V 5.0 packages.*

When an App-V 5.0 package is imported into the Application Catalog, Application Manager mines package elements for Microsoft App-V-specific deployment data. You can view and modify data for App-V 5.0 packages and add new data by editing the properties on the subtabs of the **App-V Deployment Data** tab. AdminStudio displays App-V deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

- App-V Information Subtab
- Advanced Settings Subtab

App-V Information Subtab

The **App-V Information** subtab of the **App-V Deployment Data** tab includes the following properties:

Table 7-1 • App-V Deployment Data Tab / App-V Information Subtab

Property	Description
Grant Access	Enter the name of the Active Directory security group that will have access to the package. Enter the group name in `domain_name\group_name` format. To specify more than one group, enter the values in comma-separated format: `asdev\group1,asdev\group2`
Publish Package for Client	Set this property to **True** to publish this package so that it is available to users running the App-V client.

Advanced Settings Subtab

The **Advanced Settings** subtab of the **App-V Deployment Data** tab includes the following properties:

Table 7-2 • Advanced Settings Tab / App-V Information Subtab

Property	Description
Dynamic Deployment Configuration Path	Virtual application packages contain a manifest that provides all the core information for the package, including the defaults for the package settings. However, you can use XML-based dynamic configuration files to customize App-V 5.0 packages. This provides a more convenient method for package customization by removing the need to re-sequence packages using the desired settings, and provides a way to keep package content and custom settings independent, similar to the way that you can customize a Windows Installer package using a transform file (`.mst`). You can create a dynamic *deployment* configuration file to specify the default settings for this App-V package for all users. Enter the path to a dynamic deployment configuration file in the **Dynamic Deployment Configuration Path** field. This file will be used to override the default behavior provided in the package's manifest. If you do not specify a path to a dynamic deployment configuration file, the App-V agent will deploy the package with the default behavior provided in the package's manifest. ***Note*** • *Only one dynamic deployment configuration file can be entitled to a package.* ***Note*** • *If you specify both a dynamic **deployment** configuration file and a dynamic **user** configuration file, both will be applied: first the dynamic **deployment** configuration file and then the dynamic **user** configuration file.*

Table 7-2 • Advanced Settings Tab / App-V Information Subtab

Property	Description
Dynamic User Configuration Path	Virtual application packages contain a manifest that provides all the core information for the package, including the defaults for the package settings. However, you can use XML-based dynamic configuration files to customize App-V 5.0 packages. This provides a more convenient method for package customization by removing the need to re-sequence packages using the desired settings, and provides a way to keep package content and custom settings independent, similar to the way that you can customize a Windows Installer package using a transform file (`.mst`). In addition to being able to create a dynamic **deployment** configuration file to specify the default settings for this App-V package for all users, you can also create a dynamic **user** configuration file to customize these settings for *specified groups of users*. Enter the path to a dynamic user configuration file in the **Dynamic User Configuration Path** field. If you do not specify a path to a dynamic user configuration file, the App-V agent will deploy the package with the behavior provided by a dynamic *deployment* configuration file, if specified, or the default behavior provided in the package's manifest. ***Important*** *• If you specify a dynamic user configuration file, but do not enter a user group in the User Configuration Grant Access field, then that dynamic user configuration file will not be applied to any users.* ***Note*** *• Only one dynamic user configuration file can be entitled to a package.* ***Note*** *• If you specify both a dynamic **deployment** configuration file and a dynamic **user** configuration file, both will be applied: first the dynamic **deployment** configuration file and then the dynamic **user** configuration file.*
User Configuration Grant Access	If you want to apply a dynamic user configuration file to a specific group of users, enter that group here in `domain\groupname` format. I To specify more than one group, enter the values in comma-separated format: asdev\group1,asdev\group2 ***Important*** *• If you specify a dynamic user configuration file in the Dynamic User Configuration Path field, but do not enter a user group in this field, then that dynamic user configuration file will not be applied to any users.*

Table 7-2 • Advanced Settings Tab / App-V Information Subtab

Property	Description
Connection Group	Lists the connection groups that this package has been added to. ***Note*** *• This is a read-only field that lists the connection groups that this package has been added to, as described in Creating an App-V Server Virtual Environment.*

Casper Deployment Data Tab

Note *• Because Casper only supports macOS desktop packages, the* ***Casper Deployment Data*** *subtab is only displayed for* `.pkg` *files,* `.dmg` *files, and links to Apple Mac App Store apps.*

When a macOS desktop package is imported into the Application Catalog, Application Manager mines package elements for Casper-specific deployment data. You can view and modify deployment data for macOS desktop packages and add new data by editing the properties on the subtabs of the **Casper Deployment Data** tab. AdminStudio displays Casper deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

The **Casper Deployment Data** subtab of the **Home Deployment Type View** can include up to three subtabs that display Casper deployment data: **General**, **Options**, and **Limitations**. The **Options** and **Limitations** subtabs are not displayed for Mac App Store apps.

- General Subtab
- Options Subtab
- Limitations Subtab

General Subtab

The **General** subtab of the **Casper Deployment Data** tab is displayed for all macOS desktop applications.

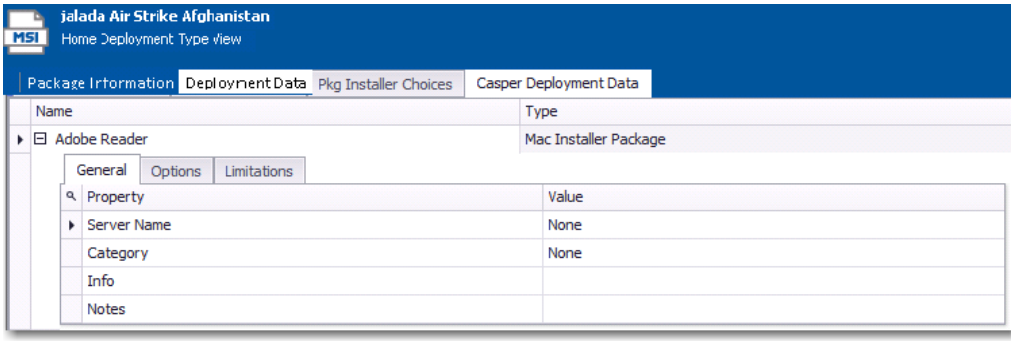


Figure 7-1: Casper Deployment Data / General Subtab

The **General** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-3 • Casper Deployment Data Tab / General Subtab

Property	Description
Server Name	Name of the Casper server.
Category	Category in Casper that the package will be added to. ***Note*** *• Casper lets you create custom categories. If AdminStudio has matched this application to an entry in the Application Recognition Library (ARL), AdminStudio will use the ARL category when publishing to Casper, creating it if necessary.*
Info	Information to display to the administrator when the package is deployed or uninstalled
Notes	Notes to display about the package (such as the name of the person who built it and when it was built). ***Note*** *• Not displayed for Mac App Store Apps.*
Free	Indicates whether or not the Mac App Store app is available for free (**True**) or whether it requires payment (**False**). ***Note*** *• Only displayed for Mac App Store Apps.*

Options Subtab

The **Options** subtab of the **Casper Deployment Data** tab is only displayed for PKG and DMG packages.

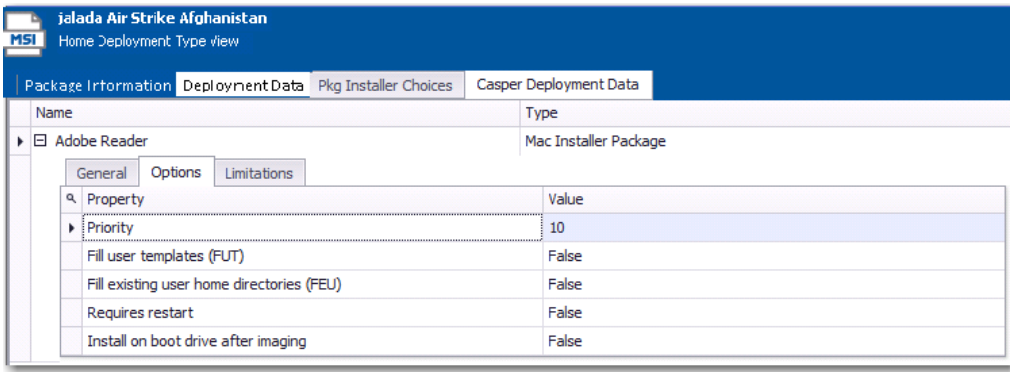


Figure 7-2: Casper Deployment Data / Options Subtab

The **Options** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-4 • Casper Deployment Data Tab / Options Subtab

Property	Description
Priority	Priority to use for deploying or uninstalling the package. For example, a package with a priority of **1** is deployed or uninstalled before other packages. When several applications are deployed together, the one with the highest priority is installed first. Therefore, if one application requires that another application be installed first before it can be successfully installed, you should assign the required application a higher priority (lower number) than the dependent application.
Fill user templates (FUT)	Set this property to **True** to fill new home directories with the contents of the home directory in the package's Users folder. This setting can be changed when deploying or uninstalling the package using a policy.l ***Note*** • *Only applicable to DMG packages.*
Fill existing user home directories (FEU)	Set this property to **True** to fill existing home directories with the contents of the home directory in the package's **Users** folder. This setting can be changed when deploying or uninstalling the package using a policy ***Note*** • *Only applicable to DMG packages.*
Requires restart	Set this property to **True** to require that computers must be restarted after installing the package.
Install on boot drive after imaging	Set this property to **True** to ensure that the package is installed on the boot drive after imaging. ***Note*** • *This setting is only used when deploying a package with an OS image, like with an OSD. It does not affect day-to-day package delivery.*

Limitations Subtab

The **Limitations** subtab of the **Casper Deployment Data** tab is only displayed for PKG and DMG packages.

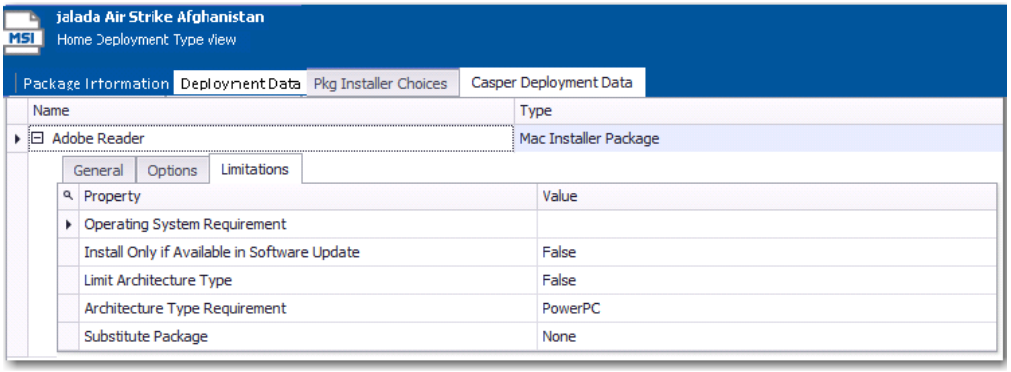


Figure 7-3: Casper Deployment Data / Limitations Tab

The **Limitations** subtab of the **Casper Deployment Data** tab includes the following properties.

Table 7-5 • Casper Deployment Data Tab / Limitations Subtab

Property	Description
Operating System Requirement	Enter operating system version numbers, separated by commas, to specify that the package only be permitted to be deployed to computers with these operating system versions. To restrict installation to OS X 10.6.8, 10.7.x, or 10.8, you would enter the following: `10.6.8, 10.7.x, 10.8`
Install Only if Available in Software Update	Set to **True** to require that this package only be installed if it is available in a software update.
Limit Architecture Type	Set to **True** to require that this package only be installed on machines matching the selected **Architecture Type Requirement**.
Architecture Type Requirement	If **Limit Architecture Type** is set to **True**, select the one of the following to specify the architecture type required to deploy the package: • PowerPC • Intel/X86
Substitute Package	If you want to specify a different package to deploy to computers that do not meet the architecture type requirement, click the Browse button in this field to open the Select Substitute Package Dialog Box, and select a substitute package from either Casper or the Application Catalog.

XenApp Deployment Data Tab

Note • *Because Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles, the* ***XenApp Deployment Data*** *subtab is only displayed for App-V 4.x packages and Citrix XenApp profiles.*

When a XenApp profile or App-V 4.x package is imported into the Application Catalog, Application Manager mines package elements for Citrix XenApp-specific deployment data. You can view and modify data for Citrix XenApp profiles and App-V 4.x packages and add new data by editing the properties on the subtabs of the **XenApp Deployment Data** tab. AdminStudio displays XenApp deployment data in a multi-tabbed, organized format that is easy to navigate through and to update.

- XenApp Deployment Data Tab / XenApp Information Subtab
- XenApp Deployment Data Tab / Advanced Settings Subtab

XenApp Deployment Data Tab / XenApp Information Subtab

Note • *Because Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles, the* ***XenApp Deployment Data*** *subtab is only displayed for App-V 4.x packages and Citrix XenApp profiles.*

When a XenApp profile or App-V 4.x package is imported into the Application Catalog, Application Manager mines package elements for Citrix XenApp-specific deployment data. You can view and modify data for Citrix XenApp profiles and App-V 4.x packages and add new data by editing the properties on the **XenApp Information** subtab of the **XenApp Deployment Data** tab.

The **XenApp Information** subtab of the **XenApp Deployment Data** tab includes the following properties:

Table 7-6 • XenApp Deployment Data Tab / XenApp Information Subtab

Property	Description
Application type	Select one of the following options to specify how this application can be accessed by the user: • **Streamed to server**—Stream the application to the server for access by the user. (Default) • **Streamed to client**—Stream the application to the client device only. • **Streamed to client or Streamed to server**—Stream the application to the client device whenever possible. If the application cannot be streamed to the client device, stream the application to the server for access by the user.
Enabled	To allow users to open this published application, set this property to **True**. (Default) If you set this property to **False**, users will be unable to open this published application even if the application is displayed in the users' application sets (see Hide when disabled property). When users attempt to access the disabled application, they will receive the following message: `ERROR: The application you have requested is not enabled. For more information, contact your Citrix administrator.`
Hide when disabled	Set this property to **True** to prevent this application from appearing in the users' application sets if it is disabled. (Default) If you set this property to **False**, this application will be listed in the users' application sets even if it is disabled. If the user then attempts to access the disabled application, they will receive an error message.

Table 7-6 • XenApp Deployment Data Tab / XenApp Information Subtab

Property	Description
Client application folder	(Optional) Applications can be organized into folders when they are presented to the end user in the users' application set. Whether or not the application's shortcut will appear in a folder depends upon whether you enter a folder name in this property. For example: • **If you enter the folder name** of `Mobile Apps`, the shortcut to open this application will be found in the `Mobile Apps` folder of the users' application set. • **If you do not enter a folder name**, the shortcut to open this application will be found in the root directory of the users' application set.
Add to the client's Start menu	Set this property to **True** to create a shortcut to this application in the user's local Start menu. (Default)
Start menu folder	If you have set the **Add to the client's Start menu** property to **True**, use this property to enter the name of the folder that you want the shortcut to appear in, if any. If you do not enter a folder structure, the application's shortcut will appear in the root directory of Start menu. To specify more than one level of folders, separate each folder name with a backslash, such as: `Marketing\Design\Print`
Add shortcut to the client's desktop	Select one of the following values: • **False**—Do not add a shortcut to the user's local desktop. (Default) • **True**—Add a shortcut to the user's local desktop.
Enable offline access	Select one of the following values: • **False**—Do not permit users to have offline access to this package. (Default) • **True**—Permit users to have offline access to this package.
Cache preference	Select when to cache the streamed application: • **Cache application at launch time**—Caches the application when users launch it. Use this option if the number of users logging on at the same time (and pre-caching their applications) could overload the network. (Default) • **Pre-cache application at login**—Caches the application when the user logs on (selected by default). However, concurrent logons may slow network traffic.
Citrix streaming application profile address	Enter the Citrix streaming application profile address, including the location of the manifest file (`.profile`). For example, enter the UNC path, such as: **`\\MyCitrixServer\Shared\App-V_IntegrationKit\AppStreamingToAppVConduit\AppStreamingToAppVConduit.profile`** ***Important*** • *This field is required for App-V packages.*

Table 7-6 • XenApp Deployment Data Tab / XenApp Information Subtab

Property	Description
Extra command line parameters	Optionally, enter extra command-line parameters. These parameters are used when the profiled application includes asterisks (**) as a placeholder for additional parameters. If no asterisks are in the command-line string, the extra parameters are added at the end of the command-line.
Server names	Enter the server names where this application will be available. Click the browse button to open the **Servers** dialog box, where you can enter multiple server names or import a list of servers from an application server list file (`*.asl`). ***Important*** • *This is a required field.*
Allow anonymous users	Select one of the following values: • **False**—Do not grant access to anonymous users. (Default) • **True**—Grant access to anonymous users.
Accounts	Enter the accounts that you want to have access to this XenApp profile. Click the browse button to open the **Users** dialog box, where you can enter multiple user accounts or import a list of users from an application user list file (`*.aul`). ***Note*** • *If **Allow anonymous users** is set to **True**, this field is not required. If **Allow anonymous users** is set to **False**, this is a mandatory field.*

XenApp Deployment Data Tab / Advanced Settings Subtab

Note • *Because Citrix XenApp server only supports App-V 4.x packages and Citrix XenApp profiles, the **XenApp Deployment Data** subtab is only displayed for App-V 4.x packages and Citrix XenApp profiles.*

When a XenApp profile or App-V 4.x package is imported into the Application Catalog, Application Manager mines package elements for Citrix XenApp-specific deployment data. You can view and modify data for Citrix XenApp profiles and App-V 4.x packages and configure advanced settings by editing the properties on the **Advanced Settings** subtab of the **XenApp Deployment Data** tab.

The **XenApp Information** subtab of the **XenApp Deployment Data** tab includes the following properties:

Table 7-7 • XenApp Deployment Data Tab / Advanced Settings Subtab

Property	Description
Allow connections made through Access Gateway Advanced Edition (version 4.0 or later)	Set to **True** to allow connections that are made through the Citrix Access Gateway Advanced Edition (version 4.0 or later). ***Note*** • *Access Gateway is a universal SSL VPN appliance that can be used to secure client connections to XenApp environments as well as provide secure access to other internal network resources.*
Any connection that meets any of the following filters	Set this field to **True** to only allow connections that meet one or more of the Access Gateway connection filters specified in this list.
Access gateway filters	Click the browse button to open the **Access Gateway Filter** dialog box, where you can enter Access Gateway filters.
Allow all other connections	Set this field to **True** to allow all other connections other than those made through Access Gateway.
Alternate profile locations	Click browse to open the **Alternate Profile Location** dialog box, where you can specify an alternate profile for connections that come from specific IP addresses. For example, an administrator could use an alternate profile to direct users on either side of a WAN to stream applications only from the file or web server on their side of the WAN. When an alternate profile is created, a duplicate of the primary profile is created and stored on a different file share, making it more accessible to the client device. On the **Alternate Profile Location** dialog box, enter the starting and ending IP range for which the alternate profile applies, and then select the alternate profile. After you configure the range, user devices from IP addresses within the specified range access the applications from the alternate profile instead of from the default profile. ***Note*** • *For streamed applications only.*
Maximum instances	Select an integer from the list to specify the maximum number of concurrent connections a user can establish.
Allow only one instance of application for each user	Set this field to **True** to prevent any user from running more than one instance of this application at the same time.

Table 7-7 • XenApp Deployment Data Tab / Advanced Settings Subtab

Property	Description
Application importance	If Preferential Load Balancing is available, select one of the following options to assign importance levels to specific user sessions and applications: • **Low**, which has a value of 1 • **Normal**, which has a value of 2 (default) • **High**, which has a value of 3 Preferential Load Balancing gives administrators the ability to prioritize the allocation of CPU shares to specific users and applications and to direct important user sessions to the XenApp server running the fewest number of important sessions. ***Note*** • *The higher the importance level of the session, the higher the percentage of CPU cycles that will be allotted to it.*
Legacy audio minimum requirement	Set to **Basic** to specify that the client system must have a sound card installed or the published application will fail to launch on the client device.
Enable legacy audio	Set to **True** to allow audio support for applications to which HDX MediaStream Multimedia Acceleration does not apply.
Enable SSL and TLS protocols	Set to **True** to request the use of the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols for plug-ins connecting to the published resource.
Encryption	Select one of the following options to specify which plug-ins are allowed to connect based on their encryption level: • **Basic** • **128-Bit (RC-5)** • **40-Bit (RC-5)** • **56-Bit (RC-5)** • **128-Bit Login Only (RC-5)** ***Note*** • *The **Basic** encryption level should not be used in a secure environment.*
Encryption required	Set this field to **True** to require encryption.
Start this application without waiting for printers to be created	Set to **True** to specify that this application will not open until the client printers are created. Set to **False** to specify that this application will open immediately.

Table 7-7 • XenApp Deployment Data Tab / Advanced Settings Subtab

Property	Description
Session window size	Select one of the following options to configure the resolution of the session size when the application session is started on the XenApp server. • **640x480** • **800x600** • **1024x768** • **1280x1024** • **1600x1200** • **Custom** • **Percent of client desktop**
Width	If you have set **Session window size** to **Custom**, select an integer from this list to set the width of the session window size. The default setting is 1024.
Height	If you have set **Session window size** to **Custom**, select an integer from this list to set the height of the session window size. The default setting is 768.
Percent	If you have set **Session window size** to **Percent of client desktop**, select an integer from this list to set the percentage. The default setting is 75.
Maximum color quality	Select one of the following options to specify the maximum color quality: • **Better Speed (16-bit)** • **Better Appearance (32-bit)** • **256-color (8-bit)**
Hide application title bar	Set to **True** to hide the application title bar. ***Tip*** *• If the application does not have an Exit button built into the user interface, you may want to leave this option unchecked so that the user can use the red "X" to close the application.*
Maximize application at startup	Set to **True** to maximize application at startup. ***Tip*** *• This setting is very useful for mobile devices with smaller screens, where you want the application to always start up at full resolution. On larger screen devices, where you manage multiple windows on the desktop simultaneously, you probably do not want to use this setting.*

Table 7-7 • XenApp Deployment Data Tab / Advanced Settings Subtab

Property	Description
Run application as a least-privileged user account	For applications configured to stream to client devices, you can use this setting to reduce the user privileges for the application, thus reducing security risks. To reduce the user privileges for the application, set this property to **True**. Setting this property to **True** configures all users, even those with an administrator account, to run the application with normal user privileges. The default setting is **False**. ***Important*** *• Before you set this property to* ***True****, test the application with a limited access configuration. Some applications expect users to have elevated privileges and might fail to operate correctly when launched by users with a least-privileged user account.*

Software Identification Tag Tab

Important *• The* ***Software Identification Tag*** *tab is only displayed for Windows Installer packages.*

You can view and edit the software ID tag information for an individual Windows Installer package on the **Software Identification Tag** tab of the **Home Deployment Type View** in Application Manager.

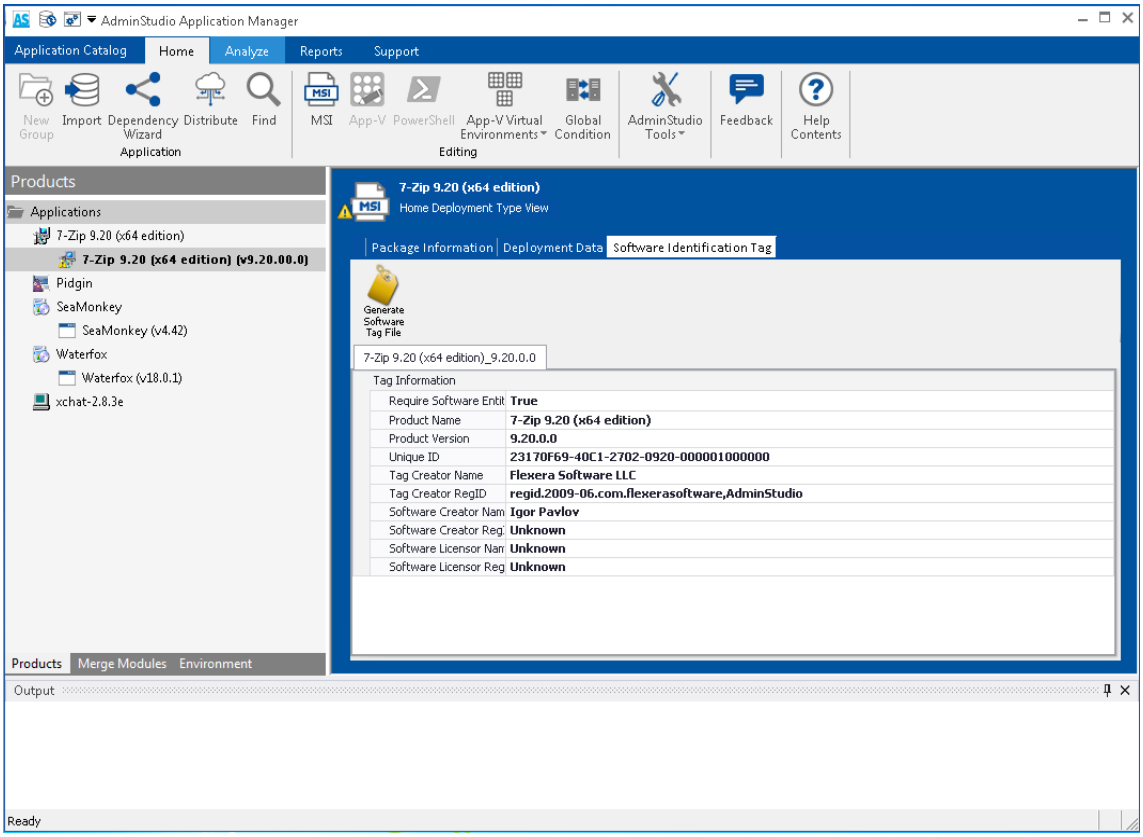


Figure 7-4: Software Identification Tag Subtab of Home Deployment Type View

The **Software Identification Tag** tab includes the following properties:

Table 7-8 • Home Deployment Type View / Software Identification Tag Subtab Properties

Property	Description
Require Software Entitlement	To specify that you want to require your product to have a corresponding software entitlement in order for software reconciliation to be considered successful, set this property to **True**. In general, if the software must be purchased, this property should be set to **True**; if the software is free, this property should be set to **False**.
Product Name	Name of the product, read from the Product Name property of the Windows Installer package.
Product Version	Version of the product, read from the Product Version property of the Windows Installer package.
Unique ID	The product GUID, which is the ProductCode of the MSI package or the unique string used for the Add and Remove Programs uninstall key name, is used to uniquely identify the product in the software identification tag file.

Table 7-8 • Home Deployment Type View / Software Identification Tag Subtab Properties

Property	Description
Tag Creator Name	Enter a name to identify the creator of this tag file. The default value is: `Flexera`

Tag Creator RegID	Enter a RegID to identify the creator of this tag file, using the following format:
	`regid.YYYY-MM.ReversedDomainName,optional_division`
	For example:
	`regid.2009-06.com.yourcompany,GlobalProductDivision`
	
	Note *• For more information, see About Software Tagging RegIDs and About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields.*

Software Creator Name	(Optional) Enter a name to identify the creator of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field. ***Note*** *• For more information, see About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields.*
Software Creator RegID	(Optional) Enter a RegID to identify the creator of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field. ***Note*** *• For more information, see About Software Tagging RegIDs and About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields.*
Software Licensor Name	(Optional) Enter a name to identify the licensor of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field. ***Note*** *• For more information, see About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields.*
Software Licensor RegID	(Optional) Enter a RegID to identify the licensor of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field. ***Note*** *• For more information, see About Software Tagging RegIDs and About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields.*

Table 7-8 • Home Deployment Type View / Software Identification Tag Subtab Properties

Property	Description
Generate Software Tag File	Click to save your edits and generate an updated tag file. When this button is clicked, Application Manager will generate a new transform file for this package that includes the newly modified tag file, and will then reimport the package into the Application Catalog along with its updated transform file.

Altiris Deployment Data Tab

Important • *The **Altiris Deployment Data** tab is only displayed for Windows Installer, VMware ThinApp, and legacy installer packages.*

When a Windows Installer, VMware ThinApp, or legacy installer package is imported into the Application Catalog, Application Manager mines package elements for Altiris-specific deployment data. You can view and modify data for these packages and configure command line settings by editing the properties on the **Package Information** and **Command Lines** subtabs of the **Altiris Deployment Data** tab.

Package Information Subtab

On the **Package Information** subtab of the **Altiris Deployment Data** tab, you can view and modify Altiris-specific data for packages.

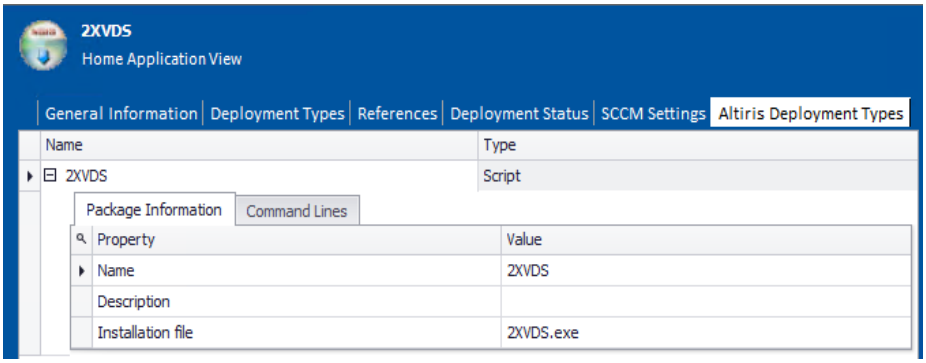


Figure 7-5: Altiris Deployment Data Tab / Package Information Subtab

The **Package Information** subtab of the **Altiris Deployment Data** tab includes the following properties:

Table 7-9 • Altiris Deployment Data Tab / Package Information Subtab

Property	Description
Name	Name that will identify this package on the Symantec Altiris server.
Description	Description that will be associated with this package on the Symantec Altiris server.

Table 7-9 • Altiris Deployment Data Tab / Package Information Subtab

Property	Description
Installation file	Lists the name of the imported package file, the file that will launch the application deployment. ***Note*** *• The value in this field will populate the* ***Installation file*** *field in Altiris. An application being installed by Altiris can consist of multiple files and subfolders of files. Therefore, if additional files are added to this application on the Altiris server, the* ***Installation file*** *field will identify the one file that launches application deployment.*

Command Lines Subtab

On the **Command Lines** subtab of the **Altiris Deployment Data** tab, you can configure a package's Altiris-related command line settings.

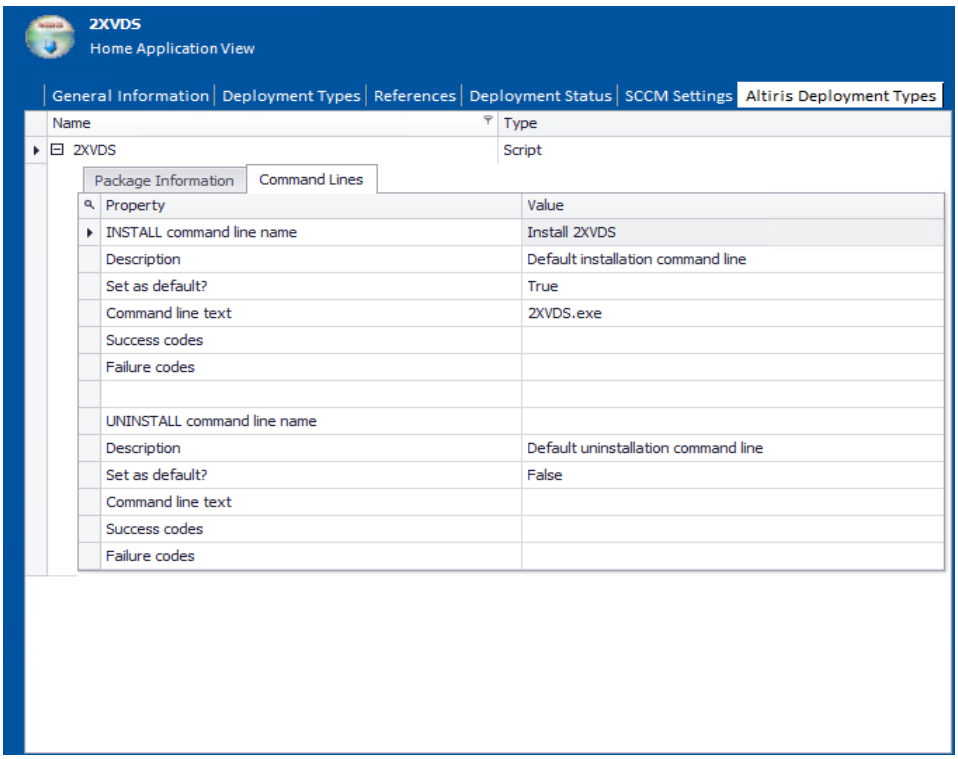


Figure 7-6: Altiris Deployment Data Tab / Command Lines Subtab

The **Command Lines** subtab of the **Altiris Deployment Data** tab includes the following properties:

Table 7-10 • Altiris Deployment Data Tab / Command Lines Subtab

Property	Description
INSTALL / UNINSTALL / REPAIR / CUSTOM Command Line Name	User-specified name of the command line resource, such as: • Uninstall Adobe Photoshop Elements • Repair Adobe Photoshop Elements
Description	Name which describes the purpose of the specified command line, such as: • Default uninstallation command line • Default repair command line
Set as default?	Select **True** to set this command line as the default for the Install, Uninstall, or Repair action when a package is deployed from the Altiris Server to a client. ***Note*** • *For example, you can define multiple installation command lines for a software resource, but only one installation command line can be the default.* ***Note*** • *Not required for CUSTOM command lines.*
Command line text	Actual command line code, such as: `msiexec /x "{E6A418EA-7AF1-42E2-A7B6-9D7B7382856D}" /q`
Success codes	Codes returned when this command line is successful.
Failure codes	Codes returned when this command line fails.

Workspace ONE Deployment Data Tab

Important • *The **Workspace ONE Deployment Data** tab is only displayed for Apple iOS,* MSI, EXE, and Google Android *packages.*

Workspace ONE is a leading global Mobile Device Management (MDM) provider. Using AdminStudio, you can manage and publish Apple iOS (local and public store), MSI, EXE, and Google Android (local and public store) packages to Workspace ONE. You can view and modify data for these packages by editing the properties on the **Workspace ONE Deployment Data** tab.

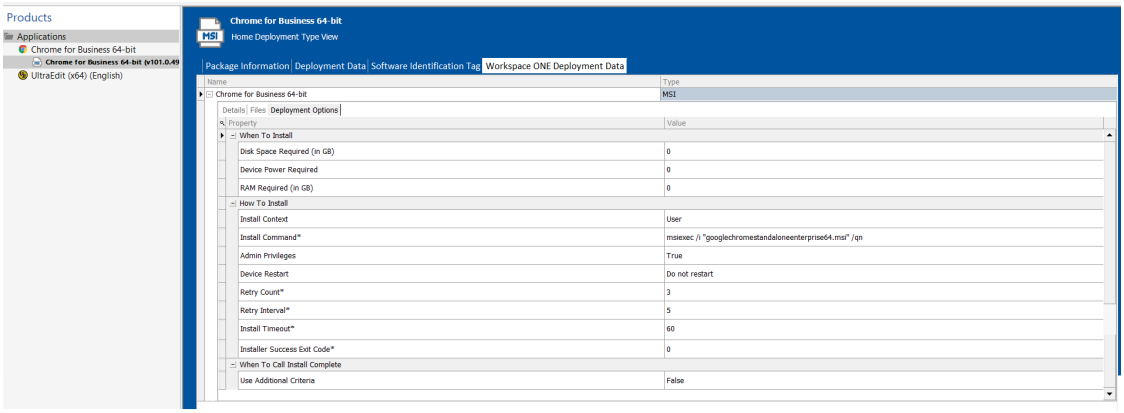


Figure 7-7: Workspace ONE Deployment Data Tab

The **Workspace ONE Deployment Data** tab of the Home Deployment Type View has the following subtabs:

- Workspace ONE Deployment Data tab / Details Subtab
- Workspace ONE Deployment Data tab / Files Subtab
- Workspace ONE Deployment Data tab / Deployment Options Subtab

Workspace ONE Deployment Data tab / Details Subtab

The Details subtab of the Workspace ONE Deployment Data tab lists general information about the package and application ID, application version, and supported modules.

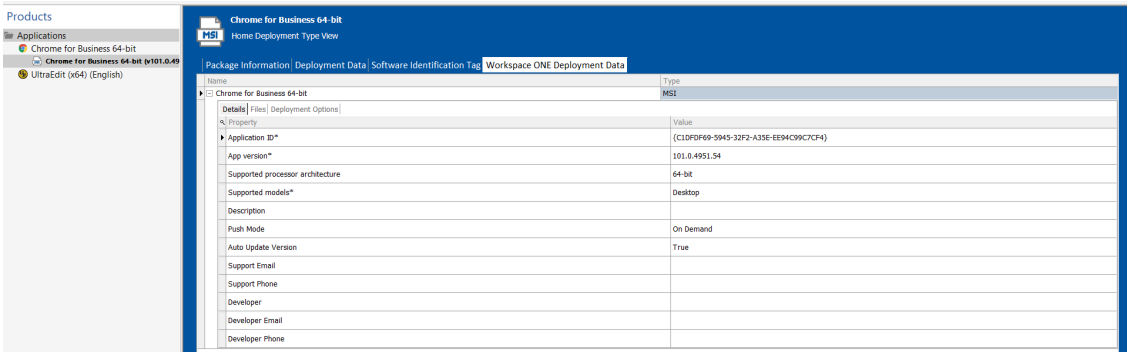


Figure 7-8: Workspace ONE Deployment Data Tab / Details Subtab

The Details subtab of the Workspace ONE Deployment Data tab includes the following properties:

Table 7-11 • Workspace ONE Deployment Data Tab / Details Subtab

Property	Description
Application ID	The unique identifier of the application within the package. The Application ID will be auto populated. In case of EXE packages, initially this field will be empty. The Application ID will be updated upon successful publish to Workspace ONE. This is a mandatory property and non-editable. ***Note*** • *This is applicable only for MSI and EXE packages.*
App Version	The release number (or release identifier) of an application. This is a mandatory property and non-editable. ***Note*** • *This is applicable only for MSI and EXE packages.*
Supported processor architecture	Processor architecture of the package. ***Note*** • *This is applicable only for MSI and EXE packages.*
Supported models	Supported models of the package. By default, it will be set to Desktop. This is a mandatory property. ***Note*** • *This is applicable only for MSI and EXE packages*
Description	Description providing details of the package. ***Note*** • *This is applicable only for MSI and EXE packages.*
Push Mode	Select one of the following options: • **Auto**—Install the application immediately. • **On Demand**—Have the end user install the application manually. (Default)
Auto Update Version	Select one of the following options: • **True**—Automatically update the application to the latest version. • **False**—Do not automatically update the application.
Support Email	Email address identifying the end user's Support point-of-contact for this application.
Support Phone	Phone number of the end user's Support site for this application.

Table 7-11 • Workspace ONE Deployment Data Tab / Details Subtab

Property	Description
Developer	Name of the developer who created this application.
Developer Email	Email address of this application's developer.
Developer Phone	Phone number of this application's developer.

Workspace ONE Deployment Data tab / Files Subtab

The Files subtab of the Workspace ONE Deployment Data tab lists custom script type, uninstall command, and specifies the custom script for MSI and EXE uninstall process.

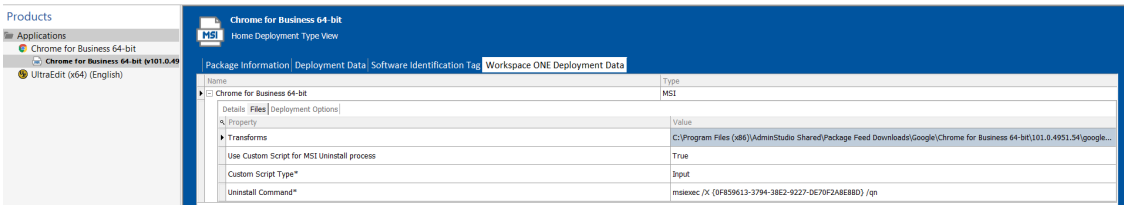


Figure 7-9: Workspace ONE Deployment Data Tab / Files Subtab

The Files subtab of the Workspace ONE Deployment Data tab includes the following properties:

Table 7-12 • Workspace ONE Deployment Data Tab / Files Subtab

Property	Description
Transforms	Lists the number of transforms associated with a Windows Installer package, and the transform file locations. ***Note*** • *Applicable only for MSI.*
Use Custom Script for MSI Uninstall process	Select one of the following options: • **True** • **False** ***Note*** • *Applicable only for MSI.*

Table 7-12 • Workspace ONE Deployment Data Tab / Files Subtab

Property	Description
Custom Script Type	Select one of the following options: • **Input** • **Upload** ***Note*** *• This is a mandatory property.*
Uninstall Script	This field populates only when you select **Upload** option in the **Custom Script Type**. `Click browse and select js, jse, ps1, ps1xml, psc1, psd1, psm1, pssc, cdxml, vbs, vbe, wsf, wsc files.`
Uninstall Command	Specify the command that use to uninstall this package. ***Note*** *• This is a mandatory property.*

Workspace ONE Deployment Data tab / Deployment Options Subtab

The Deployment Options subtab of the Workspace ONE Deployment Data tab lists install command, When to Install, How to install, and When to Call Install Complete options to deploy MSI and EXE packages successfully.

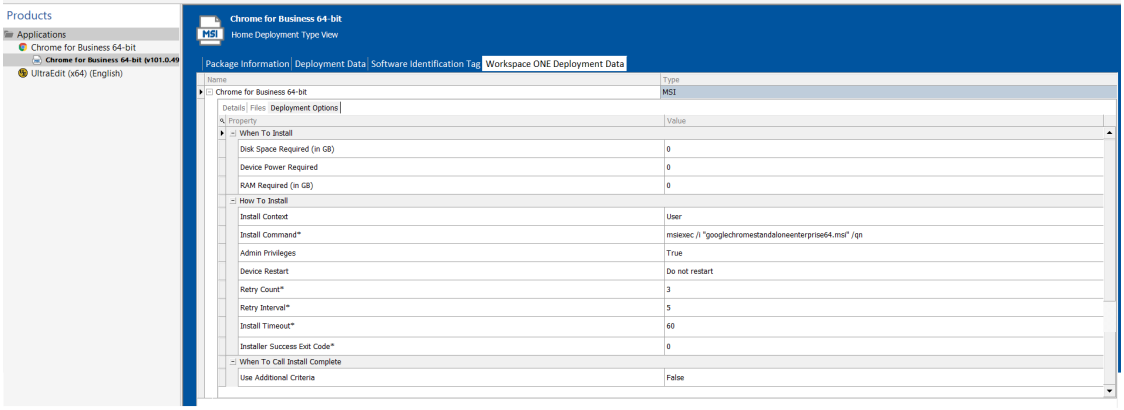


Figure 7-10: Workspace ONE Deployment Data Tab / Deployment Options Subtab

The Deployment Options subtab of the Workspace ONE Deployment Data tab includes the following properties:

Table 7-13 • Workspace ONE Deployment Data Tab / Deployment Options Subtab

Property	Description
Disk Space Required (in GB)	Specify free disk space required to install the package, in GB. ***Note*** • *Once published to Workspace ONE, it gets converted into KB format.*
Device Power Required	Specify the required device power.
RAM Required (in GB)	Specify the RAM requirement, in GB. ***Note*** • *Once published to Workspace ONE, it gets converted into MB format.*
Install Context	Select one of the following options: • **User** • **Device** By default, it will be selected to **User**.
Install Command	Specify the command that use to install this package. ***Note*** • *This is a mandatory property.*
Admin Privileges	Select one of the following options: • **True** • **False** By default, it will be selected to **True**.
Device Restart	Select one of the following options: • **Do not restart** • **Force restart** • **User-engaged restart** By default, it will be set to **Do not restart**.
Number of days after which device automatically reboots	This field populates only when you select **User-engaged restart** option in the **Device Restart**. Specify the value between 0 to 7.

Table 7-13 • Workspace ONE Deployment Data Tab / Deployment Options Subtab

Property	Description
Retry Count	Specify the value between 0 to 10. By default, it will be set to 3. ***Note*** *• This is a mandatory property.*
Retry Interval	Specify the value between 0 to 10. By default, it will be set to 5. ***Note*** *• This is a mandatory property.*
Install Timeout	Specify the value between 0 to 150. By default, it will be set to 60. ***Note*** *• This is a mandatory property.*
Installer Reboot Exit Code	Specify the reboot exit code. By default, it will be set to 1641.
Installer Success Exit Code	Specify the exit code. By default, it will be set to 0. ***Note*** *• This is a mandatory property.*
Use Additional Criteria	Select one of the following options. By default, it will be selected to **False**. • **True** • **False** ***Note*** *• Applicable only for MSI.*
Identify Application By	Select one of the following options: • **Defining criteria** • **Using custom script** ***Note*** *• The **Identify Application By** field populates only when you set **Use Additional Criteria** to **True**.* ***Note*** *• A warning message "Adding criteria to when to call install complete is currently not supported" will be displayed on selection of **Defining criteria**.*

Table 7-13 • Workspace ONE Deployment Data Tab / Deployment Options Subtab

Property	Description
Script Type	The **Script Type** field populates only when you select **Use custom script** in the **Identify Application By** field. Select one of the following type: • **Jscript** • **PowerShell** • **VBScript** ***Note*** *• This is a mandatory property.*
Command to Run the Script	Specify the command that use to run the script.
Custom Script File	Click browse and select the script file type. ***Note*** *• This is a mandatory property.*
Success Exit Code	Specify the exit code. ***Note*** *• This is a mandatory property.*
Using custom script	Specify the custom script.

Microsoft Intune Deployment Data Tab

Important *• The **Intune Deployment Data** tab is only displayed for MSI, MSIX, and Intunewin packages.*

AdminStudio supports publishing of MSI, MSIX and Intunewin package types to Microsoft Intune. The Intune Deployment Data tab appears only for the supported package types. The Intune Deployment Data Tab displays various deployment related properties organized in subtabs for easy navigation. You can view and edit properties values either inline or by using the easy-to-using wizards in Detection rules, Requirements and Return code subtabs.

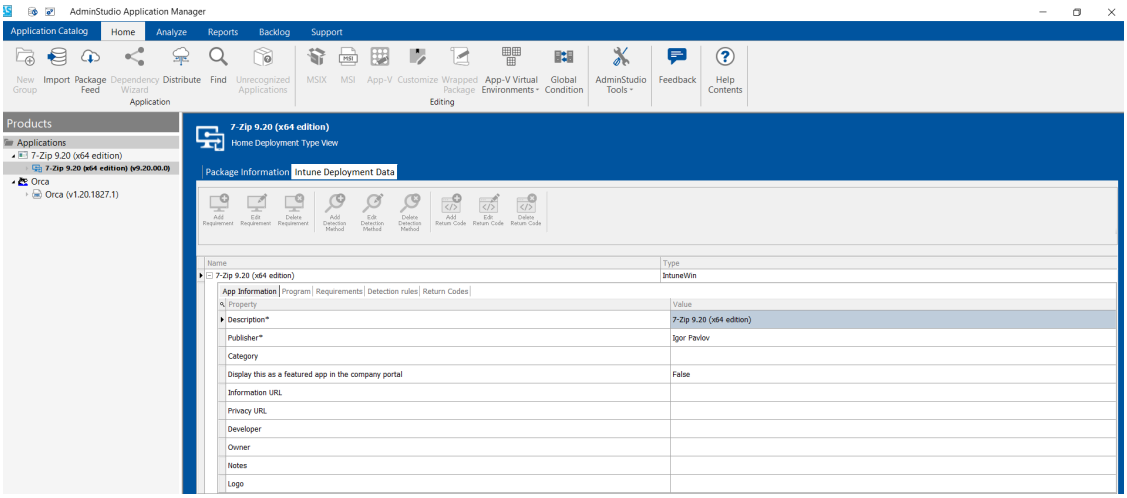


Figure 7-11: Home Deployment Type View / Intune Deployment Data Tab

The **Intune Deployment Data** tab of the **Home Deployment Type View** has the following subtabs:

- Intune Deployment Data Tab / App Information Subtab
- Intune Deployment Data Tab / Program Subtab
- Intune Deployment Data Tab / Requirements Subtab
- Intune Deployment Data Tab / Detection rules Subtab
- Intune Deployment Data Tab / Return Codes Subtab

Intune Deployment Data Tab / App Information Subtab

The **App Information** subtab of the **Intune Deployment Data** tab of the **Home Deployment Type View** lists general information about package contents, including application description and publisher.

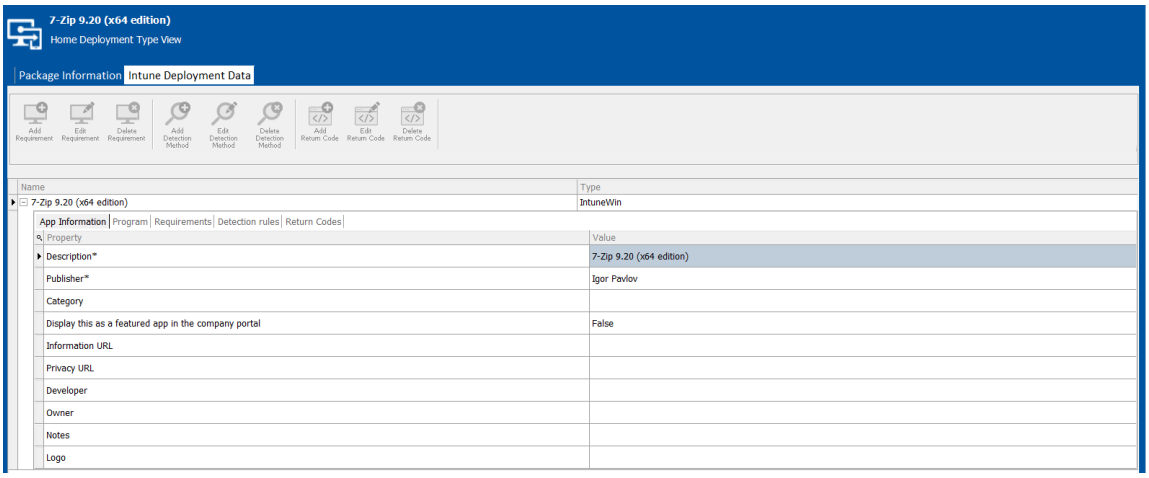


Figure 7-12: Intune Deployment Data Tab / App Information Subtab

The App Information subtab of the Intune Deployment Data tab includes the following properties:

Table 7-14 • Intune Deployment Data Tab / App Information Subtab

Property	Description
Description	Description of the package. ***Note*** • *This is a mandatory property.*
Publisher	Publisher of the package. ***Note*** • *This is a mandatory property.*
Category	Select one or more of the built-in app categories, or select a category that you created.
Display this as a feature app in the company portal	Select **True** to display the app prominently on the main page of the company portal when users browse for apps.
Information URL	Optionally, enter the URL of a website that contains information about this app. The URL appears in the company portal.
Privacy URL	Optionally, enter the URL of a website that contains privacy information for this app. The URL appears in the company portal.
Developer	Optionally, enter the name of the app developer.
Owner	Optionally, enter a name for the owner of this app. An example is HR department.
Notes	Enter any notes that you want to associate with this app.
Logo	Upload an icon that is associated with the app. This icon is displayed with the app when users browse through the company portal.

Intune Deployment Data Tab / Program Subtab

The **Program** subtab of the **Intune Deployment Data** tab of the **Home Deployment Type View** lists the install and uninstall commands and specifies the install and restart behavior.

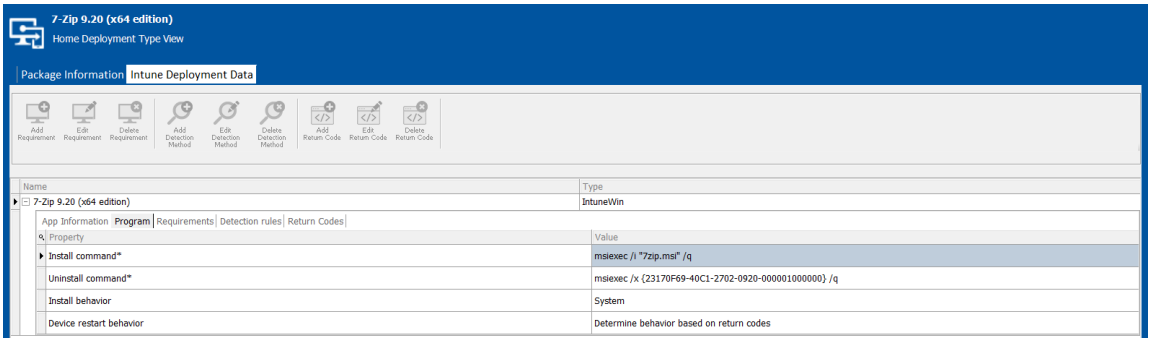


Figure 7-13: Intune Deployment Data Tab / Program Subtab

The Program subtab of the Intune Deployment Data tab includes the following properties:

Table 7-15 • Intune Deployment Data Tab / Program Subtab

Property	Description
Install command	Specify the command line that use to install this package on a client machine, including any required installation parameters. ***Note*** • *This is a mandatory property.*
Uninstall command	Specify the command line that use to uninstall this package from a client machine, including any required parameters. ***Note*** • *This is a mandatory property.*
Install behavior	Select one of the following options: • **User**—The application installs for only the user who it is deployed to. • **System**—The application installs only once and is available to all users. For dual-purpose MSI apps, changes will prevent updates and uninstalls from successfully completing until the value applied to the device at the time of the original install is restored.

Table 7-15 • Intune Deployment Data Tab / Program Subtab

Property	Description
Device restart behavior	Configure the device restart behavior in the client machine after the application is installed. Select one of the following option: • **Determine behavior based on return codes**—Restarts the device based on the return codes configuration settings. • **No specific action**—Suppress device restarts during the application install for MSI-applications. • **App Install may force a device restart**—Allows the application install to complete without suppressing restarts. • **Intune will force a mandatory device restart**—Restart the device after successful application installation.

Intune Deployment Data Tab / Requirements Subtab

You can use the Requirements subtab of the Intune Deployment Data tab to add device or additional requirements that enable you to successfully deploy this package.

To add a requirement to the Requirements subtab, click the **Add Requirement** button the ribbon toolbar to open the Requirement Wizard. You can set device requirements and additional requirements. For more information, see Requirement Wizard of Intune Deployment Data Tab.

To modify an existing requirement, select the requirement and click **Edit Requirement** in the ribbon toolbar.

To delete an existing requirement, select the requirement and click **Delete Requirement** in the ribbon toolbar.

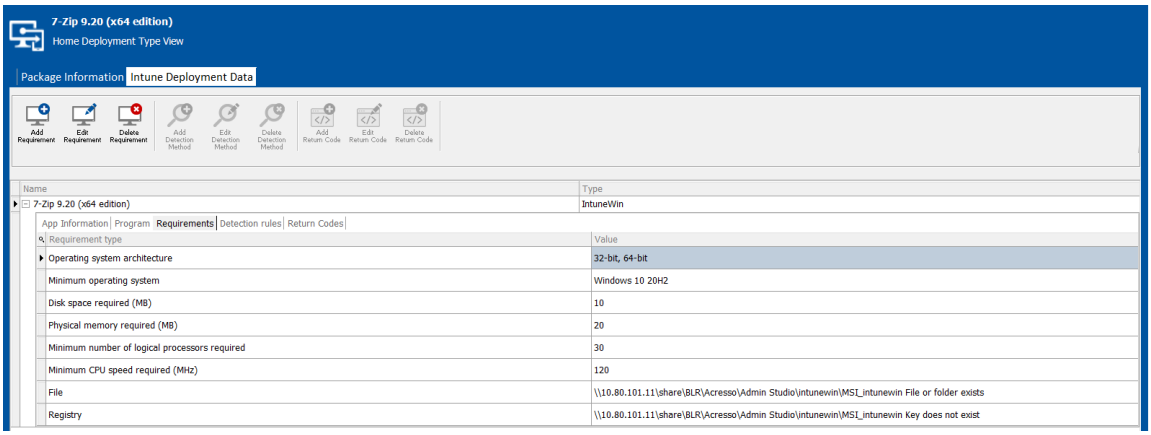


Figure 7-14: Intune Deployment Data Tab / Requirements Subtab

The Requirements subtab of the Intune Deployment Data tab includes the following properties:

Table 7-16 • Intune Deployment Data Tab / Requirements Subtab

Property	Description
Operating system architecture	Choose the Operating system architecture required to install the application.
Minimum operating system	Select minimum operating system to install the application.
Disk space required (MB)	Specify free disk space required on the system drive to install the application.
Physical memory required (MB)	Specify Physical memory (RAM) required to install the application.
Minimum number of logical processors required	Specify the minimum number of logical processors required to install the application.
Minimum CPU speed required (MHz)	Specify the minimum CPU speed required to install the application.

Intune Deployment Data Tab / Detection rules Subtab

The **Detection rules** subtab of the Intune Deployment Data tab of the Home Deployment Type View lists methods to detect whether this package is already installed on the system.

To add a detection rule to the Detection rule subtab, click the **Add Detection Method** button in the ribbon toolbar to open the **Detection rule Wizard**. For more information, see Detection Rule Wizard of Intune Deployment Data Tab.

Note • *Make sure at least one detection rule has been added for a successful publish of the Intunewin package.*

To modify an existing detection rule, select the detection rule and click **Edit Detection Method** in the ribbon toolbar.

To delete an existing detection rule, select the detection rule and click **Delete Detection Method** in the ribbon toolbar.

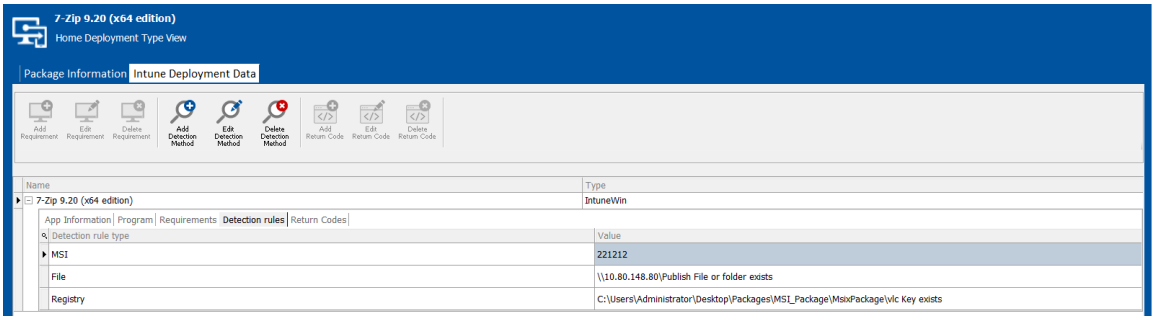


Figure 7-15: Intune Deployment Data Tab / Detection rules Subtab

Intune Deployment Data Tab / Return Codes Subtab

You can view and edit a package's return codes on the **Return Codes** subtab of the **Intune Deployment Types** tab. Return codes are used to indicate whether a restart is required, whether an installation is complete, and to customize the text shown to users when a specific code is returned.

The following return codes are populated by default during package import:

- 0—Success
- 1707—Success
- 3010—Soft Reboot
- 1641—Hard Reboot
- 1618—Retry

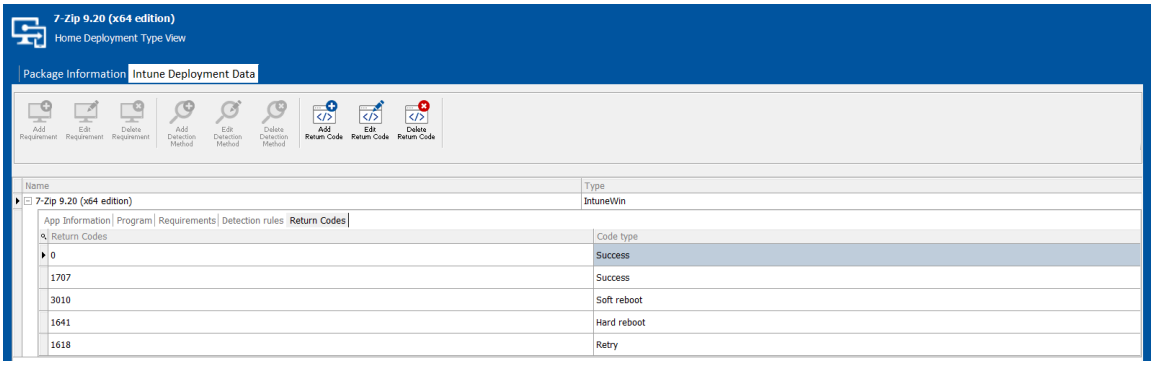


Figure 7-16: Intune Deployment Data Tab / Return Codes Subtab

On the Return Codes tab, you can add, edit, and delete return codes.

- **Adding a return code**—Click **Add Return Code** in the ribbon and define a new return code on the **Return code** wizard. For more information, see Return Code Wizard of Intune Deployment Data Tab.
- **Editing a return code**—Select a return code, click **Edit Return Code** in the ribbon, and edit the details of the return code.
- **Deleting a return code**—Select a return code, click **Delete Return Code** in the ribbon, and confirm the deletion.

Home Deployment Type View Subnode Views

If you click on the plus sign to expand a package in the Application Manager Home Deployment Type View, a node is listed for each available constituent view. For example, for a Windows Installer package when the **Home** tab is selected, the following nodes are listed:

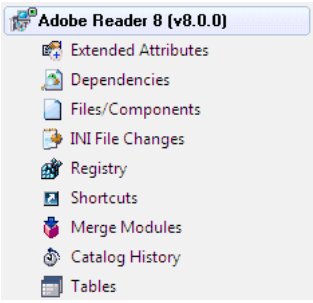


Figure 7-17: Windows Installer Package Nodes / **Home** Tab

When you select one of these nodes, a constituent view opens in the right pane:

- Extended Attributes View (Packages)
- Dependencies View
- Files View
- INI File Changes View
- Registry View
- Shortcuts View
- Merge Modules View
- Catalog History View
- App-V History View
- Tables View
- File Type Associations View
- Environment Variables View

Note • *The **Only Display View Nodes With Data** option on the **General** tab of the Application Manager Options Dialog Box controls whether product nodes (constituent views) appear if no data is contained in that view. If you select the option, products containing views without data will not display those views. For example, if a product has no dependencies, then the **Dependencies** node is not displayed for that product.*

Extended Attributes View (Packages)

The Application Manager Extended Attributes View displays the optional extended attributes associated with the package. To open a package's Extended Attributes View, select the **Extended Attributes** node under that package in the package tree.

These attributes are dynamically populated based on an external Package Extended Attribute Description File (in XML format). You can specify the name and location of this file on the **General** tab of the Application Manager Options dialog box.

On the left side of the view, the name for each attribute is displayed; the right side displays the value for the attribute. These values are in read-only fields, from which you can highlight and copy text, or, in the case of file links, launch the linked file. The file location displayed represents the location from which the file was originally imported; the file is actually stored within the Application Catalog and extracted into a temp directory when you click on the file link. This temp directory is purged when Application Manager closes, as long as the file is not locked by another application or process.

If you click on the attribute name, you can either provide the value in the **Extended Attribute Property** dialog box (for text attributes) or browse for a file in a Browse dialog (for file attributes).

Dependencies View

The **Dependencies View**, which opens in the Application Manager **Home** tab when a package's **Dependencies** node is selected in the tree, varies by deployment type:

- Dependencies View / Windows Installer Package
- Dependencies View / App-V Package

Dependencies View / Windows Installer Package

On the **Dependencies View**, you can view a list of all of the files of a selected Windows Installer package that have dependencies with files used by other packages or operating systems in the Application Catalog. This view is displayed for Windows Installer `.msi` packages in which file dependency information exists.

The **Dependencies View** is populated when you scan a Windows Installer package for dependencies using the **Auto detect dependencies** option of the **Dependency Wizard**.

Note • *If the **Only Display View Nodes With Data** option on the **General** tab of the Application Manager Options dialog box is selected, if no dependencies are found, the **Dependencies** node will not be displayed.*

The following information is displayed on the **Dependencies** view:

Table 7-17 • Dependencies View

Option	Description
File Name	Name of the file contained in the Windows Installer package; all other columns describe dependencies for this file.
Architecture	Machine architecture for the file.
16 bit	Signifies whether the file is meant for 16-bit machines.
Terminal Server Aware	Signifies whether the file is Terminal Server aware or not.
.NET Assembly	Shows **NotCLR** if the file is not a .NET assembly; otherwise it displays the version of .NET it depends upon.
SubSystem	Signifies the sub-system for the file.

Table 7-17 • Dependencies View

Option	Description
Signed	Signifies whether the file is digitally signed.
Signee	If the file is signed, this columns lists the name of the signee.
PE Dependent on	Lists other files that this one depends on.
PE Language	The language for the file.

Note • *For Microsoft App-V packages, a dependency scan is automatically performed during import into the Application Catalog, and the results are displayed on the App-V package's* ***Dependencies View****. For more information, see Viewing App-V Package Dependencies.*

Dependencies View / App-V Package

Note • *This information applies to App-V 4.x packages.*

The **Dependencies View** lists both the applications the App-V package is dependent on and the applications dependent upon this App-V package.

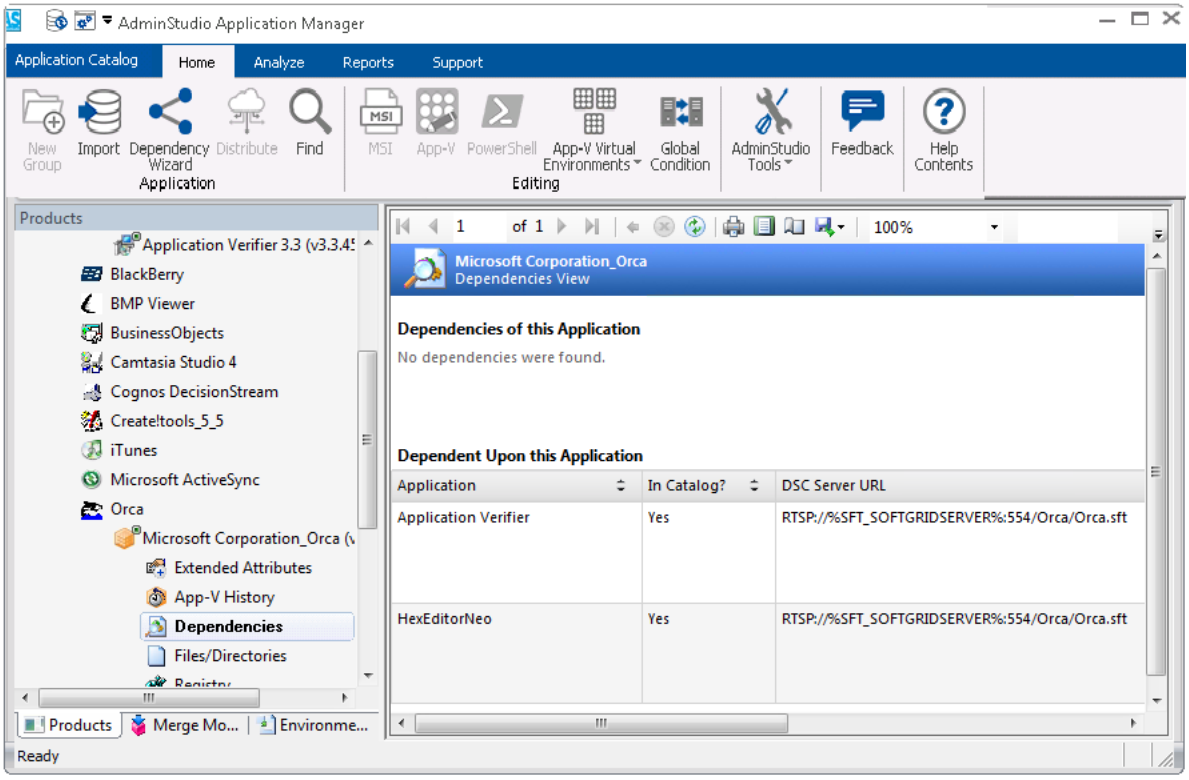


Figure 7-18: Dependencies View / App-V Package

For each dependency, the following information is listed:

- Application
- In Catalog? (Yes / No)
- DSC Server URL
- Server URL
- Mandatory? (Yes / No)

Files View

The **Files View**, which opens in the Application Manager **Home** tab when a package's **Files/Components** or **Files/Directories** node is selected in the tree, varies by deployment type:

- Files/Components View / Windows Installer Package
- Files/Directories View / App-V Package

Files/Components View / Windows Installer Package

The **Files/Components View** displays the files and components in the Windows Installer package. The following information is displayed:

Table 7-18 • File/Components View Information

Column	Description
Component	Name of component that the file listed in the FileName column is associated with.
FileName	Name of file.
FileSize	Size of the file listed in the FileName column.
Version	Version of the file listed in the FileName column.
Path	Installation location of the file listed in the FileName column.

Files/Directories View / App-V Package

The **App-V Files/Directories View** lists the files and directories included in the App-V package.

For each file/directory, the following information is listed:

Table 7-19 • App-V File Information

Column	App-V 4.x	App-V 5
Directory Path	X	X
Short Name	X	

Table 7-19 • App-V File Information

Column	App-V 4.x	App-V 5
Long Name	X	
File Name		X
File Size	X	X
VFS Path	X	
Feature Block 1	X	X
App-V Version	X	
App-V Data Type	X	

INI File Changes View

The **INI File Changes View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **INI File Changes** node, displays any INI file changes made by that package. The following information is displayed:

Table 7-20 • INI File Changes View Information

Column	Description
Component	Name of component that makes an entry in the INI File.
FileName	Name of INI File that the component listed in the Component column makes an entry in.
DirProperty	The directory location where the INI File will be installed.
Section	The section of the INI file where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Registry View

The **Registry View**, which opens in the Application Manager **Home** tab when a package's **Registry** node is selected in the tree, varies by deployment type:

- Registry View / Windows Installer or Microsoft UWP App Packages
- Registry View / App-V Package

Registry View / Windows Installer or Microsoft UWP App Packages

Note • *AdminStudio must be installed on a Windows 8 or higher operating system in order to import registry information for Microsoft UWP app packages (.appx) or Windows 8 app packages (.appx).*

The **Registry View** displays any registry entries created or changed by the Windows Installer or Microsoft UWP app package. The following information is displayed:

Table 7-21 • Registry View Information

Column	Description
Component	The name of the component that is creating a Registry entry.
Root	Default value of Key.
Key	Key of the Registry Entry that this component is making.
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Registry View / App-V Package

The **Registry View** lists any registry entries created or changed by the App-V package.

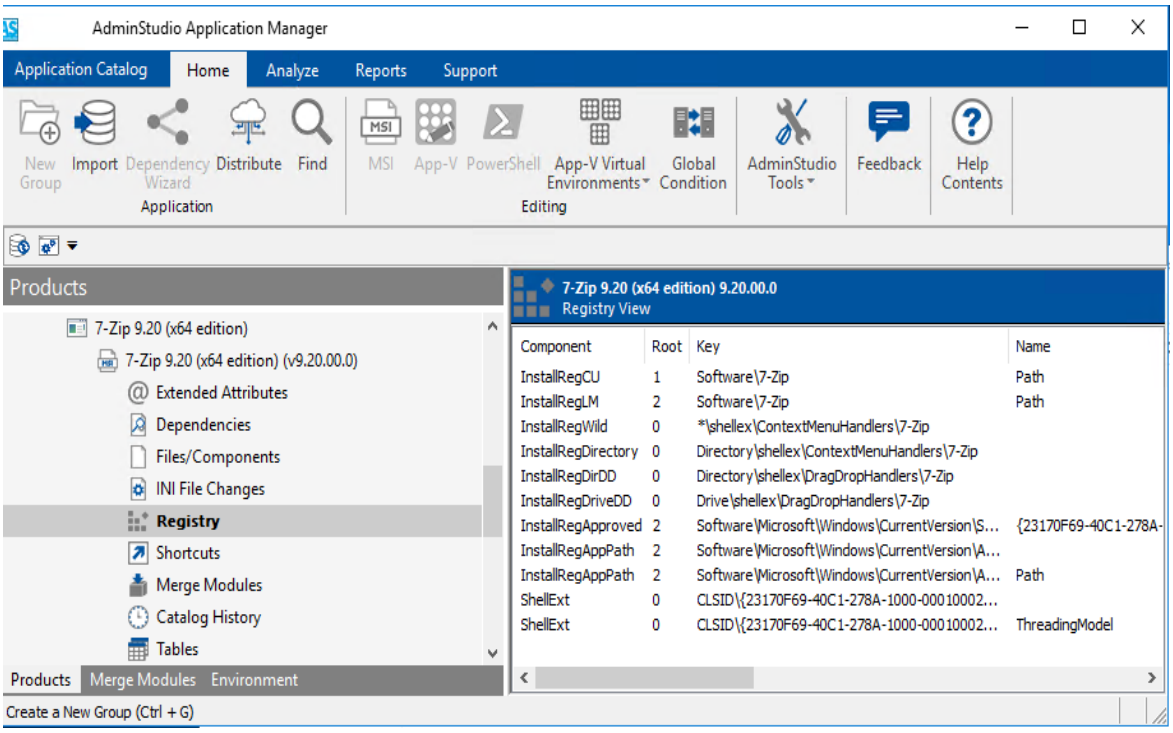


Figure 7-19: App-V Registry View

For each registry entry, the following information is listed:

- Key Path
- App-V Override
- Value Name
- Data
- Type

Shortcuts View

The **Shortcuts View**, which opens in the Application Manager **Home** tab when a package's **Shortcuts** node is selected in the tree, varies by deployment type:

- Shortcuts View / Windows Installer Package
- Shortcuts View / App-V Package

Shortcuts View / Windows Installer Package

The **Shortcuts View** displays any shortcuts created by the Windows Installer package. The following information is displayed:

Table 7-22 • Shortcuts View Information

Column	Description
Component	Name of the component that the shortcut listed in the Name column is associated with.
Name	Name of the shortcut.
Directory_	Directory where the shortcut will exist.
Target	Directory and executable that the shortcut invokes.

Shortcuts View / App-V Package

The **Shortcuts View** displays any shortcuts created by the App-V package.

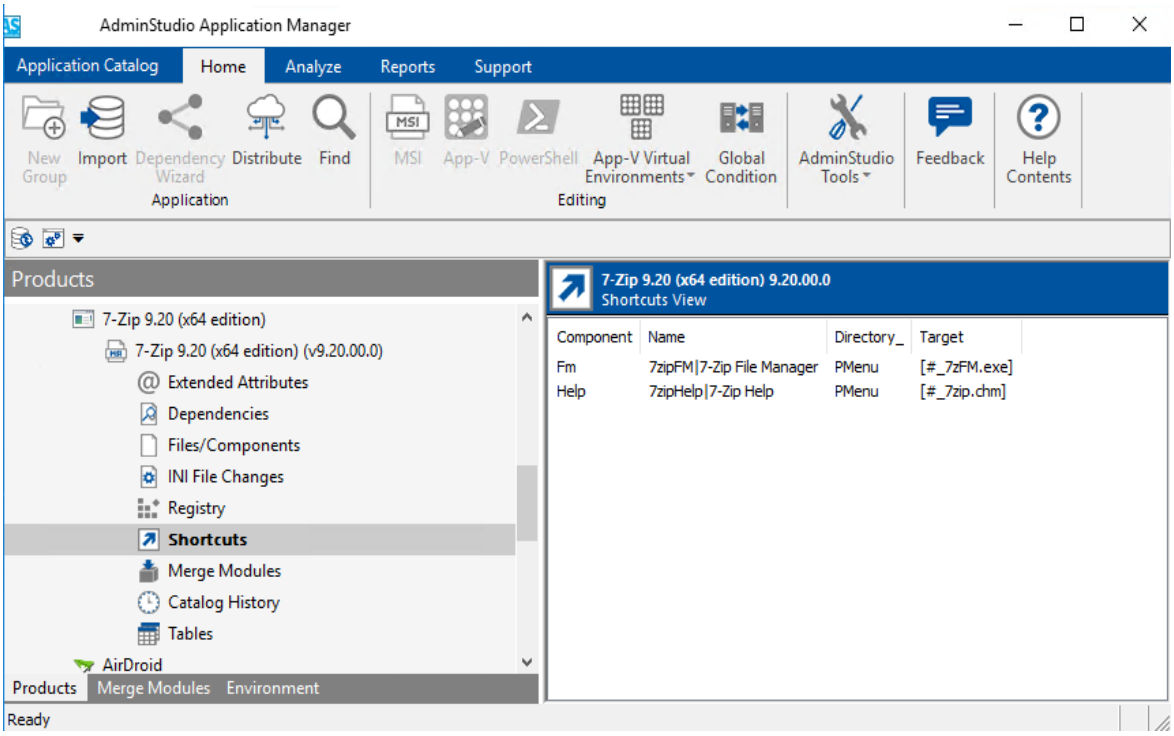


Figure 7-20: App-V Shortcuts View

Under the **Shortcuts** subheading, the following information is listed for each shortcut:

- Shortcut Name
- Target
- Arguments
- Working Directory
- Target Version
- Location

Under the **Shortcut Dependency** subheading, the following information is listed for each shortcut dependency:

- Shortcut Name
- Href
- GUID
- Is Mandatory

Under the **Shortcut Script** subheading, the following information is listed for each shortcut script:

- Shortcut Name
- Body (of script)

Important • *The **Shortcut Dependency** and **Shortcut Script** subheadings only apply to App-V 4.x packages.*

Merge Modules View

The **Merge Modules View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Merge Modules** node, displays any merge modules included the package. The following information is displayed:

Table 7-23 • Merge Modules View Information

Column	Description
Title	The title of the Merge Module included with this package.
ModuleID	The number which uniquely identifies the Merge Module listed in the Title column.
Version	The version of the Merge Module listed in the Title column.
Language	The language that the Merge Module listed in the Title column was written for.

Catalog History View

The **Catalog History View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Catalog History** node, displays a list of logged events for that package. The following information is included:

Table 7-24 • Information Displayed in the Catalog History View

Item	Description
Action	Name of the event which was logged: • Import/Reimport • Validation • Conflict Detection • Conflict Resolution • Extended Attribute Modification • Package Description Modification • Package Move/Copy • Patch Analysis
Date	Date and time logged event occurred.
User	User who performed the logged event.
Description	Description providing details of the logged event.

App-V History View

Note • *This information applies to App-V 4.x packages.*

The **App-V History View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **App-V History** node, lists an entry for each time this App-V package was saved.

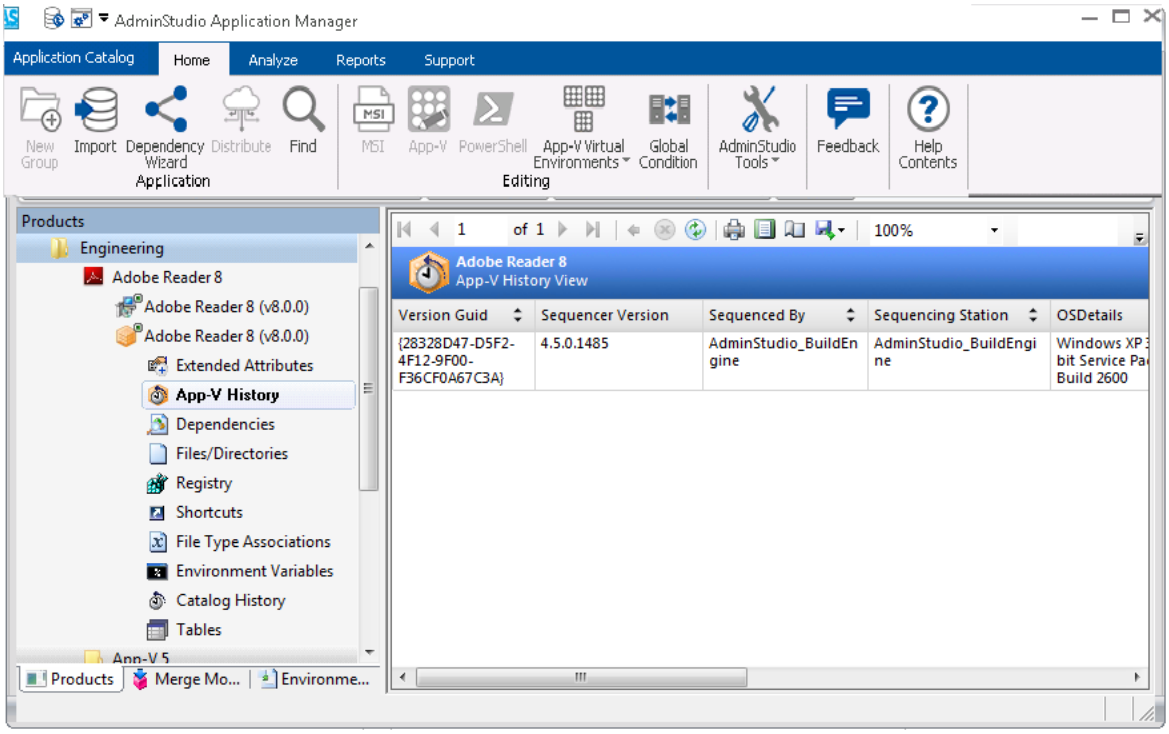


Figure 7-21: App-V History View

For each entry, the following information is displayed:

- Version Guid
- Sequencer Version
- Sequenced By
- Sequencing Station
- OSDetails
- System Folder
- Windows Folder
- User Folder
- .Net Framework Version
- IEVersion

Tables View

The **Tables View**, which opens in the Application Manager **Home** tab when a package's **Tables** node is selected in the tree, provides a way to view table data for Windows Installer packages, macOS packages, App-V packages, and mobile apps in the Application Catalog.

Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the `ASCMPackageJavaSummary` table in the **Tables View**.

Select the specific table you want to view from the **Tables** list at the top of the view.

Note • *Most Windows Installer tables are derived directly from standard MSI tables, as described in the Windows Installer SDK online help. When building your own ACE rules to use for conflict identification, it is important to understand the data available for packages so you can construct the necessary rule.*

File Type Associations View

The **App-V File Type Associations View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **File Type Associations** node, displays a list of this package's file type associations.

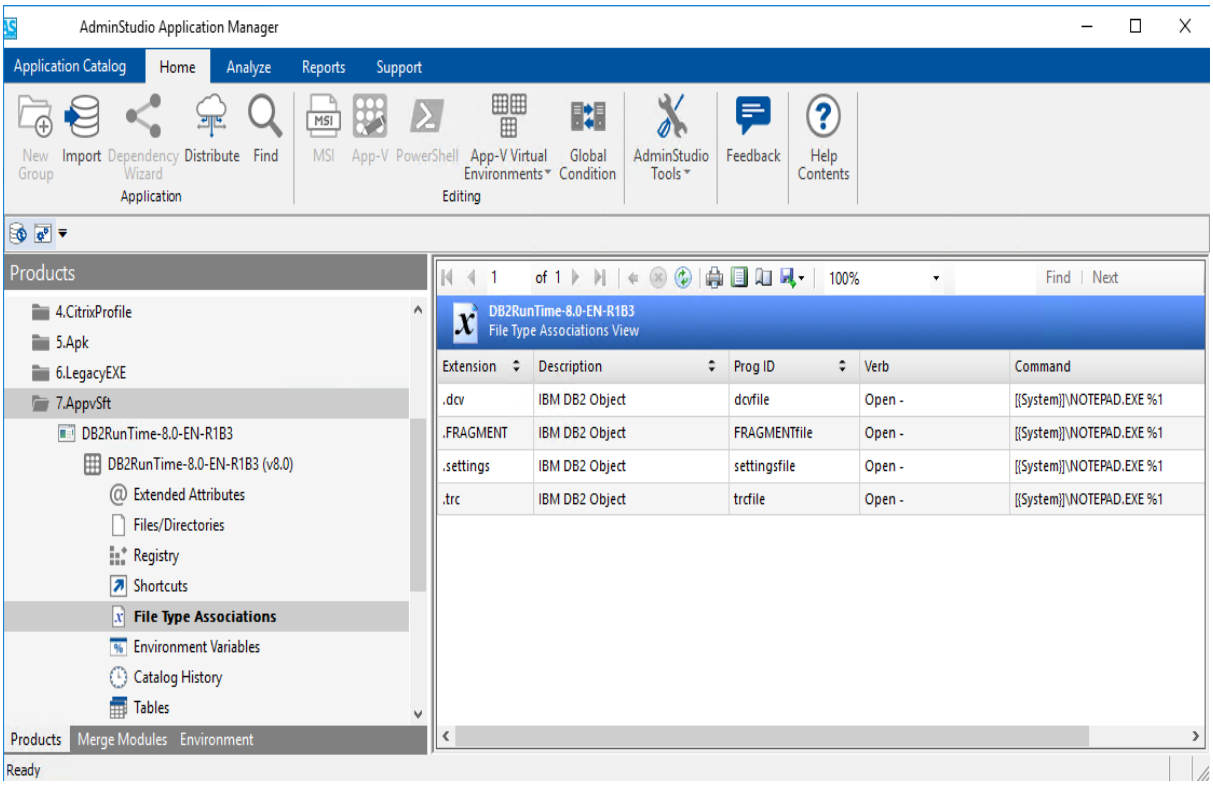


Figure 7-22: App-V File Type Associations View

For each file type association, the following information is listed:

- Extension
- Description

- Prog ID
- Verb
- Command

Environment Variables View

The **App-V Environment Variables View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Environment Variables** node, lists the environment variables used in this App-V package.

For each environment variable, the following information is listed:

- Name
- Value

Merge Module Tree Views

When the **Merge Modules** tab is selected in the Application Manager tree, you can access the following views:

- **All Merge Modules View**—Select this option to view a list of all of the Merge Modules in your catalog. See All Merge Modules View.
- **Merge Module View**—Select an individual Merge Module to see detailed information on that module. See Merge Module View.

All Merge Modules View

The All Merge Modules view is the root node of the merge modules explorer. It contains a list of all merge modules in the Application Catalog, including titles, versions, languages, and identifiers for each module.

Double-click on a merge module to see information about it in the Merge Module View.

Merge Module View

When you select a merge module in the merge modules explorer, details are displayed in this view.

These details include:

Table 7-25 • Merge Module View Information

Field	Description
Version	The version of the Merge Module.
Language	The language that the Merge Module was written for.
Identifier	String that uniquely identifies the Merge Module.
File	The path and file name of the Merge Module file that was imported.

Table 7-25 • Merge Module View Information

Field	Description
Imported On	The date and time the Merge Module was imported.

Click the plus sign next to the Merge Module to view these Merge Module constituent views:

- Components View
- Dependency View
- Exclusion View
- Files View
- Products View

Components View

When you expand a merge module in the merge modules explorer, you can click on Components to display any components created or changed by the merge module.

The following information is displayed for each component included in this merge module:

- Component
- ComponentId
- Directory_
- csFullPath

Dependency View

When you expand a merge module in the merge modules explorer, you can click on Dependency to display any dependencies in the merge module.

The following information is displayed for each dependency included in this merge module:

- ModuleLanguage
- RequiredID
- RequiredLanguage
- RequiredVersion

Exclusion View

When you expand a merge module in the merge modules explorer, you can click on Exclusion to display any exclusions in the merge module.

The following information is displayed for each exclusion included in this merge module:

- ModuleLanguage

- ExcludedID
- ExcludedLanguage
- ExcludedMaxVersion
- ExcludedMinVersion

Files View

When you expand a merge module in the merge modules explorer, you can click on Files to display any files in the merge module.

The following information is displayed:

Table 7-26 • Merge Module Files View Information

Column	Description
Component_	Name of component that this Merge Module file is associated with.
File	Name of this Merge Module file.
FileName	File name of this Merge Module file.
FileSize	Size of this Merge Module file.
Version	Version of this Merge Module file.

Products View

When you expand a merge module in the merge modules explorer, you can click on Products to display any products in the Application Catalog that use the merge module.

The following information is displayed:

Table 7-27 • Merge Module Products View Information

Column	Description
ProductName	Name of product associated with this Merge Module.
ProductVersion	Version of product associated with this Merge Module.
Manufacturer	Manufacturer of the product associated with this Merge Module.

Environment Tree Views

When the **Environment** tab is selected in the Application Manager tree, you can view information about the Security Patches and OS Snapshots that have been imported into the Application Catalog. The following views are available:

Table 7-28 • Environment Tree Views

View	Description
Security Patches Group View	Opens when the root group in the **Environment** tab is selected, and lists all of the groups that have been created in the **Environment** tab.
New Security Patches Group View	All new patches are imported into the **New Security Patches** group, and this view lists all of the patches in that group.
Group View of a Selected Group	Opens when a group other than the root group in the **Environment** tab is selected. For each selected group, a list of all of the patches in that group is displayed.
Patch View	Lists general content information on a selected patch.
OS Snapshot View	Lists information detailing an imported OS Snapshot.

Security Patches Group View

The **Security Patches Group View** opens when the **Security Patches** group of the **Environment** tab is selected. The **Security Patches Group View** lists all groups that have been created in the **Environment** tab.

All new patches are imported into the **New Security Patches** group, and then you can organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Security Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

Shortcut Menu Options

When the **Security Patches** group in the **Environment** tab is selected, the following items are available on the shortcut menu:

- **Refresh**—Refresh the patch listing to reflect the most recent modifications.
- **Import Patches**—Import a Microsoft Operating System Security Patch.
- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

New Security Patches Group View

The **New Security Patches Group View** opens when the **New Security Patches** group on the Application Manager **Environment** tab is selected. All new patches are imported into the **New Security Patches** group. You can then organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Security Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

The **New Security Patches Group View** displays a list of all of the patches in that group, including the following information:

Table 7-29 • New Security Patches Group View Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Shortcut Menu Options

When the **New Security Patches** group in the **Environment** tab is selected, the following items are available on the shortcut menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

Group View of a Selected Group

The Group View of a selected group opens when a group other than the root group or the **New Security Patches** group in the **Environment** tab is selected. For each selected group, a list of all of the patches in that group is displayed, including the following information:

Table 7-30 • Group View of a Selected Group Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.

Table 7-30 • Group View of a Selected Group Information

Option	Description
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Shortcut Menu Options

When a group other than the root group or the **New Security Patches** group in the **Environment** tab is selected, the following items are available on the shortcut menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Delete**—Delete the selected group.
- **Properties**—Open the **Group Properties** dialog box.

Patch View

The **Patch View**, which is displayed when a patch is selected on the **Environment** tab, lists general information on the selected patch. The following information is included:

Table 7-31 • Patch View Information

Option	Description
ID	Microsoft Security Bulletin ID. Click this link to view this bulletin on the Microsoft website.
Title	Title of patch.
Release Date	Date the patch was released by Microsoft.
KB Article	Microsoft Knowledge Base article ID. Click this link to view this article on the Microsoft website.
Imported On	Date patch was imported into the Application Catalog
Groups	List of all of the groups that this patch is included in.
Description	You can enter a description of the patch in this field.

In Application Manager, you can view additional detailed patch information by right-clicking on a patch and then selecting **Properties** from the shortcut menu.

Shortcut Menu Options

When you right-click on a patch in the **Environment** tab, the following items are available on the shortcut menu:

- **Rename**—Rename the selected patch.
- **Delete**—Delete the selected patch.
- **Generate Report**—Generate a Patch Impact Analysis Report for that patch. See Generating the Patch Report.
- **Properties**—Open the **Patch Properties** dialog box for that patch.

OS Snapshot View

When you click on an OS Snapshot in the Application Manager Environment View, details about the snapshot appear in the right pane of the user interface.

The following information is displayed:

Table 7-32 • OS Snapshot View Information

Field	Description
Version	Version of the operating system of the OS Snapshot, such as Windows 10 21H1 - 19043.1237.
Language	The language the operating system was written for.
File	This can be either a hard-coded path or a UNC path.
Imported On	The date and time the OS Snapshot was imported.
Description	You can edit this with additional information about the OS Snapshot.

Click the plus sign next to the OS Snapshot icon to view these OS Snapshot constituent views:

- Files View for OS Snapshots
- INI File Changes View for OS Snapshots
- Registry View for OS Snapshots
- Shortcuts View for OS Snapshots

Files View for OS Snapshots

When you expand an OS Snapshot in the **Environment** tab of the Application Manager tree, you can click on **Files** to display a list of the files included in the OS Snapshot.

The following information is displayed for each of the files included in the OS Snapshot:

Table 7-33 • Files View Information

Column	Description
FileName	Name of the file.
csFilePath	Path
FileSize	Size of the OS Snapshot file.
Version	Version of the OS Snapshot file.
Language	Language that the OS Snapshot file was written for.
Attributes	Any attributes associated with the file.

INI File Changes View for OS Snapshots

When you expand an OS Snapshot in the **Environment** tab of the Application Manager tree, you can click on **INI File Changes** to display any INI file changes made by the snapshot.

The following information is displayed for each change to the INI file that is made by the snapshot:

Table 7-34 • INI File Changes View Information

Column	Description
FileName	Name of INI File that the OS Snapshot makes an entry in.
csFilePath	Path
Section	The section of the INI File where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Registry View for OS Snapshots

When you expand an OS Snapshot in the **Environment** tab of the Application Manager tree, you can click on **Registry** to display any registry entries created or changed by the product.

The following information is displayed for each Registry Entry:

Table 7-35 • Registry View Information

Column	Description
Root	Default value of Key.

Table 7-35 • Registry View Information

Column	Description
Key	Key of the Registry Entry that this component is making.
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Shortcuts View for OS Snapshots

When you expand an OS Snapshot in the **Environment** tab of the Application Manager tree, you can click on **Shortcuts** to display any shortcuts that were found on this OS Snapshot's operating system.

The following information is displayed:

Table 7-36 • Shortcuts View Information

Column	Description
Name	Name of shortcut.
Directory	Location of shortcut.
Target	The application executable that the shortcut points to.

Tables View for OS Snapshots

The **Tables View** for an OS Snapshot is identical to the **Tables View** shown when a package is selected on the **Products** tab of the Application Manager tree. See Tables View in the Application Manager Home Deployment Type View section.

Enterprise Policy View

When you click on an Enterprise Policy in the Application Manager Environment View, the properties of the Enterprise Policy appear in the right pane of the user interface.

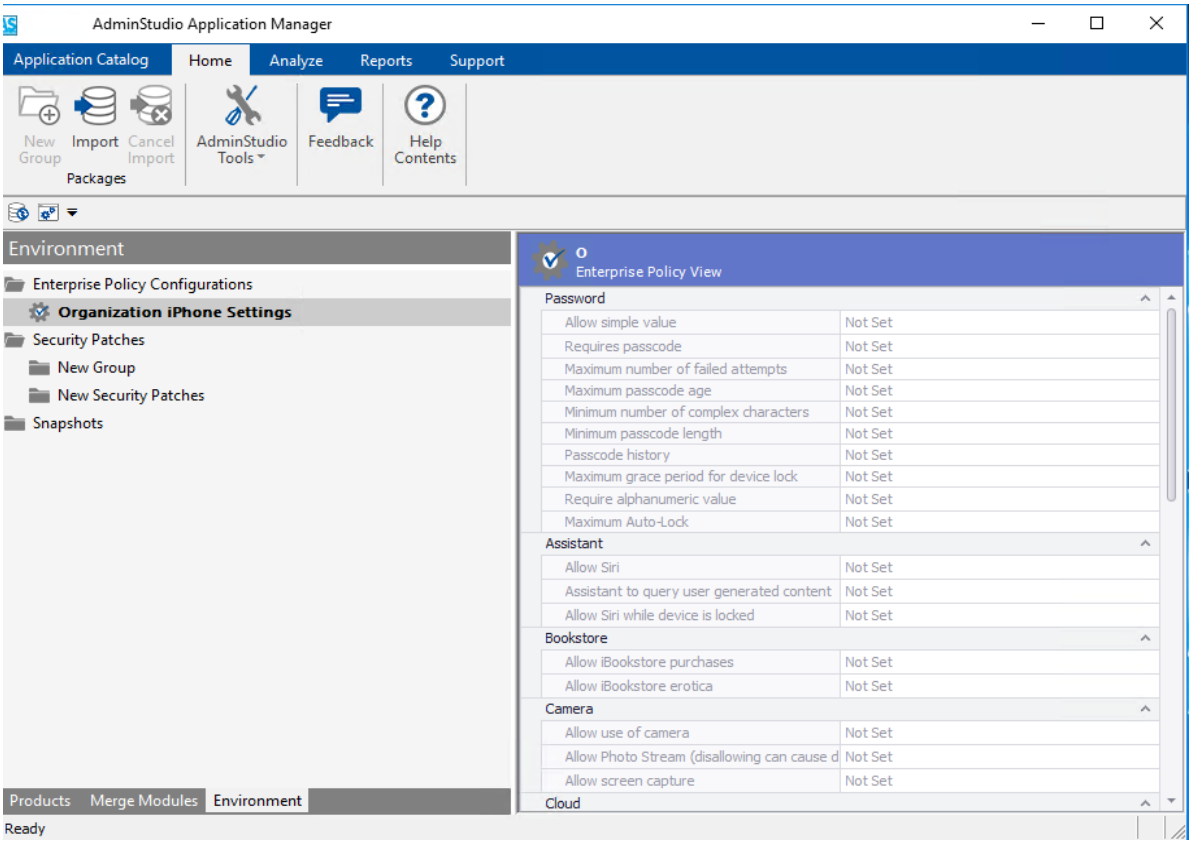


Figure 7-23: Enterprise Policy View

Dialog Boxes

The following dialog boxes are available in Application Manager when the **Home** tab is selected in the ribbon:

- Add Applications Dialog Box
- Add Connection Group Packages Dialog Box
- Add/Edit Applications Dialog Box
- Add/Edit Return Code Dialog Box
- AdminStudio Host Process
- AdminStudio Tools Dialog Box
- App-V Server Connection Groups Dialog Box
- App-V Virtual Environments Dialog Box
- Application Search Results Dialog Box
- Associate with Workflow Manager Workflow Request Dialog Box
- Categories Dialog Box
- Change Deployment Type Priority Dialog Box

- Change Enterprise Server Password Dialog Box
- Configure Connection Group Dialog Box
- Connect Application Catalog Dialog Box
- Create Global Condition Dialog Box
- Create Virtual Environment / Properties Dialog Box
- Default Application Catalog Dialog Box
- Edit Keywords Dialog Box
- Extended Attribute Property Dialog Box
- Find Dialog Box
- Flexera Identifier Dialog Box
- Flexera Local Identifier Dialog Box
- Global Conditions Dialog Box
- Keywords Dialog Box
- Login Required Dialog Box
- Properties Dialog Box
- Options Dialog Box
- References Dialog Box
- SCCM Server Environment Dialog Box
- Select Application Catalog Dialog Box
- Select AdminStudio Enterprise Server URL Dialog Box
- Select Substitute Package Dialog Box
- Servers Dialog Box
- Specify Applications Dialog Box
- Users Dialog Box
- Virtual Package Association Dialog Box
- XML Namespaces Dialog Box

Add Applications Dialog Box

On the **Add Applications** dialog box, which is opened by clicking **Add** on the **Create Virtual Environment** dialog box, you can add an App-V deployment type group to a ConfigMgr (Formerly called as System Center Configuration Manager) Server virtual environment.

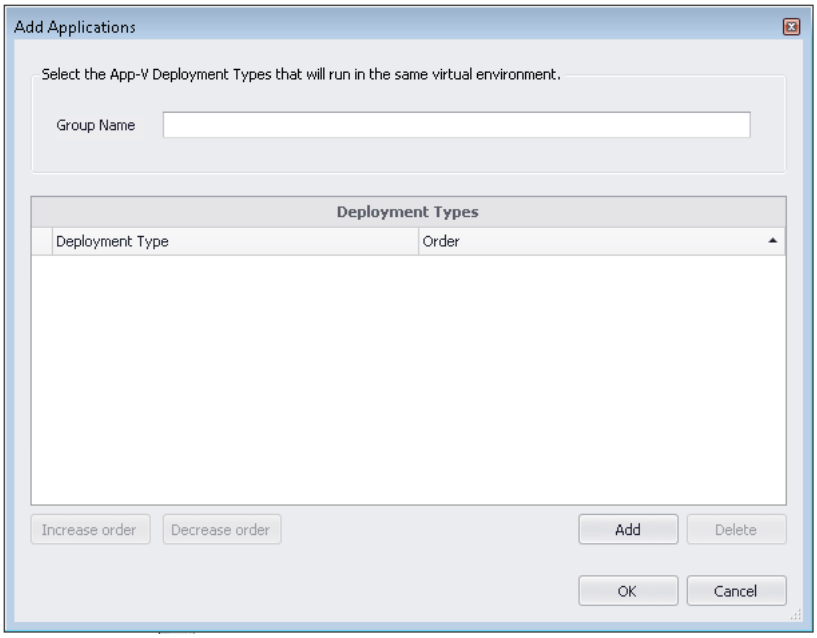


Figure 7-24: Add Applications Dialog Box

The **Add Applications** dialog box includes the following properties:

Table 7-37 • Add Applications Dialog Box Properties

Property	Description
Group Name	Enter a name to identify the group of App-V 5.0 packages that you are going to add.
Deployment Types list	List of **Deployment Types** in this App-V deployment type group, along with their **Order**. Click **Add** to add applications to this list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.
Add	Click to open the **Specify Applications** dialog box, where you can add applications to the deployment type group.
Delete	Click to delete the selected deployment type group.
Increase order **Decrease order**	If more than one deployment type is listed in the **Deployment Types** list, you can use the **Increase order** and **Decrease order** buttons to reorder the list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.

Add Connection Group Packages Dialog Box

On the **Add Connection Group Packages** dialog box, which is opened by clicking **Add** on the **Configure Connection Group** dialog box, you select App-V 5.0 packages to add to an App-V Server Virtual Environment.

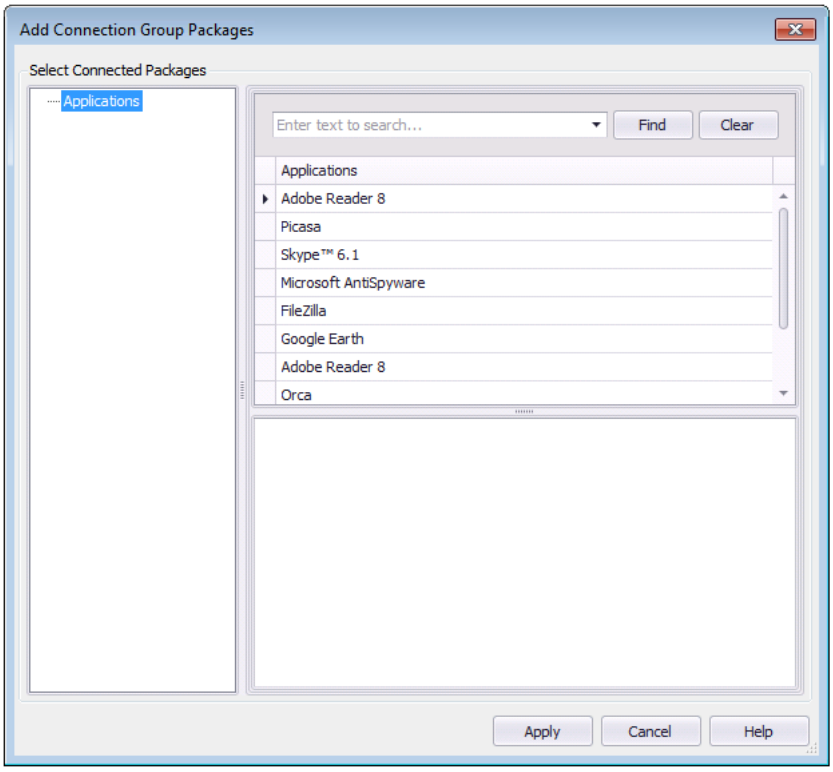


Figure 7-25: Add Connection Group Packages Dialog Box

The **Add Connection Group Packages** dialog box includes the following properties:

Table 7-38 • Add Connection Group Packages Dialog Box Properties

Property	Description
Select Connected Packages tree	Expandable tree listing the groups in the Application Catalog. When you select a group, all of the applications in the group are listed in the **Applications** list.
Search box	Use search the application in the **Applications** list.
Applications list	List of all of the applications in the selected Application Catalog group. Select an application to display its App-V 5.0 packages in the lower pane. ***Note*** • *If you select an application that does not have an App-V 5.0 deployment type, nothing will be listed in the lower pane.*
Apply	Click to add the selected App-V 5.0 package to the connection group.

Add/Edit Applications Dialog Box

On the **Add/Edit Applications** dialog box, which is opened by clicking **Add** or **Edit** on the **Create Virtual Environment** dialog box, you can enter a group name to identify the group of App-V 5.0 packages that are going to be members of the virtual environment, and add deployment types to the virtual environment.

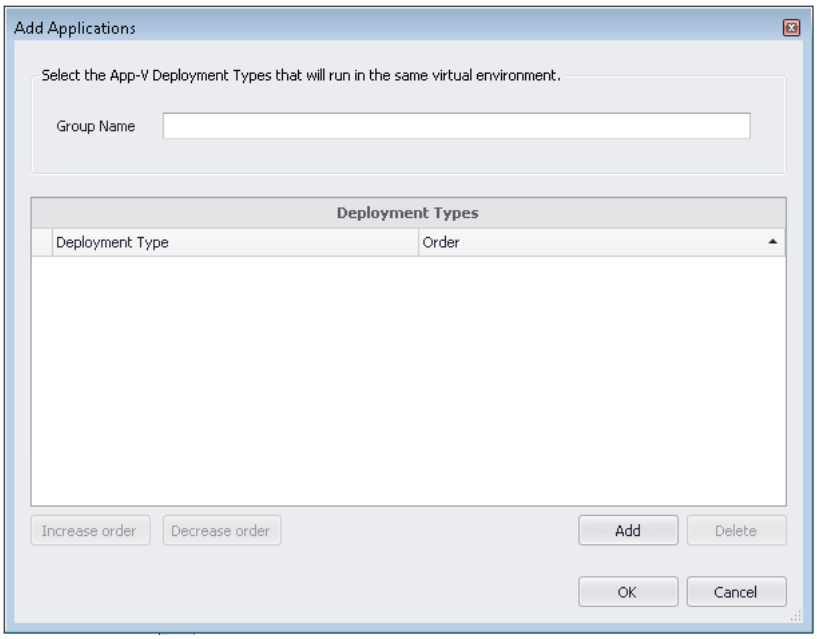


Figure 7-26: Add/Edit Application Dialog Box

The Add/Edit Application dialog box includes the following properties:

Table 7-39 • Add/Edit Application Dialog Box

Property	Description
Group Name	Name that identifies this group of App-V 5.0 packages.
Deployment Types	List of deployment types (App-V 5.0 packages) that are members of this virtual environment.
Add	Click to open the Specify Applications dialog box, where you can select an App-V 5.0 package to add to this virtual environment.
Delete	Click to delete the selected deployment type (App-V 5.0 package).
Increase order **Decrease order**	If more than one deployment type is listed, you can use the **Increase order** and **Decrease order** buttons to reorder the list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.

Add/Edit Return Code Dialog Box

Return codes are used to indicate whether a restart is required, whether an installation is complete, and to customize the text shown to users when a specific code is returned. A package's return codes are populated by default when the package is imported.

You can edit a MSI and EXE package's return codes on the **Return Codes** subtab of the **Deployment Data** tab of the **Home Deployment Type View**. If you click the **Add Return Code** or **Edit Return Code** button in the **Deployment Data** tab ribbon, the **Add/Edit Return Code** dialog box opens.

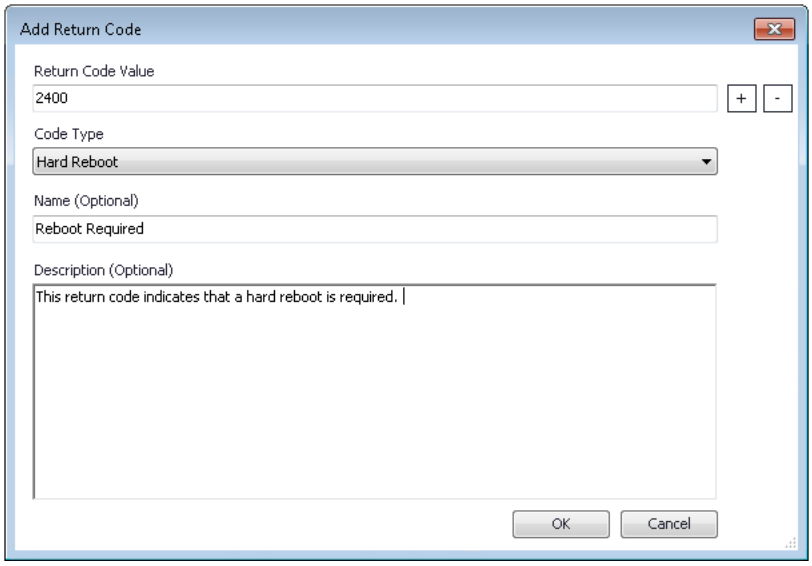


Figure 7-27: Add Return Code Dialog Box

The **Add/Edit Return Code** dialog box includes the following properties:

Table 7-40 • Add/Edit Return Code Dialog Box

Property	Description
Return Code Value	Enter a unique value. The following return codes are populated by default during import: • **0**—Success (no reboot) • **1707**—Success (no reboot) • **3010**—Soft Reboot • **1641**—Hard Reboot • **1618**—Fast Retry ***Note*** *• When you are editing an existing return code, this field cannot be edited.*

Table 7-40 • Add/Edit Return Code Dialog Box

Property	Description
Code Type	Select one of the following options to identify the return code's type: • **Success (no reboot)** • **Failure (no reboot)** • **Hard Reboot** • **Software Reboot** • **Fast Retry**
Name (Optional)	Enter a name to identify this return code.
Description (Optional)	Enter text to explain the meaning of this return code.

AdminStudio Host Process

Starting with AdminStudio 2013, most of Application Manager's core functionality now resides in the AdminStudio Host Process, separate from its user interface.

Using a host process gives AdminStudio better scalability and enables the development of clients that use Application Manager's core functionality. For example the Application Manager user interface and the AdminStudio PowerShell Cmdlet are now clients to this AdminStudio Host process.

Important • *AdminStudio Host must be running in order to use Application Manager or the AdminStudio PowerShell Cmdlets.*

When Application Manager is launched, the AdminStudio Host process is automatically started, and its icon is added to the System Tray.

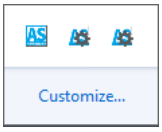


Figure 7-28: AdminStudio Host Icon in System Tray

AdminStudio Tools Dialog Box

On the **AdminStudio Tools** dialog box, which is opened by clicking the **AdminStudio Tools** button in the Application Manager ribbon. From this dialog box you can use the Repackager, Distribution Wizard, OS Snapshot Wizard, Process Template Editor, Process Assistants, Automated Application Converter, Virtual Package Editor, Tuner, and Editor.

Property	Description
AdminStudio Tools	List of AdminStudio Tools.
Repackager	To open the Repackaging Wizard and repackage legacy setups and perform tasks such as the following: ● Open the Exclusions Editor and configure exclusions. ● Convert Microfocus ZENworks, Microsoft SMS, and WinINSTALL projects into Repackaging projects. ● Create a package exclusion li ● Build a Repackager project into an InstallShield Editor project and Windows Installer package.
Distribution Wizard	To publish an application or group of applications from the Application Catalog to a distribution system.
OS Snapshot Wizard	To create a snapshot of your current operating system configuration.
Process Template Editor	To create and modify workflows for all the projects in AdminStudio.
Process Assistants	To create, execute, and delete existing/new project (s).
Automated Application Converter	To combine the functionality of the Windows Installer Batch Converter with the additional capability to automatically repackage and convert Windows Installer packages.
Virtual Package Editor	To edit App-V packages and perform tasks such as the following: ● Customize your App-V applications. ● Resolve virtualization best practice issues and application conflicts. ● Fix run-time problems.
Tuner	To add, modify, or remove information from a Windows Installer package.
Editor	To create, add, or modify information in InstallShield.

App-V Server Connection Groups Dialog Box

App-V Server virtual environments are called connection groups. You can create App-V 5.0 connection groups to generate connections between virtualized applications that allow the applications to communicate with each other while they run in the virtual environment.

The **App-V Server Connection Groups** dialog box, which is opened by clicking the **App-V Virtual Environments** button in the ribbon and selecting **App-V Server Environment**, lists the connection groups that have already been defined. You click **Add** on this dialog box to create a new connection group or **Edit** to edit an existing connection group.

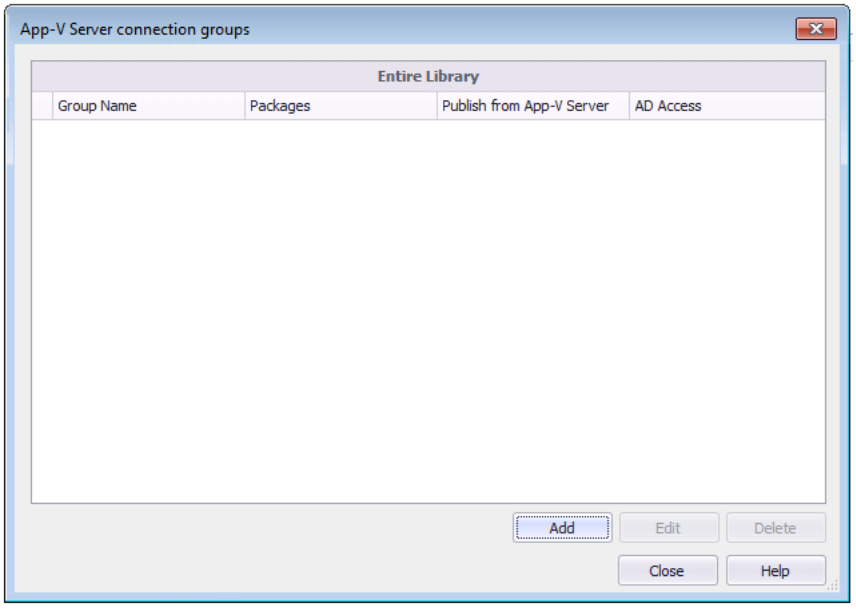


Figure 7-29: App-V Server Connection Groups Dialog Box

Note • *You can also open the **App-V Server Connection Groups** dialog box by selecting an App-V 5.0 package in the tree, and then clicking in the **Connection Group** field on the **Home Deployment Type View > App-V Deployment Data > Advanced Settings** tab.*

The App-V Server Connection Group dialog box includes the following properties:

Table 7-41 • App-V Server Connection Group Dialog Box

Property	Description
Group Name	Name that identifies the connection group.
Packages	List of packages in the connection group.
Publish from App-V Server	Icon indicating whether the **Publish from App-V Server** option is set to **True** or **False**. A checked checkbox indicates that it was set to **True**, while an empty checkbox indicates that it is set to **False**.
AD Access	Active Directory group that has access to this connection group.
Add	Click to open the **Configure Connection Group** dialog box where you can create a new connection group.
Edit	Click to open the **Configure Connection Group** dialog box where you can edit the selected connection group.

App-V Virtual Environments Dialog Box

On the **App-V Virtual Environments** dialog box, which is opened by clicking the **App-V Virtual Environment** button in the Application Manager ribbon, existing App-V virtual environments are listed. From this dialog box you can add a new virtual environment or edit an existing one.

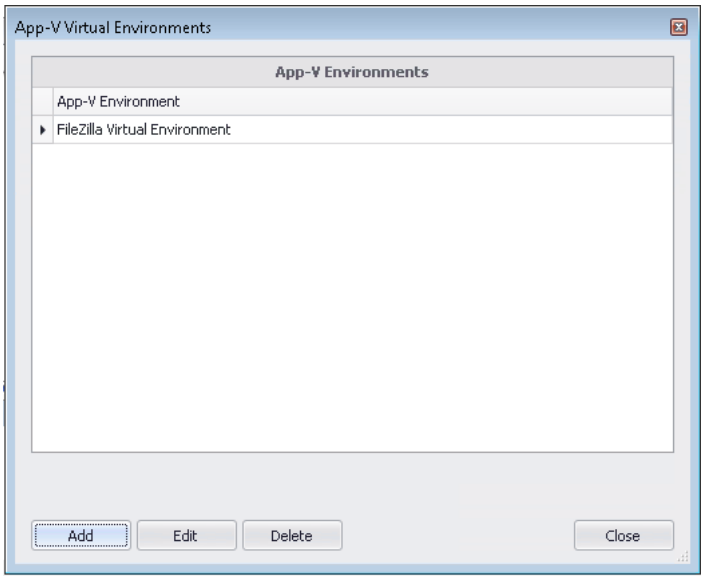


Figure 7-30: App-V Virtual Environments Dialog Box

App-V virtual environments in Microsoft System Center 2012 Configuration Manager enable deployed virtual applications to share the same file system and registry on client computers. This means that unlike standard virtual applications, these applications can share data with each other.

Tip • *Using virtual environments to group dependent packages together in App-V 5.0 is similar to the Dynamic Suite Composition feature used with App-V 4.x packages.*

The App-V Virtual Environments dialog box includes the following properties:

Table 7-42 • App-V Virtual Environments Dialog Box

Property	Description
App-V Environment	List of existing App-V virtual environments.
Add	Click to open the **Create Virtual Environment** dialog box, where you can create a new Virtual Environment. See Create Virtual Environment / Properties Dialog Box.
Edit	Select a virtual environment in the list and click to open its **Properties** dialog box, where you can edit its settings.

Application Search Results Dialog Box

When you click **Unrecognized Applications** in the toolbar of the Application Manager **Home** tab, the **Application Search Results** dialog box opens, listing all applications in the Application Catalog that do not have an associated Flexera Identifier

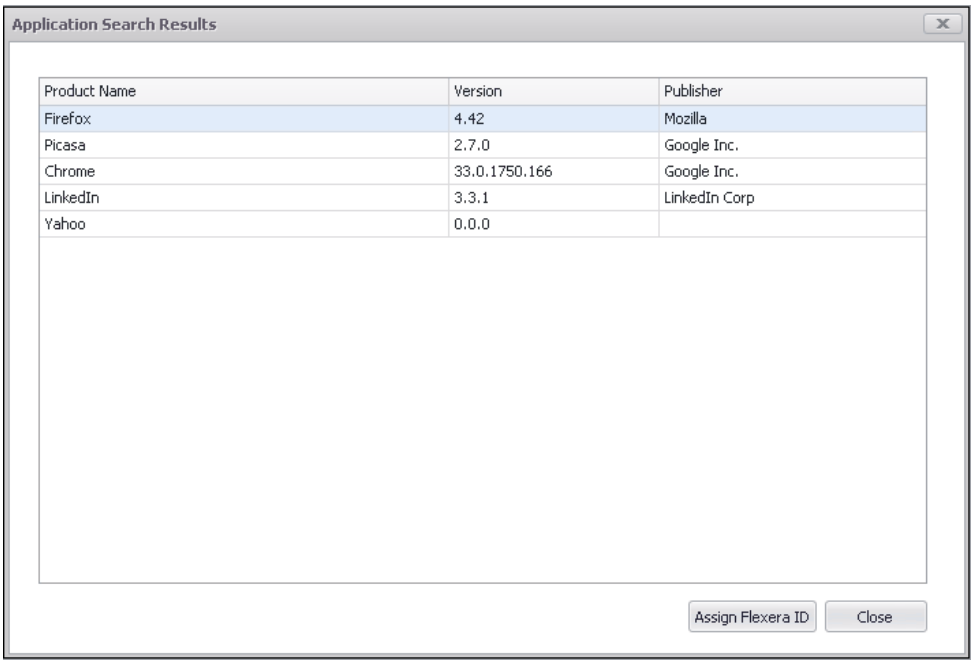


Figure 7-31: Application Search Results Dialog Box

To search for an existing Flexera Identifier or create a local Flexera Identifier, select an application in the list and click **Assign Flexera ID** to open the **Flexera Identifier** dialog box.

Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

Associate with Workflow Manager Workflow Request Dialog Box

Note • *AdminStudio Workflow Manager is a Web-based application management system that has integrated functionality with AdminStudio.*

The **Associate with Workflow Manager Workflow Request** dialog box is displayed when you right-click on a package in the Application Manager tree and then select **Associate with Workflow Manager Workflow Request** from the shortcut menu. It allows you to associate extended attribute data for a package with a Workflow Manager workflow request.

Note • *This dialog (and its corresponding command) are only available if you select the **Integrate with Workflow Manager** option on the **General** tab of the Application Manager Options dialog box.*

Note • *Be sure to select a Workflow Manager workflow request that uses a workflow template that contains at least one major data group that was specified with the group's extended attribute description file, as described in the AdminStudio Workflow Manager user documentation.*

Categories Dialog Box

On the **Categories** dialog box, which is opened by clicking the browse button in the **Categories** field on the **App Portal Information** tab of the **Application View**, you can specify whether to create an App Portal catalog item when this application is published to System Center 2012 Configuration Manager or Symantec Altiris Server, and you can also specify the category or categories that you want the application's catalog item to be placed in.

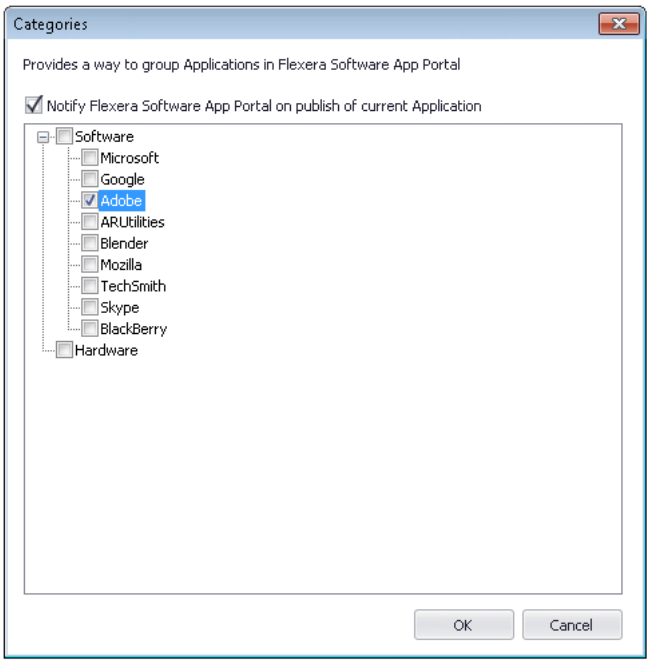


Figure 7-32: Categories Dialog Box

When an end user browses the App Portal catalog on the **Browse Home** tab, catalog items are organized into categories. In the image below, the **Software > Google** category contains two catalog items.

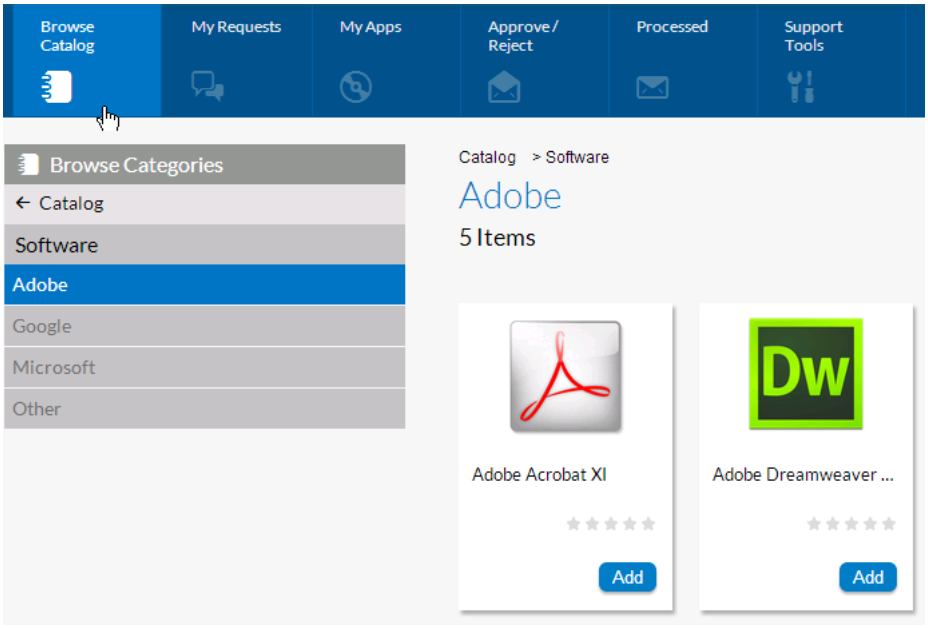


Figure 7-33: Categories Displayed on the App Portal Browse Home Tab

When a catalog item is created in App Portal, a category must be specified. On the **Categories** dialog box, you can select the category or categories that you want the application's catalog item to be placed in.

The **Categories** dialog box includes the following options:

Table 7-43 • Categories Dialog Box

Option	Description
Notify Flexera App Portal on publish of current Application	If you want a new catalog item to be created in App Portal when this application is published to System Center 2012 Configuration Manager or Symantec Altiris Server, select this option.
Category list	All of the categories that have been defined in App Portal are listed. Select the category or categories that you want this application's App Portal catalog item to be placed in.

Change Deployment Type Priority Dialog Box

When an application has multiple deployment types, the order in which they will be evaluated in System Center 2012 Configuration Manager depends upon the deployment type's assigned priority. When a deployment type meets the specified requirements, it will be run and then no further deployment types on the priority list will be evaluated. By default, Application Manager assigns a deployment type a priority based upon their import order.

You can modify the priority setting of an application's deployment types on the **Change Deployment Type Priority** dialog box, which is opened by clicking the **Change Priority** button in the **Deployment Types** tab ribbon.

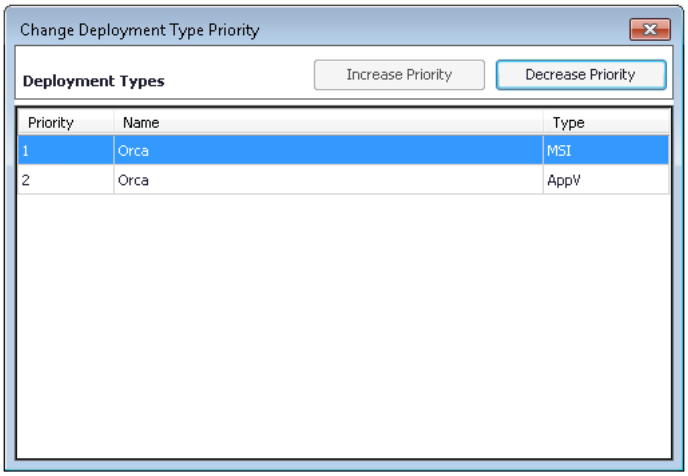


Figure 7-34: Change Deployment Type Priority Dialog Box

Just select the deployment type and click **Increase Priority** or **Decrease Priority** to move it up and down in the list.

Note • *You can only assign a priority to Windows Installer, App-V, and .exe packages. All other packages are assigned a priority of -1, which cannot be changed.*

Change Enterprise Server Password Dialog Box

This dialog box, which is opened by selecting **AdminStudio Enterprise Server > Change AES Password** on the Application Catalog tab menu, allows you to change your password to connect to the AdminStudio Enterprise Server Application Catalog (which is the same password you use to log in to the AdminStudio Enterprise Server). The **Change AES Password** selection is enabled when you are connected to the AdminStudio Enterprise Server Application Catalog.

Note • *If you are not connected to the AdminStudio Enterprise Server Application Catalog, the **Change AES Password** selection is disabled.*

Table 7-44 • Change Enterprise Server Password Dialog Box

Option	Description
User Name	(Read Only) User name of user who is connected to the AdminStudio Enterprise Server Application Catalog.
Old Password	Enter the existing password for current user.
New Password	Enter the new password.
Confirm New Password	Enter the new password a second time.

Configure Connection Group Dialog Box

On the **Configure Connection Group** dialog box, which is opened by clicking **Add** on the **App-V Server Connection Groups** dialog box, you enter the properties for an App-V Server connection group and add App-V 5.0 packages to it.

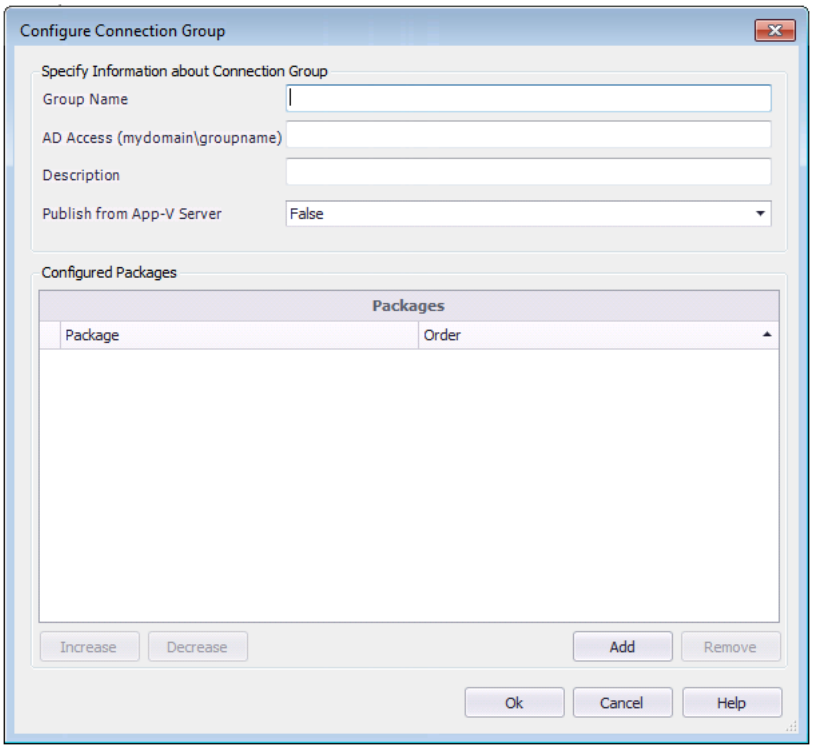


Figure 7-35: Configure Connection Group Dialog Box

The **Configure Connection Group** dialog box includes the following properties:

Table 7-45 • Configure Connection Group Dialog Box Properties

Property	Description
Group Name	Enter a name to identify this App-V Server connection group.
AD Access	Enter the name of the Active Directory group that will have permission to access this connection group.
Description	Enter a description of this connection group.
Publish from App-V Server	Select **True** or **False** to specify whether to publish this connection group from the App-V Server.
Configured Packages	Lists the App-V 5.0 packages that have been added to this connection group. To add an App-V 5.0 package to this list, click Add.
Add	Click to **Add** to open the **Add Connection Group Packages** dialog box, where you can add App-V 5.0 packages to this connection group.

Table 7-45 • Configure Connection Group Dialog Box Properties

Property	Description
Remove	Click to remove the selected App-V 5.0 package from this connection group.
Increase **Descrease**	The order of packages in the connection group is important. This determines the order in which the package contents are merged. So, if there was a conflict (example: same registry value), the content of the first package would be used. Use the **Increase** and **Decrease** buttons to change the order of the selected package.

Connect Application Catalog Dialog Box

The Connect Application Catalog dialog box opens when you choose to open an existing Application Catalog. This dialog box has three tabs:

- **Enterprise Server**—Select this tab to open the AdminStudio Enterprise Server Application Catalog database. See Enterprise Server Tab.
- **Standalone**—Select this tab to open an Application Catalog database other than the AdminStudio Enterprise Server Application Catalog. See Standalone Tab / Specify Database Information.
- **Recent**—Provides a list of recently opened Application Catalogs. When you select an Application Catalog and click **OK**, either the Application Catalog opens or you are prompted for login information (if you need authentication to the Application Catalog). See Recent Tab.

Making this the Default Shared Application Catalog

If you select the **Make this the default shared Application Catalog** option, the Application Catalog you are opening will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory).

If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Note *• In the AdminStudio Enterprise Edition, only the AdminStudio Administrator or users with the Change Default Database permission will see the* ***Make this the default shared Application Catalog*** *option. This allows the AdminStudio Administrator to configure the default Application Catalog, and then subsequent installations of AdminStudio will automatically connect to the default Application Catalog if they use the same shared directory.*

Enterprise Server Tab

To connect to the AdminStudio Enterprise Server Application Catalog, you log in on the **Enterprise Server** tab of the **Connect Application Catalog** dialog box.

Table 7-46 • Connect Application Catalog / Enterprise Server Tab Options

Option	Description
URL	Location of the AdminStudio Enterprise Server associated with this installation of AdminStudio. If the AdminStudio Enterprise Server has not yet been configured with the AdminStudio client tools (such as when it is set to its default value of `http://localhost`), click the URL link to open the **Select AdminStudio Enterprise Server URL** dialog box, and enter the URL for location of the AdminStudio Enterprise Server associated with this installation of AdminStudio.
Authentication	Select one of the following options: • Windows Authentication • AdminStudio Enterprise Server User ***Note*** *• When using **AdminStudio Enterprise Server User** authentication, if Anonymous authentication is turned off in IIS, both the user's machine and the AdminStudio Enterprise Server need to be on the same domain in order for login to succeed.*
User Name and Password	If you selected **AdminStudio Enterprise Server User**, enter your AdminStudio Enterprise Server `User Name` and **Password** (provided by your System Administrator).

Login Troubleshooting

If you are using a Web Portal with custom security zone settings, your AdminStudio Enterprise Server URL is using an IP address, and you receive Error 0x800A1518 when you attempt to login, change the AdminStudio Enterprise Server URL to the NetBios equivalent and then try again. For example, if you are connecting to `http://120.12.1.15`, the NetBios equivalent would be `http://wfmportal`.

Standalone Tab / Specify Database Information

On the **Standalone** tab of the **Connect Application Catalog** dialog box and the **Specify Database Information** panel of the Application Catalog Wizard, you have the following two options:

- Azure SQL
- On-Prem SQL

Azure SQL

Enter the information required to login to the specified Application Catalog.

Table 7-47 • Connect Application Catalog / Standalone Tab / Azure SQL Option

Option	Description
Server	Select one of the available Azure SQL Servers on the network from this list. You can also manually enter the name of the server to which you want to connect.
Authentication	Select one of the following options: • **Server Authentication**—Choose to use Azure SQL Server login identification for authentication. If you chose this option, enter the appropriate Login ID and Password. • **Azure Active Directory - Password—**Choose to use Azure Active Directory login identification for authentication. If you chose this option, enter the appropriate Login ID and Password.
Catalog	Do one of the following: • If you are connecting to an **existing** Application Catalog, select the catalog from those available on the Server. • If you are creating a **new** Application Catalog, enter a name for this new catalog.
Test	Click this button to test whether a connection can be made to the database.

On-Prem SQL

Enter the information required to login to the specified Application Catalog or enter the name of the Application Catalog that you are creating.

Table 7-48 • Connect Application Catalog / Standalone Tab / On-Prem SQL Options

Option	Description
Server	Select one of the available SQL Servers on the network from this list. You can also manually enter the name of the SQL Server to which you want to connect.
Authentication	Select one of the following options: • **Windows Authentication**—Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Server Authentication**—Choose to use SQL Server login identification for authentication. • **Login ID and Password**—If you chose Server Authentication, enter the appropriate Login ID and Password.

Table 7-48 • Connect Application Catalog / Standalone Tab / On-Prem SQL Options (cont.)

Option	Description
Catalog	Do one of the following: ● If you are connecting to an **existing** Application Catalog, select the catalog from those available on the Server. ● If you are creating a **new** Application Catalog, enter a name for this new catalog.
Test	Click this button to test whether a connection can be made to the database.
Make this the default shared Application Catalog Connect Application Catalog Dialog Box Only	When this option is selected, the Application Catalog you are trying to open or create will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory). If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Recent Tab

The **Recent** tab displays a list of all Application Catalogs that have recently been open. To login to one of these Application Catalogs, select it and click OK.

- If you are opening a standalone SQL Server Application Catalog, you will be prompted for login information.
- If you are opening a Azure SQL Server or Azure Active Directory Application Catalog, you will be prompted for login information.
- If you are opening the AdminStudio Enterprise Server Application Catalog, you are prompted for AdminStudio Enterprise Server login information before the Application Manager will open.

Create Global Condition Dialog Box

The **Create Global Condition** dialog box, which is opened by clicking the **Create** button on the **Create Custom Requirements** panel of the **Requirements Wizard**, is used to create a condition to use in the custom requirement.

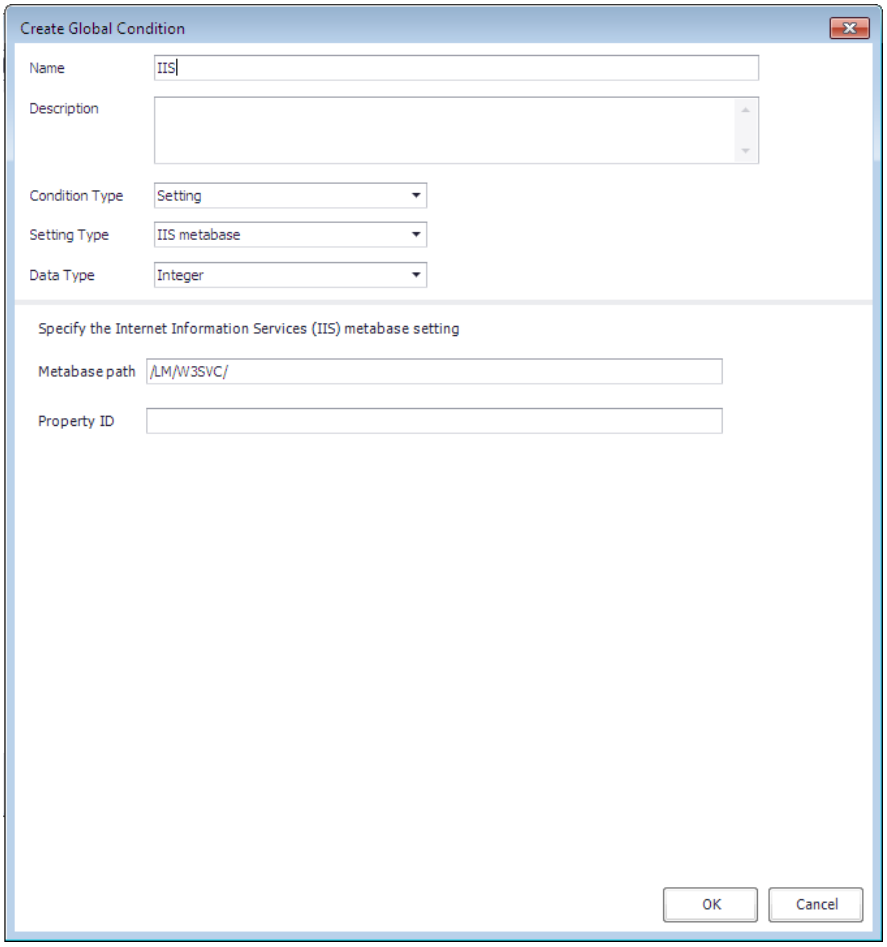


Figure 7-36: Create Global Condition Dialog Box

The fields displayed on this dialog box depend upon what is selected in the **Setting Type** field. The **Create Global Condition** dialog box includes the following properties:

Table 7-49 • Create Global Condition Dialog Box Properties

Property	Description
Name	Enter a name for this global condition.
Description	Enter a description to identify the purpose of this global condition.
Condition Type	Select one of the following options: • **Setting**—Select to create a standard condition. • **Expression**—Select to create a condition which contains expressions, as described in Building Expressions.

Table 7-49 • Create Global Condition Dialog Box Properties

Property	Description
Setting Type	Select one of the following options to identify the type of global condition that you are creating (the item you want this condition to assess for compliance): • **Active Directory query** - Select to specify a condition to assess a Active Directory query • **Assembly** - Select to specify a condition to assess an assembly from the global assembly. • **File system**—Select to specify a condition to assess a file or folder. • **IIS metabase**—Select to specify a condition to assess an IIS metabase. • **Registry key**—Select to specify a condition to assess a registry key. • **Registry value**—Select to specify a condition to assess a registry value. • **Script**—Select to specify a condition to find a script and return a value to be assessed. • **SQL query**—Select to specify a condition to assess an SQL query. • **Wql query**—Select to specify a condition to assess a Windows Management Instrumentation Query Language (WAL) script. • **XPath query**—Select to specify a condition to assess an XML file and XPath query.
Data Type	Select one of the following options to identify this condition's data type: • **String** • **Date and Time** • **Integer** • **Floating point** • **Version** • **Boolean** • **String array** • **Integer array** ***Note*** *• Only displayed for conditions with a* ***Setting Type*** *of* ***IIS metabase***, ***Registry value***, ***Script***, ***Sql query***, ***Wql query***, *and* ***XPath query****. Also, the number* ***Data Type*** *options that are listed depends upon the* ***Setting Type*** *selected.*

The rest of the properties on the **Create Global Condition** dialog box are dependent upon the **Setting Type** selection:

Table 7-50 • Create Global Condition Dialog Box / Properties for Each Setting Type

Setting Type	Property	Description
Active Directory query	**LDAP Prefix**	Enter Valid LDAP Prefix in this format **<Prefix Name>://**
	Distinguished Name	Enter the distinguished name
	Search Filter	Enter a valid search filter
	Search Scope	Select one of the following options: • Base • One Level • Subtree
	Property	Enter Property value for a specified DN
	Query	Displays the generated query based on the above inputs
Assembly	**Assembly Name**	Enter a valid Assembly Name
File system	**Type**	Select either **File** or **Folder**.
	Path	Enter the path to the file or folder you want to use to assess for compliance on computers.
	File or folder name	Enter the file or folder name.
	Include subfolders	Select to include the file or folder's subfolders in the condition.
	This file or folder is associated with a 64 bit application	Select if the specified file or folder is associated with a 64-bit application.
IIS metabase	**Metabase path**	Enter the path to the IIS metabase. ***Note*** *• The metabase is a structure for storing Microsoft IIS configuration settings. It performs some of the same functions as the Windows system registry but is specific to IIS.*
	Property ID	Enter the property ID of the specified IIS metabase.

Table 7-50 • Create Global Condition Dialog Box / Properties for Each Setting Type

Setting Type	Property	Description
Registry key	**Hive**	Select the hive of the registry key that you want to use in this condition.
	Key	Enter the registry key that you want to use in this condition.
	This registry key is associated with a 64-bit application	Select if the specified registry key is associated with a 64-bit application.
Registry value	**Hive**	Select the hive of the registry key that contains the value that you want to use in this condition.
	Key	Enter the registry key that contains the value that you want to use in this condition.
	Value	Enter the registry value that you want to use in this condition.
	This registry value is associated with a 64-bit application	Select if the specified registry value is associated with a 64-bit application.
Script	**Script Type**	Select one of the following options: • **PowerShell** • **VBScript** • **JScript**
	Script box	Click **Browse** and select the script that you want to use in this condition. After you have selected a script, the contents of that script will be listed in this box.
	Run script by using the logged on user credentials	Select this option if you want to run the script using the credentials of the logged on user.

Table 7-50 • Create Global Condition Dialog Box / Properties for Each Setting Type

Setting Type	Property	Description
Sql query	**SQL server instance**	Select one of the following options to specify the server instances you want to use in this condition: • **Use default instance** • **All instances** • **Instance name**
	Database	Specify database.
	Column	Specify column.
	Transact SQL statement	Enter SQL statement.
Wql query	**Namespace**	Enter the namespace that contains the WQL script.
	Class	Enter class.
	Property	Enter property.
	WQL query WHERE Clause	Enter WQL query.
XPath query	**Path**	Enter path.
	File Name	Enter file name.
	Include sub folders	Select to include subfolders.
	This file is associated with 64 bit application	Select to indicate that this file is associated with a 64-bit application.
	XPath Query	Click the **Namespace** button to open the **XML Namespaces** dialog box, where you can specify the XML namespaces and prefixes that you want to use when this XPath query runs. Click **Open** to select a text or XML file containing an XPath query. Click **Clear** to clear the text box. ***Note*** *• For information on adding a namespace, see XML Namespaces Dialog Box.*

Building Expressions

To create a global condition that uses expressions, select **Expression** from the **Condition Type** list. When you select **Expression** from this list, an expression builder interface is displayed.

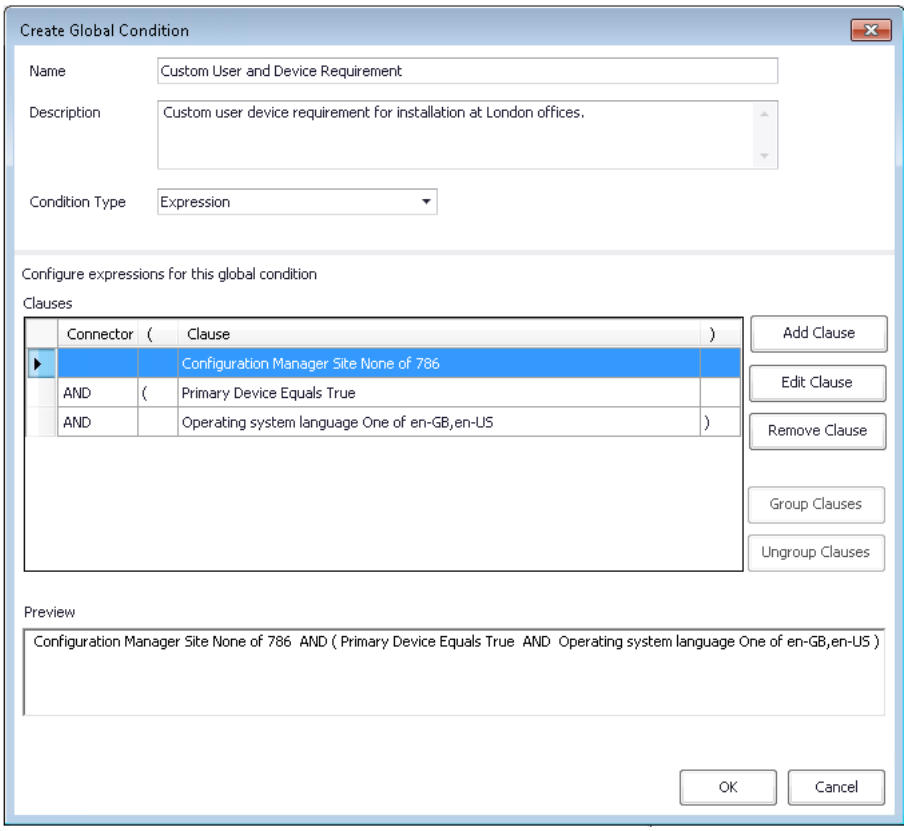


Figure 7-37: Building an Expression on the Create Global Condition Dialog Box

You can use the expression builder interface to form an expression using existing User/Device/Custom requirements. After you add multiple requirements, you can then connect them using **AND** or **OR** operators, and can group sets of clauses, which enables you to create complex requirements.

The expression building area of this dialog box includes the following options:

Table 7-51 • Create Global Condition Dialog Box

Option	Description
Add Clause	Click to open the Requirement Wizard, which you can use to add a User/Device/Custom requirement. When you click Finish on the wizard, the new requirement will be listed in the **Clauses** list. When you add the first requirement the **Connector** will be set to **None**. When adding subsequent requirements, the **Connector** will be set to **AND** by default.
Edit Clause	Click to edit the selected requirement using the Requirement Wizard.
Remove Clause	Click to delete the selected requirement.

Table 7-51 • Create Global Condition Dialog Box

Option	Description
Group Clauses	Click to group the selected requirements (if the grouping criteria matches). If grouping is successful, then the selected requirements will be marked as grouped and parentheses will be displayed the **(** and **)** columns.
Ungroup Clauses	Click to ungroup the selected requirements, if the ungroup criteria matches.
Preview	Lists the full expression.

Create Virtual Environment / Properties Dialog Box

On the **Create Virtual Environment** dialog box, which is opened by clicking **Add** on the **SCCM Server Environment** dialog box, you build a new virtual environment. On this dialog box, you enter the name and description of the virtual environment, and specify the groups of deployment types that will be included in this virtual environment.

Note *• When you select an existing virtual environment on the* ***SCCM Server Environment*** *dialog box and click* ***Edit****, this same dialog box opens, displaying the settings of the selected virtual environment, but its name is now* ***[Virtual_Environment_Name] Properties****.*

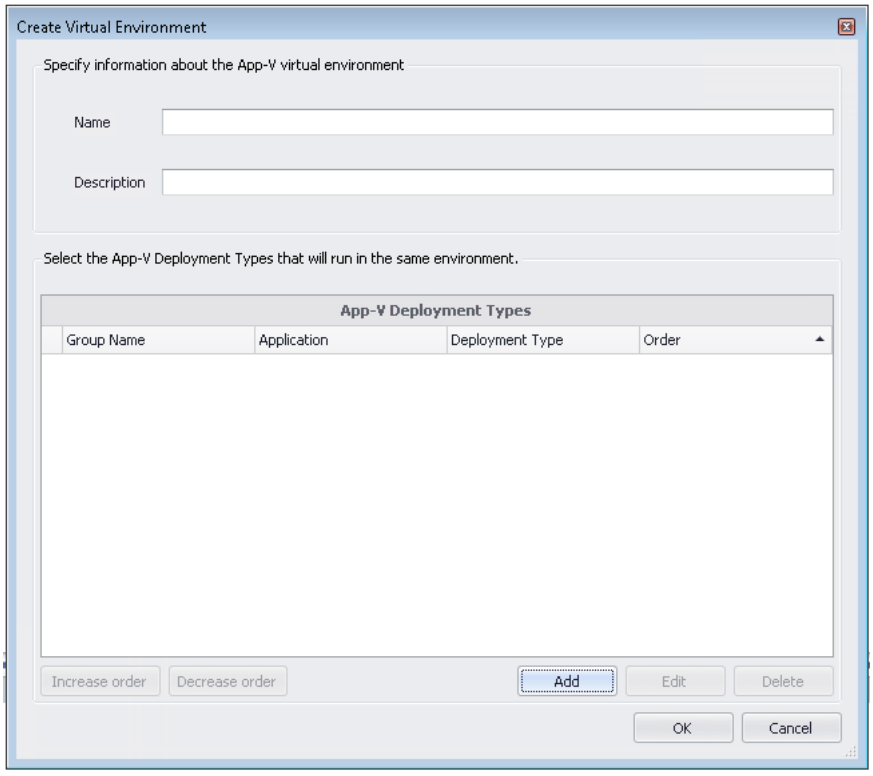


Figure 7-38: Create Virtual Environments Dialog Box

The Create Virtual Environments dialog box includes the following properties:

Table 7-52 • Create Virtual Environments Dialog Box

Property	Description
Name	Enter a name to identify this virtual environment.
Description	Enter a description of the purpose of this virtual environment.
App-V Deployment Types list	List of App-V deployment type groups that have been added to this virtual environment.
Add	Click to open the **Add Applications** dialog box, where you can add a group of App-V deployment types.
Edit	Click to open the **Edit Applications** dialog box, where you can edit an existing App-V deployment type group.
Delete	Click to delete an App-V deployment type group from the virtual environment.
Increase order **Decrease order**	If more than one group were listed, you could use the **Increase order** and **Decrease order** buttons to reorder the list. When multiple virtual applications modify the same file system or registry values on a client computer, the application with the highest order will take precedence.

Default Application Catalog Dialog Box

When you initially open AdminStudio, because a default Application Catalog has not yet been set, the **Default Application Catalog** dialog box opens, prompting you to open an Application Catalog.

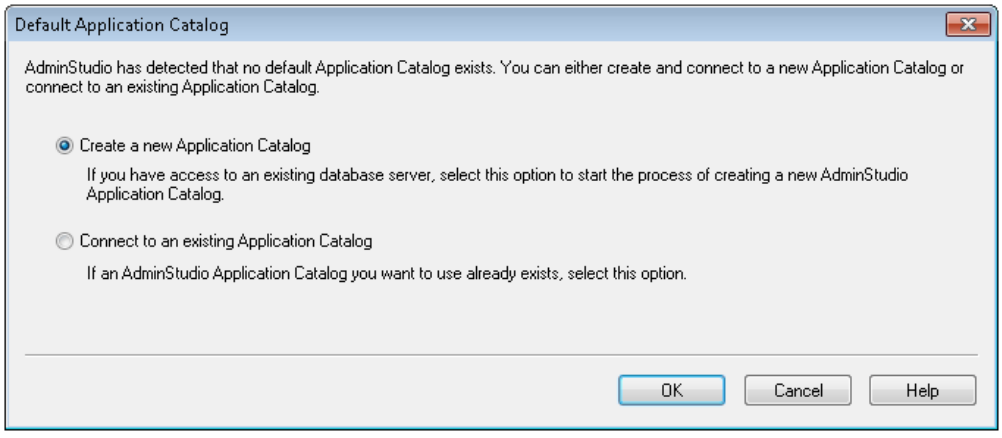


Figure 7-39: Default Application Catalog Dialog Box

You can select either of the following options:

Table 7-53 • Default Application Catalog Dialog Box Options

Option	Description
Create a new Application Catalog	Select this option to create a new, empty Application Catalog on an existing SQL Server database server that you have access to.
Connect to an existing Application Catalog	Select this option to connect to an existing Application Catalog on an SQL Server database server.

Edit Keywords Dialog Box

When App Portal performs a search on the **Browse Home** tab, it performs a search on not only the **Title**, **Brief Description**, and **Full Description** fields, but also on any **Keywords** that have been specified for that catalog item. On the **App Portal Information** tab of the **Application View**, you can specify keywords for an application's App Portal catalog item.

On the **Edit Keywords** dialog box, which is opened by clicking **Edit Keywords** on the **Keywords** dialog box, you can add new App Portal keywords and edit existing keywords.

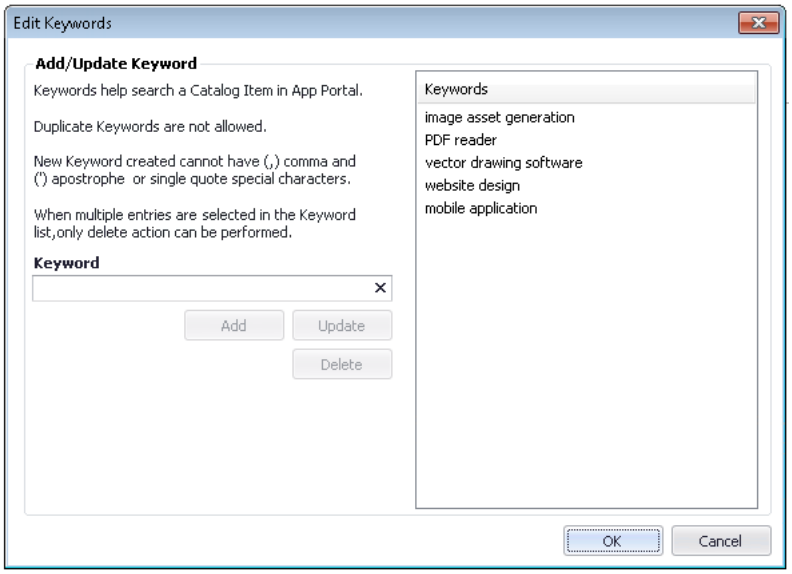


Figure 7-40: Edit Keywords Dialog Box

Note • *After you add a keyword on the **Edit Keywords** dialog box, it will be available to assign to any application in this Application Catalog.*

The **Edit Keywords** dialog box includes the following properties:

Table 7-54 • Edit Keywords Dialog Box

Property	Description
Keyword	Enter a keyword in this field and then click **Add** to add it to the **Keywords** list. ***Important*** • *Keywords must be single words only. If you enter a multiple-word keyword, all words of the phrase will be ignored when a search is performed.*
Keywords list	List of all defined keywords in this Application Catalog.
Add	Click to add the word in the **Keyword** field to the **Keywords** list.
Update	If you select an existing keyword in the **Keywords** list, it will be listed in the **Keyword** field. To change the keyword, edit it in the **Keywords** field and then click **Update**.
Delete	Click to delete the keyword selected in the **Keywords** list.

Extended Attribute Property Dialog Box

If you use extended attributes, and you click on a text extended attribute label, this dialog box opens.

Within it, you can provide the value for the extended attribute. When you click OK, the value is automatically displayed in the Extended Attributes View (Packages) next to the corresponding label.

Find Dialog Box

You can use the **Find** dialog box, which is accessed by clicking **Find** in the Application Manager ribbon, to search for data in Application Catalog tables.

Note • *This search is limited to string type columns.*

The tables that are searched depend upon what is selected when the **Find** dialog box is opened:

Table 7-55 • Search Options

If you select...	and specify these options...	this will be searched
Group	**All Tables and All Columns**	All tables and all columns in all of the Packages in the selected Group
Package	**All Tables and All Columns**	All tables and all columns in the selected Package
Package	**A Table and All Columns**	All columns of a specific table in the selected Package

Table 7-55 • Search Options

If you select...	and specify these options...	this will be searched
Package	**A Table and a Column**	A specific column in a specific table in the selected Package

Also, if you want to search for a partial match rather than an exact match, you can use the Partial Match option on the Find dialog box.

The Find dialog box can be accessed in several ways:

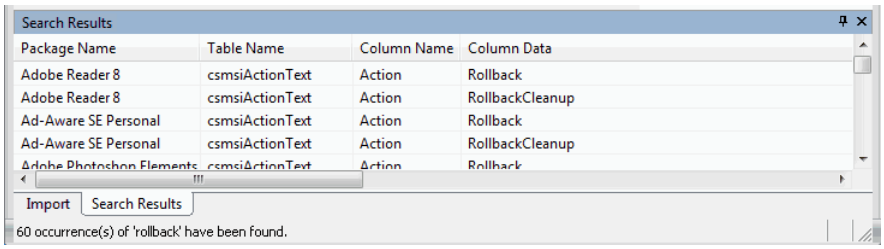
- Click **Find** in the Application Manager ribbon.
- Press **Ctrl** + **F**.
- Right-click on the node in the tree (Group, Application, MSI Package, App-V Package, etc.) that you want to search, and select **Find** on the shortcut menu.

The **Find** dialog box contains the following fields and buttons:

Table 7-56 • Find Dialog Box Properties

Properties	Description
Find What	Enter the text that you want to search for.
Look In Table	• If a package was selected when the **Find** dialog box was opened, all of the tables in that package are listed. Select the table that you would like to search, or select **<All Tables>**. When you select a table from this list, the **Look In Columns** list is populated with all of the columns in that table. • If a group is selected when you opened the **Find** dialog box, **<All Tables>** is the only option listed.
Look In Columns	• If you selected a table from the **Look in Table** list, all of the columns in that table are listed. Select the column that you would like to search, or select **<All Columns>**. • If a group was selected when you opened the **Find** dialog box, **<All Columns>** is the only option listed.
Partial Match	• If this option is not selected, Application Manager will search for an exact match of the text you entered in the **Find What** text box. The search will be case sensitive. • If this option is selected, then Application Manager will use appropriate wild card characters so that a partial data match is performed. The search will be case insensitive.

Table 7-56 • Find Dialog Box Properties

Properties	Description
Find Button	Click to initiate the search. The **Find** dialog box will close, and the data that is found is displayed in the **Search Results** tab of the Output Window, in the following format:  If you double click on this data, Application Manager will navigate to the appropriate record in the Tables View, and that record will be highlighted in red.

Flexera Identifier Dialog Box

Important • *Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

A Flexera Identifier is a unique code assigned to applications by FlexNet Manager Suite that is used to link application information from Application Manager with application information in App Portal and FlexNet Manager Suite.

If both Application Manager and FlexNet Manager Suite are connected to the same Flexera Service Gateway, each time you import an application into the Application Catalog, a search for the application's Flexera Identifier is performed, and if it is found, it is listed on the **General Information** tab of the **Application View**.

Sometimes, an application's Flexera Identifier is not found, such as when the value of the information in the application's **Product Name**, **Version**, **Edition**, or **Publisher** fields is either incorrect or too specific. If a Flexera Identifier is not found, you can use the **Flexera Identifier** dialog box to perform a search.

To open the **Flexera Identifier** dialog box, click the browse button in the **Flexera Identifier** field on the **General Information** tab of the **Application View**. You can also open it by clicking **Assign Flexera ID** on the **Application Search Results** dialog box (which is opened by clicking the **Unrecognized Applications** in the toolbar).

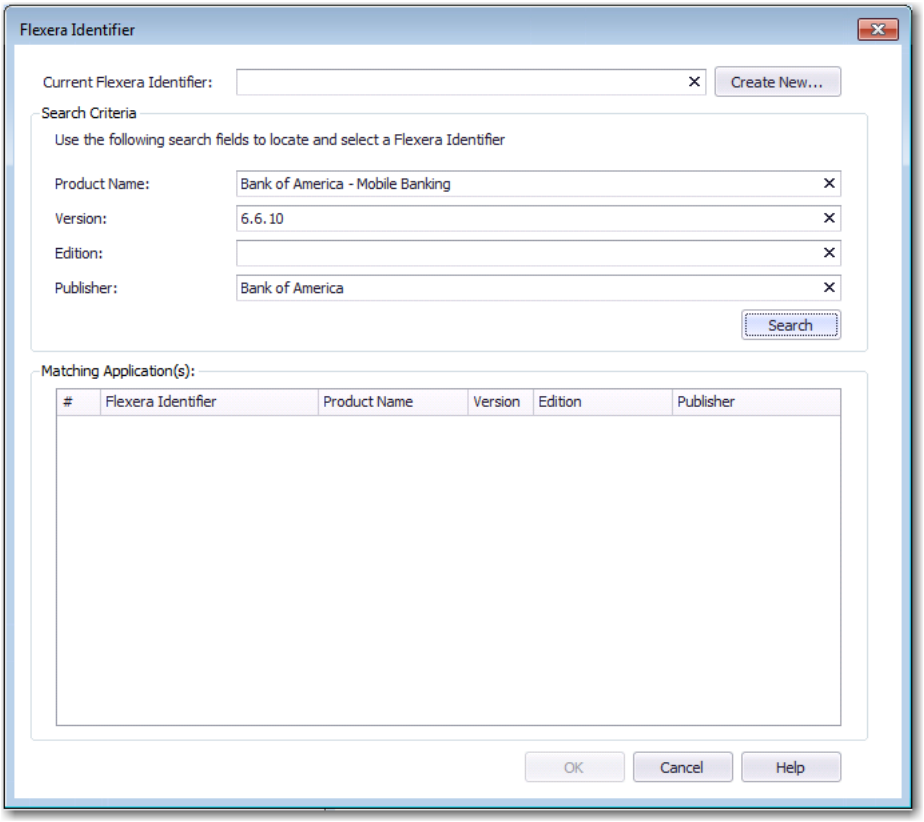


Figure 7-41: Flexera Identifier Dialog Box / Before Search

After a search has been performed, a list of possible matching applications is generated and displayed in the **Matching Application(s)** list.

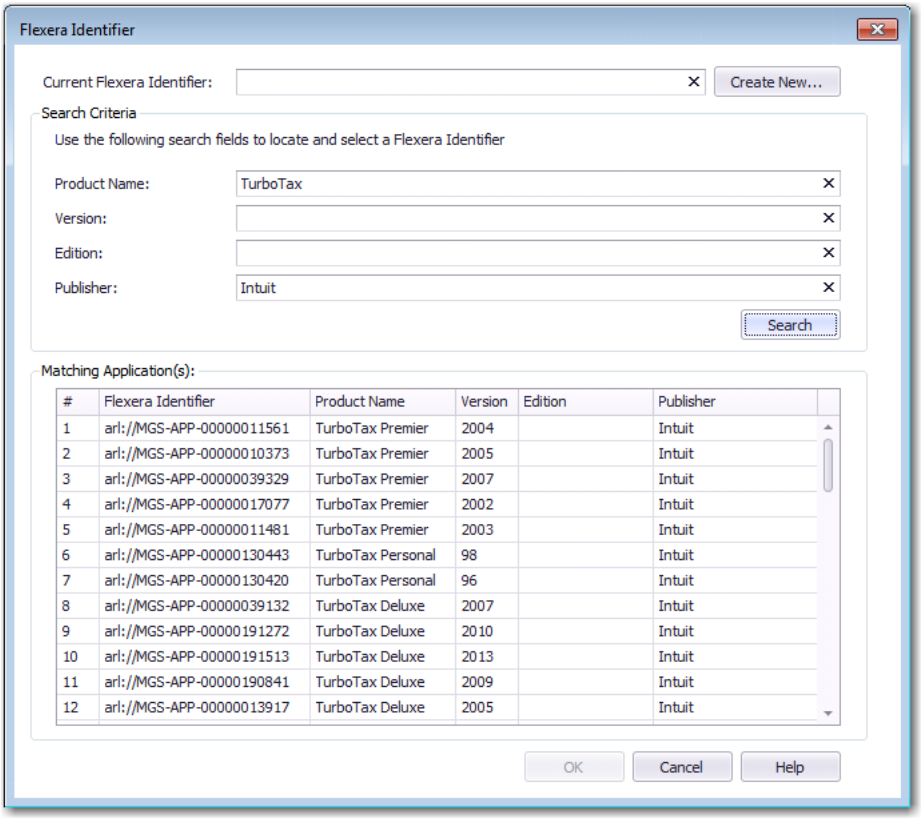


Figure 7-42: Flexera Identifier Dialog Box / After Search

The **Flexera Identifier** dialog box includes the following options:

Table 7-57 • Flexera Identifier Dialog Box

Option	Description
Current Flexera Identifier	If a Flexera Identifier has been found for this application, it will be listed in this field.
Create New	Click to open the Flexera Local Identifier Dialog Box where you can create a new, local Flexera Identifier. ***Important*** *• Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*
Search Criteria	Sometimes, if value of the information in the application's **Product Name**, **Version**, **Edition**, or **Publisher** fields is either incorrect or too specific, a Flexera Identifier will not be found. Therefore, edit the text in these fields to either correct the information or make it less specific, and then click **Search**.
Search	Click to initiate a search of the FlexNet Manager Suite database to generate a list of possible matching applications, based upon the entered criteria.

Table 7-57 • Flexera Identifier Dialog Box

Option	Description
Matching Application(s)	After a search is performed, possible matching applications are listed in this list. Select the matching application and click **OK** to select it.

Flexera Local Identifier Dialog Box

Important *• Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

When an application is imported into the Application Catalog, AdminStudio automatically queries the FlexNet Manager Suite ARL and attempts to obtain the application's Flexera Identifier. If an application still does not have an assigned Flexera Identifier, you can perform a manual search of the FlexNet Manager Suite Application Recognition Library, as described in Performing a Manual Search for a Flexera Identifier to attempt to identify an existing entry.

However, if you cannot locate an existing entry, you can create a new local Flexera Identifier entry for the FlexNet Manager Suite Application Recognition Library by clicking the **Create New** button on the **Flexera Identifier** dialog box to open the **Flexera Local Identifier** dialog box.

Important *• Creating a local Flexera Identifier requires FlexNet Manager Suite 2015 R2 SP3 or later.*

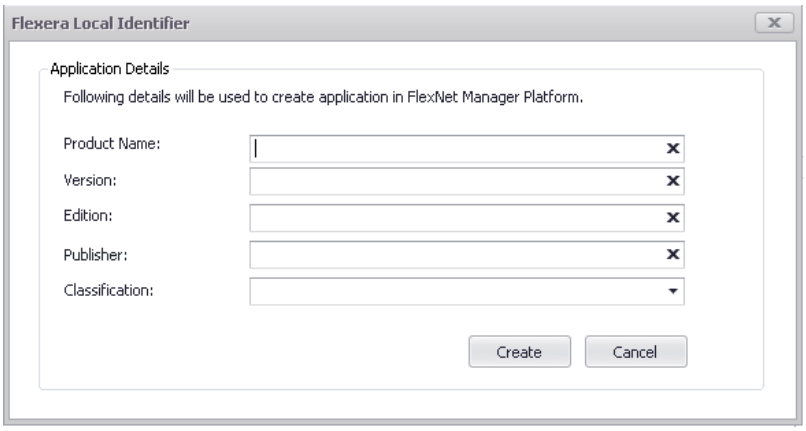


Figure 7-43: Flexera Local Identifier Dialog Box

The **Flexera Local Identifie**r dialog box includes the following properties:

Table 7-58 • Flexera Local Identifier Dialog Box

Property	Description
Product Name	The basic name of the application, excluding references to versions or editions, and without mentioning the publisher.
Version	The release number (or release identifier) of an application.

Table 7-58 • Flexera Local Identifier Dialog Box

Property	Description
Edition	Enter the edition of this application.
Publisher	The name of the publisher of this software, responsible for its development and distribution.
Classification	To indicate how this application is classified, select one of the following options: • **Beta**—A pre-release application (covers such items as beta releases, alpha releases, or release candidates) that you have under some special arrangement. • **Commercial**—The application requires a license to be purchased for use in a commercial setting. • **Freeware**—Licensed for use in a commercial environment free-of-charge. • **Malware**—A potentially harmful application (a virus, Trojan, and the like), and should be treated as malware. If installations of this application are identified, you need to address the corresponding incidents or security issues. • **Shareware**—The application is available for downloading from web sites, and typically uses a "try-before-you-buy" licensing model that might include reminder messages, functional limitations, or other restrictions until a full license is purchased. • **X Rated**—The application contains potentially objectionable or sexually explicit material. You might want to consider whether corporate policies require any action. • **Update**—The application represents an update, for example, a service pack, to another application, and is issued for free to all customers regardless of purchasing agreements or support contracts (a "minor" update).

When you click **Create**. A confirmation message appears stating that a new local Flexera Identifier has been created.

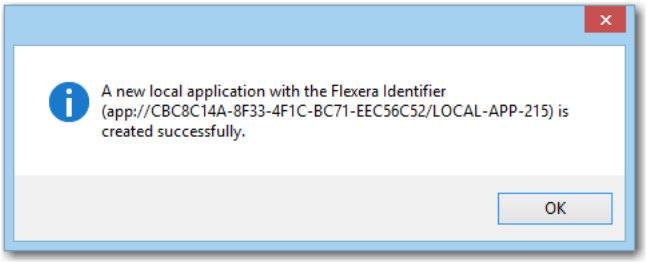


Figure 7-44: Confirmation of Creation of New Flexera Identifier

Global Conditions Dialog Box

In addition to using the **Requirements Wizard** to create global conditions, you can create new global conditions and edit existing global conditions on the **Global Conditions** dialog box, which can be opened by clicking the **Global Conditions** button on the **Home** tab of the Application Manager ribbon.

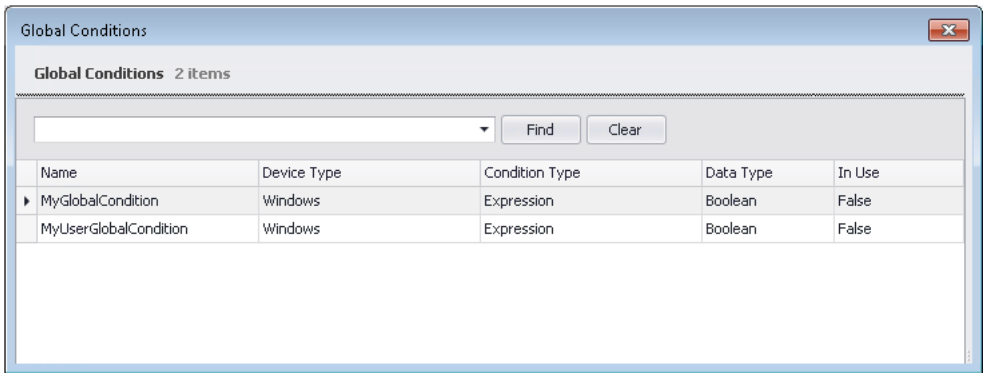


Figure 7-45: Global Conditions Dialog Box

The **Global Conditions** dialog box lists all of the global conditions present in the current Application Catalog. On the **Global Conditions** dialog box, you can edit or delete an existing global condition or create a new global condition:

- **Editing an existing global condition**—Right-click on the condition and then select **Edit Condition** from the shortcut menu. The **Create Global Condition** dialog box opens, where you can edit the condition.
- **Deleting an existing global condition**—Right-click on the condition and then select **Delete Condition** from the shortcut menu.
- **Adding a new global condition**—Right-click anywhere on the list of conditions and select **Create New Condition** from the shortcut menu. The **Create Global Condition** dialog box opens, where you can define a new condition.
- **View references**—If a condition is in use, right-click on the condition and select **References** from the shortcut menu to open the **References** dialog box, which lists the referring applications and the referring global conditions of the selected global condition.

The **Global Conditions** dialog box includes the following properties:

Table 7-59 • Global Conditions Dialog Box

Property	Description
Name	Name of global condition.
Device Type	Identifies the device type of the global condition.
Condition Type	Indicates whether the global condition is of the **Setting** or **Expression** type.
Data Type	Indicates the data type of the global condition, such as **Boolean** or **String**.
In Use	Indicates whether the condition is in use.
Search Box	Use this box along with the **Find** button to filter a large list of global conditions.

Keywords Dialog Box

When App Portal performs a search on the **Browse Home** tab, it performs a search on not only the **Title**, **Brief Description**, and **Full Description** fields, but also on any **Keywords** that have been specified for that catalog item. On the **App Portal Information** tab of the **Application View**, you can specify keywords for an application's App Portal catalog item.

On the **Keywords** dialog box, which is opened by clicking the browse button in the **Keywords** field on the **App Portal Information** tab of the **Application View**, you can add App Portal keywords to an application.

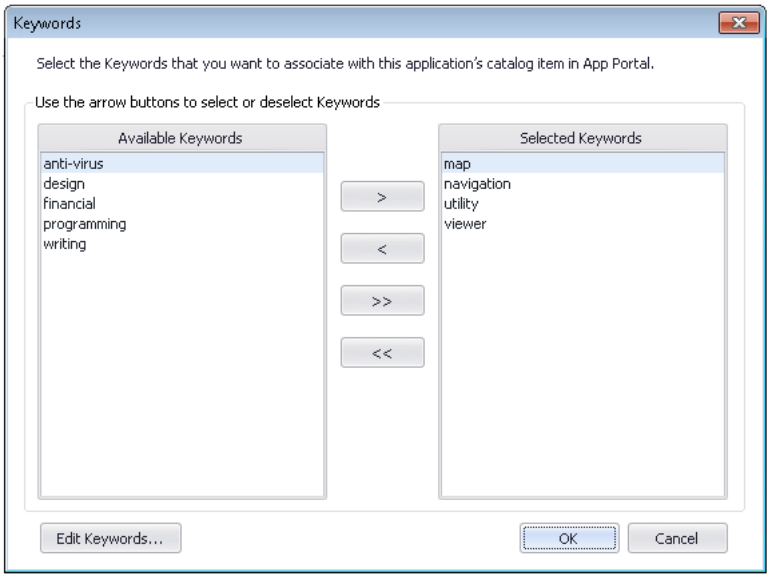


Figure 7-46: Keywords Dialog Box

The Keywords dialog box includes the following properties:

Table 7-60 • Keywords Dialog Box

Property	Description
Available Keywords List	Lists all of the existing keywords that are not currently assigned to this application.
Selected Keywords List	Lists all of the keywords that have been assigned to this application.
Arrow Keys	Click the **>** and **<** keys to move a selected keyword between the two lists. Click the **>>** and **<<** keys to move all of the keywords from one list to the other.
Edit Keywords	Click to open the Edit Keywords Dialog Box, where you can add additional keywords to this Application Catalog.

Login Required Dialog Box

If you chose to open an existing Application Catalog that was listed on the **Recent** tab of the Connect Application Catalog Dialog Box, and you are not currently logged in to that database, this dialog box opens prompting you to log in.

Properties Dialog Box

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

The **Properties** dialog box opens when you right-click on a group or application in the Application Manager tree and then, with the **Home** tab selected in the Application Manager ribbon, select **Properties** from the shortcut menu.

You can provide both a name and description for applications and groups, as well as any comments. The name entered on this dialog box appears in the Application Catalog tree to identify the application or group.

Table 7-61 • Group Properties Dialog Box Properties

Property	Description
Name	Provide a name for the group or application. This name will appear in the Application Manager tree. ***Note*** • *Name cannot exceed 70 characters in length.*
Description	Enter descriptive information about the group or application. ***Note*** • *The description will be displayed in the **Properties** area of the Group View when an application is selected in the Group View.* ***Note*** • *Description cannot exceed 126 characters in length.*
Comments	Provide comments about the group or application. ***Note*** • *Comments cannot exceed 253 characters in length.*

Options Dialog Box

On the **Options** dialog box, which you open by selecting **Options** from the **Application Catalog** tab, you can specify options relating to package import, package testing, conflict identification, server connection information, connection to the Flexera Service Gateway, as well as some general settings. This dialog box is divided into the following tabs:

- General Tab
- Import Options / General Tab
- Import Options / Duplicate Package Tab
- Import Options / Application Model Defaults Tab
- Import Options / Software Tagging Tab

- Analyze Tab
- Windows Installer Validation Tab
- ACE Tests Tab
- Mobile Tests Tab
- Plugin Options Tab
- Package Feed Options Tab
- Wrap Options Tab
- Server Options / Distribution System Tab
- Server Options / Microsoft ACT Tab
- Package Automation Options / Monitored Directory
- Package Automation Options / Configure Actions
- Package Automation Options / Schedule Automation
- Package Automation Options / Notifications
- Software Repository Tab
- Flexera Integration / Flexera Service Gateway (FSG) Tab
- AdminStudio Services via FSG Tab

General Tab

On the **General** tab of the **Options** dialog box, you can configure the following options:

Table 7-62 • General Tab Properties

Property	Description
Confirm All Drag-Drop Operations	Select this option if you want Application Manager to prompt you for confirmation whenever you drag and drop items.
Display Broken MSI/MST Package Links	When selected, any broken package links will be indicated by an icon change in the Application Manager tree and with a message in the Home Deployment Type View, which allows you to attempt to locate the package.
Only Display View Nodes With Data	When this option is selected, packages containing views without data will not display those views. For example, if a package has no shortcuts, then the **Shortcuts** view is not displayed for that package.

Table 7-62 • General Tab Properties (cont.)

Property	Description
Select Temporary Location	Packages are temporarily copied to this location to extract all of the data needed during import and testing. In other words, it is a "working directory" for AdminStudio. After package extraction, the files are deleted. In some cases, certain locations on a machine might be locked down and AdminStudio would be unable to access them, which would cause package extraction to fail. In this field, specify a location that AdminStudio has access to.
Extended Attribute Description File	Specify the name and location of the extended attribute description file (`.xml`) which specifies the extended attributes available for each package in the Application Catalog. AdminStudio includes a default XML file for extended attributes, which is stored in the AdminStudio shared location. You can also construct your own Package Extended Attribute Description File. Each new Application Catalog automatically points to this file, and displays the name and location of the file in this tab. ***Tip*** *• If you overwrite the default XML file with your extended attributes data, all subsequent Application Catalogs created include your attributes by default.*
Integrate with Workflow Manager	Select this option to integrate extended attributes with Workflow Manager. When this option is selected, you can associate extended attribute data for packages in Application Manager with workflow requests in Workflow Manager. This is accomplished by right-clicking on the package name in the Application Manager tree and selecting **Associate with Workflow Manager Workflow Request** from the shortcut menu.

Import Options / General Tab

On the **Import Options / General** tab of the **Options** dialog box, you can configure several options that affect how packages are imported into the Application Catalog.

You can configure the following options:

Table 7-63 • Import Options / General Tab Properties

Property	Description
Automatically Execute Tests After Import	If this option is selected, Application Manager will automatically test packages against all selected compatibility, best practices, and risk assessment tests as part of the import process. All of the tests selected on the **Select Tests to Execute** dialog box (other than those in the **Application Conflicts** category) will be run. By default, this option is not selected. While having this option selected will mean longer import times for each application, packages will have all testing details populated immediately after import. However, if you are concerned with the length of import time, you may not want to select this option. ***Note*** • *If this option is not selected, no tests will be performed on a package immediately following import. To manually run the tests, select the package (or select the group that contains the package) and then click the **Execute Tests** button in the ribbon on the **Analyze** tab.* ***Note*** • *In AdminStudio 2016, the **Automatically Execute Tests After Import** option was selected by default; in AdminStudio 2016 SP1, this option is no longer selected by default. Therefore, for users upgrading from AdminStudio 2016 to 2016 SP1, you will inherit the new default selection, which means that, by default, this option will not be selected and no testing will be performed immediately after import. To have the same user experience as you did in AdminStudio 2016, you need to select this option. In versions of AdminStudio prior to 2016, this option was not selected by default.*
Integrate InstallScript Headers into Application Model Data	Extract the data in the imported package's InstallScript header files. The InstallScript header data is used to help populate the basic application and package metadata (such as product code, product name, etc.).
Ignore Tables list	The **Ignore Tables** list displays all of the tables that will be ignored during import (not imported into the Application Catalog). You can select tables and delete them from the list, or you can add new tables to the list by clicking **Add**, which opens the **Add Ignore Table** dialog box.

Import Options / Duplicate Package Tab

When you import a package into the Application Catalog, Application Manager checks specific identifiers that are selected on the **Duplicate Package** tab to determine if that package has already been imported.

If Application Manager determines that you are attempting to import a duplicate package (based upon the selected identifiers), the package is renamed using the specified **Duplicate Package Naming Syntax**.

The identifiers you can select on the **Duplicate Package** tab are as follows:

Table 7-64 • Import Options / Duplicate Package Tab Properties

Property	Description
Duplicate Package Identification Options	Select one or more of the following options to specify the identifiers that Application Manager will check to determine if a Windows Installer package has already been imported: • **Package Code property**—Identifier of package product was installed from. No two non-identical `.msi` files should ever have the same package code. • **Product Code property**—Unique identifier for the particular product release, represented as a string GUID, for example {12345678-1234-1234-1234-123456789012}. • **Product Language property**—The language the installer should use for any strings in the user interface that are not authored into the database. • **Product Version property**—Version of the product in string format. The format of the string is: `major.minor.build`. • **List of Transform Files**—A list of the transformations associated with this package import operation. • **[None Selected]**—If you do not select any of these five identifiers, Application Manager checks the ProductName Property identifier to determine if a package is a duplicate.
Duplicate Virtual Package Identification Options	Select one or more of the following options to specify the identifiers that Application Manager will check to determine if an App-V package has already been imported: • **PackageGUID property**—Unique identifier of App-V package. • **VersionGUID property**—Unique identifier of App-V package version. • **[None Selected]**—If you do not select either of these identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.

Duplicate Package Naming Syntax	When it identifies a duplicate package, Application Manager generates a new name for that package using the syntax specified in this field. The default syntax is:
	`[Manufacturer]_[ProductName]`
	This means that if Application Manager encountered a duplicate package, it would pre-pend the duplicate's Product Name with the Manufacturer's name and, if necessary, append the Product Name with numbers. For example, the second time that PowerPoint is imported, its name would be changed to:
	`Microsoft Corporation_PowerPoint`
	To change the default naming syntax, edit this field.
	This generated name will be displayed in the Application Manager tree view.
	
	Note • *Changing this "display" name does not change the ProductName Property that appears in the title bar of the Home Deployment Type View.*

Example of Importing a "Duplicate" Package

One common reason why you might want to import the same package into your Application Catalog database more than once would be if you wanted to use InstallShield Editor to create custom installation SKUs of a common MSI package to distribute to different departments in your organization, each installation including certain features that are appropriate for the department and excluding certain features that are not appropriate. For example, if you were distributing a copy of Microsoft Office, you could add transforms to the Microsoft Office MSI package so that:

- Accounting's installation would include only Word and Excel
- Marketing's installation would include only Word and PowerPoint, and
- Development's installation would include only Word and Access.

Therefore, you might want to import the same package into your database more than once, each time with a different set of transformations. What happens when you import the package the second time depends upon the identifiers you selected on the **Duplicate Package** tab. In this example:

- If you select the **List of Transform Files** and **ProductCode** identifiers on the **Import Options / Duplicate Package** tab of the **Options** dialog box, Application Manager will not identify these two packages as duplicate, even though they have the same ProductCode, because they have a different set of transformations. Therefore, the package will be imported with the same display name as the first package.
- If you only select the **ProductCode** identifier on the **Import Options / Duplicate Package** tab of the Application Manager Options dialog box, Application Manager will identify the second package as a duplicate because the two packages have the same ProductCode, and the package will be renamed.

Note • *The options that you select on the **Import Options / Duplicate Package** tab of the Application Manager Options dialog box apply globally to all packages that you attempt to import; you cannot apply different identifiers to different packages. Also, since these options are saved in the AdminStudio Shared Directory, everyone using AdminStudio at your organization who shares the same directory will share the same Duplicate Package options.*

Import Options / Application Model Defaults Tab

On the **Import Options / Application Model Defaults** tab of the **Options** dialog box, you can specify the default values for Microsoft System Center 2012 Configuration Manager application model properties. These default values will be assigned to new applications being imported into the Application Catalog, if they do not already have a value specified.

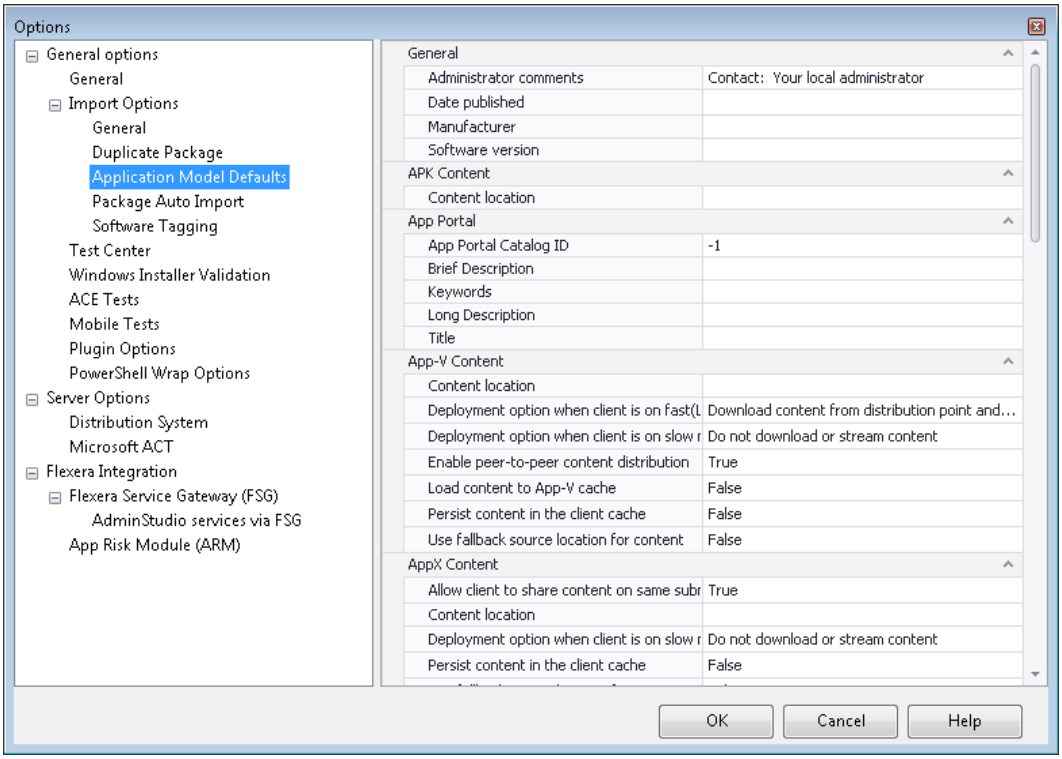


Figure 7-47: Options Dialog Box / Application Model Defaults Tab

The **Application Model Defaults** tab of the **Options** dialog box includes the following properties:

Table 7-65 • Import Options / Application Model Defaults Tab Properties

Category	Option	Description
General	**Administrator comments**	Description of the application.
	Date published	The purpose of this field is to display the date the application was published to System Center 2012 Configuration Manager. When you create an application in Application Manager (usually by importing a package), this field is left blank. • If you do not enter a value in this field, when you publish the application to System Center 2012 Configuration Manager, this field will be automatically updated to display the published date. • If you enter a value in this field, and then publish the application to System Center 2012 Configuration Manager, the date that you entered will be listed as the published date in Configuration Manager.
	Manufacturer	Manufacturer of the application, as discovered from its deployment types.
	Software version	Version of the application, as discovered from its deployment types.
APK Content	**Content location**	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
App Portal	**App Portal Catalog ID**	Number identifying Application Catalog item in App Portal.
	Brief Description	Default entry for App Portal Brief Description property.
	Keywords	Default App Portal keywords.
	Long Description	Default entry for App Portal Long Description property.
	Title	Default entry for App Portal Title property.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
App-V Content	**Content location**	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
	Deployment option when client is on fast (LAN) network	Select one of the following options to specify how the client should download content when on a fast network: • **Download content from distribution point and run locally**—Select this option to download the content from the distribution point and run it locally. • **Stream content from distribution point**—Select this option for App-V packages to stream content from the distribution point.
	Deployment option when client is on slow network	Select one of the following options to specify whether the client should download content when on a slow network: • **Do not download content**—When the client is connected within a slow or unreliable network boundary, do not download content. Select this option to save network bandwidth. (Default) • **Download content from distribution point and run locally**—Select this option if, when the client is connected within a slow or unreliable network boundary, you want it to download the content from the distribution point and run it locally.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
App-V Content (continued)	**Enable peer-to-peer content distribution**	Select this option to reduce load on the network by allowing clients to download content from other clients on the network that have already downloaded and cached the content. This option utilizes Windows BrancheCache and can be used on computers running Windows Vista SP2 and later.
	Load content to App-V cache	Entire package (instead of just Feature Block 1) is loaded completely into the App-V cache prior to launch.
	Persist content in the client cache	To retain content in the cache on the client computer indefinitely even if it has already been run, select **True**. ***Note*** *• Setting this property to* ***True*** *will reduce the available cache space. This might cause a large deployment to fail at a later point if there is insufficient space available in the cache.*
	Use fallback source location for content	To enable clients to "fall back" to using an unprotected distribution point if the package is not available on a protected (preferred) distribution point, set this option to **True**. By default, this option is set to **False**.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
AppX Content	**Allow client to share content on same subnet**	To reduce the load on the network by allowing clients to download content from other local clients on the network that have already downloaded and cached the content, select **True**. ***Note*** *• Applies to Windows Installer and Legacy Installer (.exe) packages only.*
	Content location	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
	Deployment option when client is on slow network	Select one of the following options to specify whether the client should download content when on a slow network: • **Do not download content**—When the client is connected within a slow or unreliable network boundary, do not download content. Select this option to save network bandwidth. (Default) • **Download content from distribution point and run locally**—Select this option if, when the client is connected within a slow or unreliable network boundary, you want it to download the content from the distribution point and run it locally.
	Persist content in the client cache	To retain content in the cache on the client computer indefinitely even if it has already been run, select **True**. ***Note*** *• Setting this property to* ***True*** *will reduce the available cache space. This might cause a large deployment to fail at a later point if there is insufficient space available in the cache.*
	Use fallback source location for content	To enable clients to "fall back" to using an unprotected distribution point if the package is not available on a protected (preferred) distribution point, set this option to **True**. By default, this option is set to **False**.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
AppX Link User Experience	**Maximum allowed run time (minutes)**	Specifies the maximum time (in minutes) that the program is expected to run on the client computer. This setting can be specified as a whole number greater than zero. The default setting is 120 minutes. This value is used for two purposes: • To monitor results from the deployment type. • To determine if a deployment type will be installed when maintenance windows have been defined on client devices.
AppX User Experience	**Maximum allowed run time (minutes)**	Specifies the maximum time (in minutes) that the program is expected to run on the client computer. This setting can be specified as a whole number greater than zero. The default setting is 120 minutes. This value is used for two purposes: • To monitor results from the deployment type. • To determine if a deployment type will be installed when maintenance windows have been defined on client devices.
Catalog	**Classification**	Identifies whether this is a **Client** or **Server** application, or whether the application classification is **Unknown**. By default, this property is set to **Client** for all applications.
	Localized description	Localized version of application description.
	Localized display name	Localized version of the application's display name.
	User documentation	Location of documentation provided with this application.
IPA Content	**Content location**	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
Misc	**Display supersedes information to user**	Set this option to **True** to allow users to see deployments for this application and all applications that it supersedes in the Application Catalog. This may result in the user installing multiple applications on the same device, if the requirements for these applications are met.
	Distribute to preferred DP	To enable on-demand content distribution to preferred distribution points, select **True**. When enabled, the content is distributed to all preferred distribution points in the list when a client requests the content for the package and the content is not available on any preferred distribution points.
	Distribution point groups	Specify the default System Center 2012 Configuration Manager distribution point groups to which application content will be distributed. ***Note*** *• If AdminStudio is integrated with App Portal, this is a required field. If no distribution point group is entered, the App Portal administrator will be required to manually enter this information in ConfigMgr (Formerly called as System Center Configuration Manager) before App Portal will be able to distribute applications.*
	Distribution priority	When you are sending multiple packages to a distribution point, those packages are sent in priority order, with higher priority packages being sent first. Use this property to specify a package's priority. The following options are available: • **High** • **Medium** • **Low**
	Install from Install Application task sequence	Select **True** to deploy this application when deploying an operating system, as part of an Install Application task sequence. Select **False** to install this application manually.
	Prestaged DP settings	Select one of the following options to specify how you want to distribute content to prestaged distribution points: • **Automatically download content when packages are assigned to DP**—Select to ignore the prestage settings and distribute content to the distribution point. • **Download only content changes to the DP**—Select to prestage the initial content to the distribution point, and then distribute content changes to the distribution point. • **Manually copy the content in this package to the DP**—Select to always prestage content on the distribution point. (Default)

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
MSI Content	**Allow client to share content on same subnet**	To reduce the load on the network by allowing clients to download content from other local clients on the network that have already downloaded and cached the content, select **True**. ***Note*** *• Applies to Windows Installer and Legacy Installer (.exe) packages only.*
	Content location	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
	Deployment option when client is on slow network	Select one of the following options to specify whether the client should download content when on a slow network: • **Do not download content**—When the client is connected within a slow or unreliable network boundary, do not download content. Select this option to save network bandwidth. (Default) • **Download content from distribution point and run locally**—Select this option if, when the client is connected within a slow or unreliable network boundary, you want it to download the content from the distribution point and run it locally.
	Persist content in the client cache	To retain content in the cache on the client computer indefinitely even if it has already been run, select **True**. ***Note*** *• Setting this property to* ***True*** *will reduce the available cache space. This might cause a large deployment to fail at a later point if there is insufficient space available in the cache.*
	Use fallback source location for content	To enable clients to "fall back" to using an unprotected distribution point if the package is not available on a protected (preferred) distribution point, set this option to **True**. By default, this option is set to **False**.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
MSI Installer	**Install command line**	Specify the command line that Configuration Manager will use to install this package on a client machine, including any required installation parameters.
	Install folder	Specify the folder that contains the installation program for the deployment type. This folder can be an absolute path on the client or a path to the distribution point folder that contains the installation files. This field is optional.
	Installation source management product code	To enable installation source management, enter the Windows Installer product code. ***Note*** **•** *In ConfigMgr (Formerly called as System Center Configuration Manager), installation source management enables a Windows Installer file to automatically be updated or repaired from content source files on an available distribution point.*
	Run installation as 32-bit process on 64-bit client	Select **True** to run the installation of this deployment type as a 32-bit process on a 64-bit client. To run it as a 64-bit process on a 64-bit client, select **False**.
	Uninstall command line	Specify the command line that Configuration Manager will use to uninstall this package from a client machine, including any required parameters.
	Uninstall folder	Specify the folder that contains the uninstall program for the deployment type. This folder can be an absolute path on the client or a path relative to the distribution point folder that contains the package. This field is optional.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
MSI User Experience	**Allow user to view and interact with the program installation**	Set this property to **True** to enable the user to view and interact with the program installation in order to configure installation options. If it is set to **False**, the program installation is hidden from the user. ***Note*** *• This property can be set to **True** only when the **Login requirement** property is set to **Only when a user is logged on**.*
	Enforce specific behavior	Select one of the following options to enable Configuration Manager to enforce specific OS reboot behavior regardless of the application's intended behavior: • **Determine behavior based on return codes**—Handle reboots based on the codes configured on the Return Codes tab. • **No specific action**—No reboot required after installation. • **The software installation program might force a device restart**—Configuration Manager will not control reboot; the actual installation might force a reboot without warning. • **Configuration Manager client will force a mandatory device restart**—Configuration Manager will force a device reboot—either by notifying the user or without notification.
	Estimated installation time (min)	Specify the estimated time that the deployment type will take to install.
	Installation behavior	Select one of the following options: • **User**—The application installs for only the user who it is deployed to. • **System**—The application installs only once and is available to all users. • **Any**—If the application is deployed to a device, then it will install for all users. If the application is deployed to a user, then it will install for only that user.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
MSI User Experience (Continued)	**Installation program visibility**	Select one of the following options to specify the mode in which the deployment type will run on client devices: • **Maximized**—The deployment type runs maximized on client devices. Users will see all installation activity. • **Normal**—The deployment type runs in the normal mode based on system and program defaults. This is the default mode. • **Minimized**—The deployment type runs minimized on client devices. Users might see installation activity in the notification area or task bar. • **Hidden**—The deployment type runs hidden on client devices and users will see no installation activity.
	Logon requirement	Select one of the following options to specify the login requirements for installing this application: • **Only when a user is logged on** • **Whether or not a user is logged on** • **Only when no user is logged on** ***Note*** • *If you have set the **Installation behavior** property to **User**, this option will default to **Only when a user is logged on** and cannot be changed.*
	Maximum allowed run time (min)	Specifies the maximum time (in minutes) that the program is expected to run on the client computer. This setting can be specified as a whole number greater than zero. The default setting is 120 minutes. This value is used for two purposes: • To monitor results from the deployment type. • To determine if a deployment type will be installed when maintenance windows have been defined on client devices.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
Script Content	**Allow client to share content on same subnet**	To reduce the load on the network by allowing clients to download content from other local clients on the network that have already downloaded and cached the content, select **True**. ***Note*** *• Applies to Windows Installer and Legacy Installer (.exe) packages only.*
	Content location	In ConfigMgr (Formerly called as System Center Configuration Manager), the **Content location** is the location where a deployment type's files are located. In Application Manager, this field usually remains blank. However, if you enter an application-specific location for publishing in this field, Distribution Wizard will not create a GUID folder and will, instead, publish the application from this location **only if** the source files already exist in this location. Otherwise, the source files are copied to the location specified in the **Location to Publish Packages** field on the **Server Options > Distribution System** tab of the **Options** dialog box, and published from there.
	Deployment option when client is on slow network	Select one of the following options to specify whether the client should download content when on a slow network: • **Do not download content**—When the client is connected within a slow or unreliable network boundary, do not download content. Select this option to save network bandwidth. (Default) • **Download content from distribution point and run locally**—Select this option if, when the client is connected within a slow or unreliable network boundary, you want it to download the content from the distribution point and run it locally.
	Persist content in the client cache	To retain content in the cache on the client computer indefinitely even if it has already been run, select **True**. ***Note*** *• Setting this property to* ***True*** *will reduce the available cache space. This might cause a large deployment to fail at a later point if there is insufficient space available in the cache.*
	Use fallback source location for content	To enable clients to "fall back" to using an unprotected distribution point if the package is not available on a protected (preferred) distribution point, set this option to **True**. By default, this option is set to **False**.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
Script Installer	**Install Command Line**	Specify the command line that Configuration Manager will use to install this package on a client machine, including any required installation parameters.
	Install Folder	Specify the folder that contains the installation program for the deployment type. This folder can be an absolute path on the client or a path to the distribution point folder that contains the installation files. This field is optional.
	Installation source management product code	To enable installation source management, enter the Windows Installer product code. ***Note*** **•** *In ConfigMgr (Formerly called as System Center Configuration Manager), installation source management enables a Windows Installer file to automatically be updated or repaired from content source files on an available distribution point.*
	Run installation as 32-bit process on 64-bit client	Select **True** to run the installation of this deployment type as a 32-bit process on a 64-bit client. To run it as a 64-bit process on a 64-bit client, select **False**.
	Uninstall Command Line	Specify the command line that Configuration Manager will use to uninstall this package from a client machine, including any required parameters.
	Uninstall Folder	Specify the folder that contains the uninstall program for the deployment type. This folder can be an absolute path on the client or a path relative to the distribution point folder that contains the package. This field is optional.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
Script User Experience	**Allow user to view and interact with the program installation**	Set this property to **True** to enable the user to view and interact with the program installation in order to configure installation options. If it is set to **False**, the program installation is hidden from the user. ***Note*** *• This property can be set to* ***True*** *only when the* ***Login requirement*** *property is set to* ***Only when a user is logged on****.*
	Enforce specific behavior	Select one of the following options to enable Configuration Manager to enforce specific OS reboot behavior regardless of the application's intended behavior: • **Determine behavior based on return codes**—Handle reboots based on the codes configured on the Return Codes tab. • **No specific action**—No reboot required after installation. • **The software installation program might force a device restart**—Configuration Manager will not control reboot; the actual installation might force a reboot without warning. • **Configuration Manager client will force a mandatory device restart**—Configuration Manager will force a device reboot—either by notifying the user or without notification.
	Estimated installation time (min)	Specify the estimated time that the deployment type will take to install.
	Installation behavior	Select one of the following options: • **User**—The application installs for only the user who it is deployed to. • **System**—The application installs only once and is available to all users. • **Any**—If the application is deployed to a device, then it will install for all users. If the application is deployed to a user, then it will install for only that user.

Table 7-65 • Import Options / Application Model Defaults Tab Properties (cont.)

Category	Option	Description
Script User Experience (Continued)	**Installation program visibility**	Select one of the following options to specify the mode in which the deployment type will run on client devices: • **Maximized**—The deployment type runs maximized on client devices. Users will see all installation activity. • **Normal**—The deployment type runs in the normal mode based on system and program defaults. This is the default mode. • **Minimized**—The deployment type runs minimized on client devices. Users might see installation activity in the notification area or task bar. • **Hidden**—The deployment type runs hidden on client devices and users will see no installation activity.
	Login requirement	Select one of the following options to specify the login requirements for installing this application: • **Only when a user is logged on** • **Whether or not a user is logged on** • **Only when no user is logged on** ***Note*** *• If you have set the* ***Installation behavior*** *property to* ***User****, this option will default to* ***Only when a user is logged on*** *and cannot be changed.*
	Maximum allowed run time (min)	Specifies the maximum time (in minutes) that the program is expected to run on the client computer. This setting can be specified as a whole number greater than zero. The default setting is 120 minutes. This value is used for two purposes: • To monitor results from the deployment type. • To determine if a deployment type will be installed when maintenance windows have been defined on client devices.

Import Options / Software Tagging Tab

On the **Software Tagging** subtab of the **Import Options > Software Tagging** tab of the Application Manager Options dialog box, you can enable or disable automatic software tag file creation and can set the default values for **Tag Creator Name** and **Tag Creator RegID**.

Important • *Any changes that you make to the software tagging options on the* ***Software Tagging*** *tab of the Application Manager Options dialog box will also automatically be made to the options on the* ***Build Options*** *tab of the Repackager* ***Options*** *dialog box.*

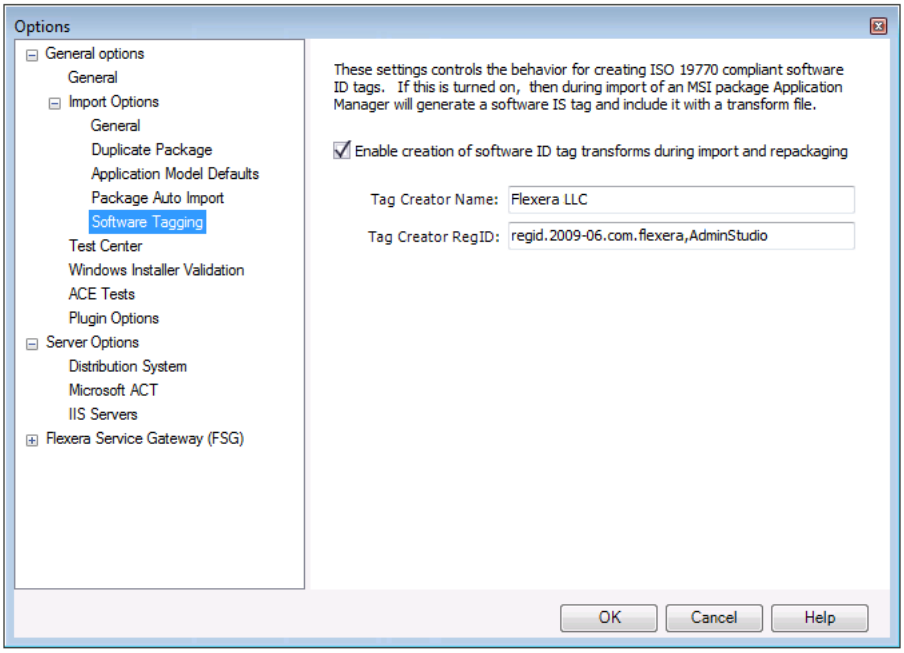


Figure 7-48: Options Dialog Box / Software Tagging Tab

On the **Software Tagging** tab of the **Options** dialog box, you can configure the following properties:

Table 7-66 • Import Options / Software Tagging Tab Properties

Option	Description
Enable creation of software ID tag transforms during import and repackaging	Select to instruct AdminStudio to automatically create a transform file containing software tag file(s) for Windows Installer packages that are imported into the Application Catalog or built using Repackager. By default, this option is selected. ***Note*** • *Whenever a Windows Installer package is imported into the Application Catalog or built using Repackager, AdminStudio creates a software ID tag file (which is stored in the Application Catalog), but if the* ***Enable creation of software ID tag transforms during import and repackaging*** *option it not selected, AdminStudio does not create the transform.*
Tag Creator Name	Enter a name to identify the creator of the software ID tag files that will be created by AdminStudio. By default, the value is `Flexera`.

Table 7-66 • Import Options / Software Tagging Tab Properties (cont.)

Option	Description

Tag Creator RegID	Enter an ID to uniquely identify the creator of the software ID tag files that will be created by AdminStudio, using the following format:
	`regid.YYYY-MM.ReversedDomainName,optional_division`
	For example:
	`regid.2009-06.com.yourcompany,GlobalProductDivision`
	By default, the value is AdminStudio's RegID:
	`regid.2009-06.com.flexera,AdminStudio`

Analyze Tab

On the **Analyze** tab of the **Options** dialog box, you can configure several options that affect how packages are tested and how issues are resolved.

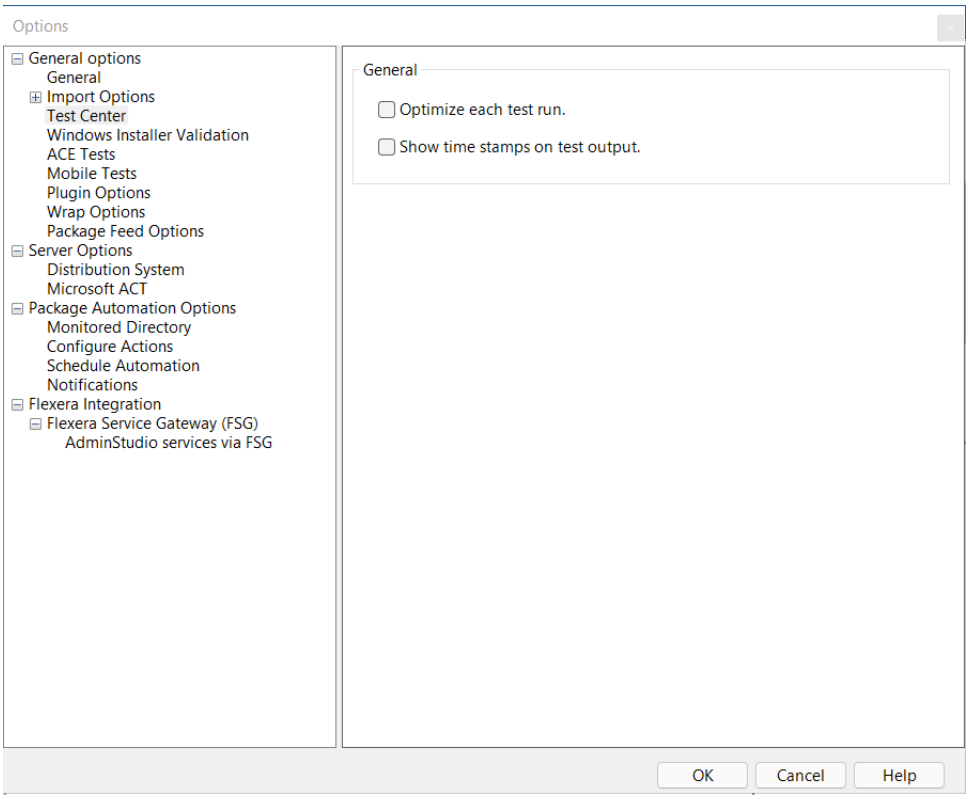


Figure 7-49: Options Dialog Box / Analyze Tab

On the **Analyze** tab of the **Options** dialog box, you can configure the following properties:

Table 7-67 • Analyze Tab Properties

Option	Description
Optimize each test run	This option determines whether Application Manager executes all tests or only tests that were not previously run: • **Selected**—If this option is selected, before beginning testing, Application Manager will check the selected packages to see which tests have been run on each package and which have not been run. It also checks to see if the Windows Installer file (or its transform file) or App-V package has changed since the last time that testing was performed. If the packages have not changed, Application Manager will then only execute those selected tests which have not yet been run. • **Not selected**—If this option is not selected, Application Manager will execute all selected tests on all selected packages each time testing is initiated, even if the test has already been run on a package and neither the package nor its transform file has changed. If you have a large number of applications in your Application Catalog, selecting this option enables you to start testing, then click **Stop** to pause testing when you want to access Application Catalog to perform other tasks. When you click **Stop**, Application Catalog would finish executing the current test. When you were ready to resume testing, you could then click **Execute Tests** and Application Catalog would immediately begin testing where it left off the last time testing was performed.

Show time stamps on test output	Select this option to include date and time indications in the test output, such as:
	`06-21-2012 17:16:54 :: Determining Microsoft ICE tests to run.`
	and
	`06-21-2012 17:17:06 :: Microsoft ICE - ICE 58`

Maximum number of links to crawl	Specify the number of hyperlinked levels deep that will be tested when performing static (non-interactive) testing.

Windows Installer Validation Tab

On the **Windows Installer Validation** tab of the **Options** dialog box, you specify the files containing the ICE rules that will be used in validation testing.

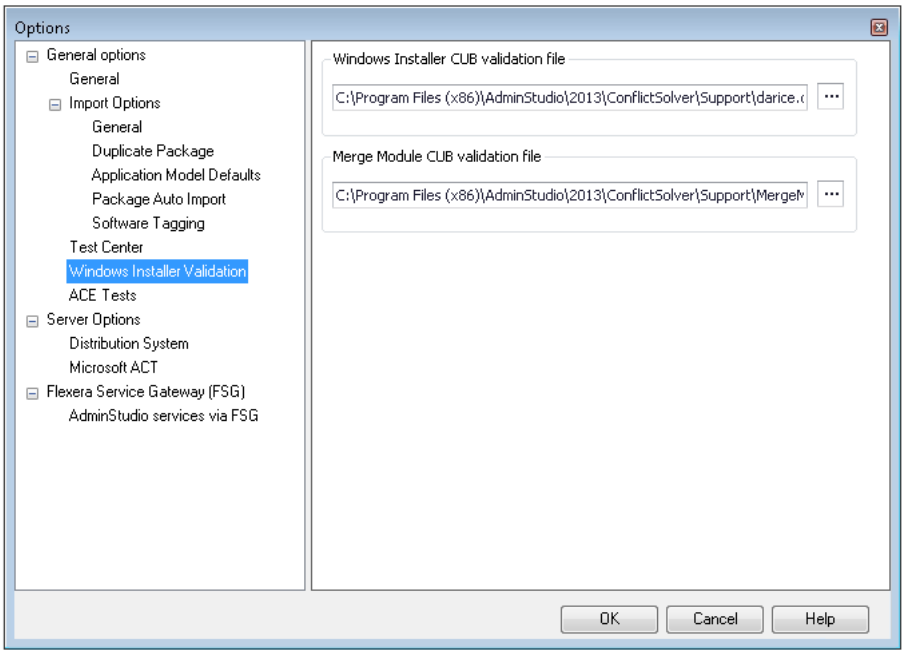


Figure 7-50: Options Dialog Box / Windows Installer Validation Tab

On the **Windows Installer Validation** tab of the **Options** dialog box, you can configure the following properties:

Table 7-68 • Windows Installer Validation Tab Properties

Property	Description
Windows Installer CUB validation file	The file specified in this field contains the Internal Consistency Evaluators (ICEs) that will be used for validation of Windows Installer packages. Either enter the location of this file directly, or use the Browse button (...) to locate it.
Merge Module CUB validation file	The file specified in this field contains the Internal Consistency Evaluators (ICEs) that will be used for validation of merge modules. Either enter the location of this file directly, or use the Browse button (...) to locate it.

ACE Tests Tab

On the **ACE Tests** tab of the **Options** dialog box, you can access the **Rules Viewer**, which can be used to add user defined ACEs, and you can specify a custom ACE rule file.

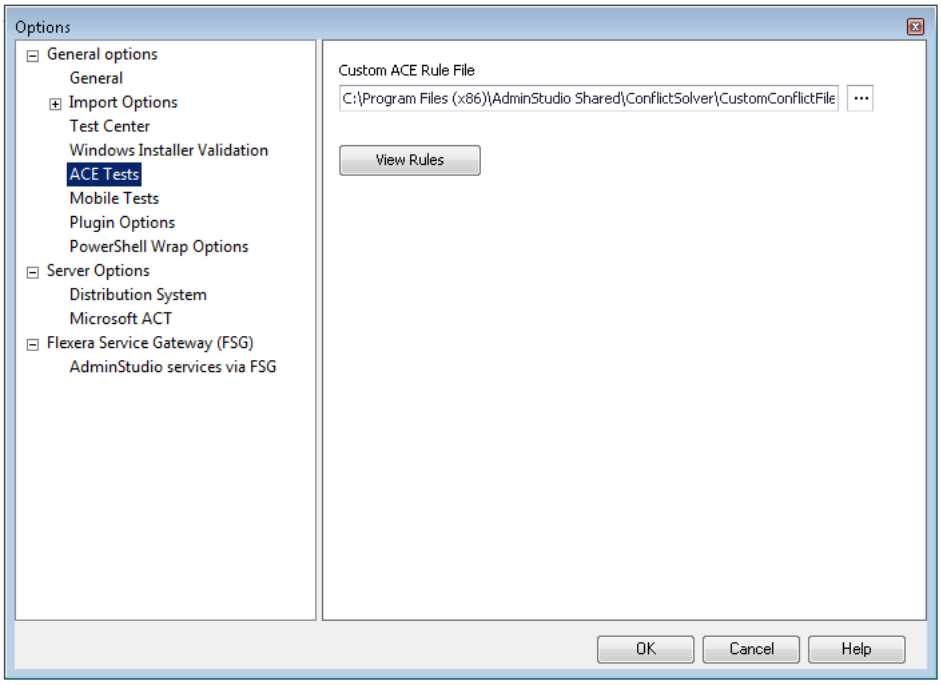


Figure 7-51: Options Dialog Box / ACE Tests Tab

On the **ACE Tests** tab of the Application Manager Options dialog box, you can configure the following options:

Table 7-69 • ACE Tests Tab Properties

Property	Description
Custom ACE Rule File	The user-defined ACE file specified here is run after the pre-defined ACE rules are run. The selection of this user-defined ACE file will affect the default Conflict Types displayed on this dialog (described above), as well as those displayed on the Rules Viewer. By default, a file path to an initially empty user-defined ACE file is provided for you. If you have already created a user-defined ACE, specify the location of that user-defined ACE file to activate it. Only one user-defined ACE file can be active at one time. You use user-defined ACEs to extend the functionality of pre-defined ACEs with company-specific functionality. By selecting different user-defined ACE files, you can organize rules appropriate for individual users in your organization. See Creating Your Own Custom ACE Tests for more information.
View Rules	Click to open the **Rules Viewer** dialog box. On the **Rules Viewer** dialog box, you can click **Add** to open the Rules Wizard, which you can use to add user-defined ACEs to Application Catalog.

Mobile Tests Tab

AdminStudio's mobile risk assessment tests enable you to find out which features a specific mobile app uses, such as telephone, location services, camera, microphone, etc. You can enhance this testing by creating custom tests that combine risk assessment checks with AND or OR operators.

For example, you could create a custom test to see if a mobile application uses a gyroscope OR accelerometer. Or you could create a test that determines whether a mobile application uses location services AND allows location tracking.

On the **Mobile Tests** tab of the **Options** dialog box, you can view or edit existing custom mobile tests. You can also click **New** to open the **Mobile Test Wizard**, which you can use to add new custom mobile tests.

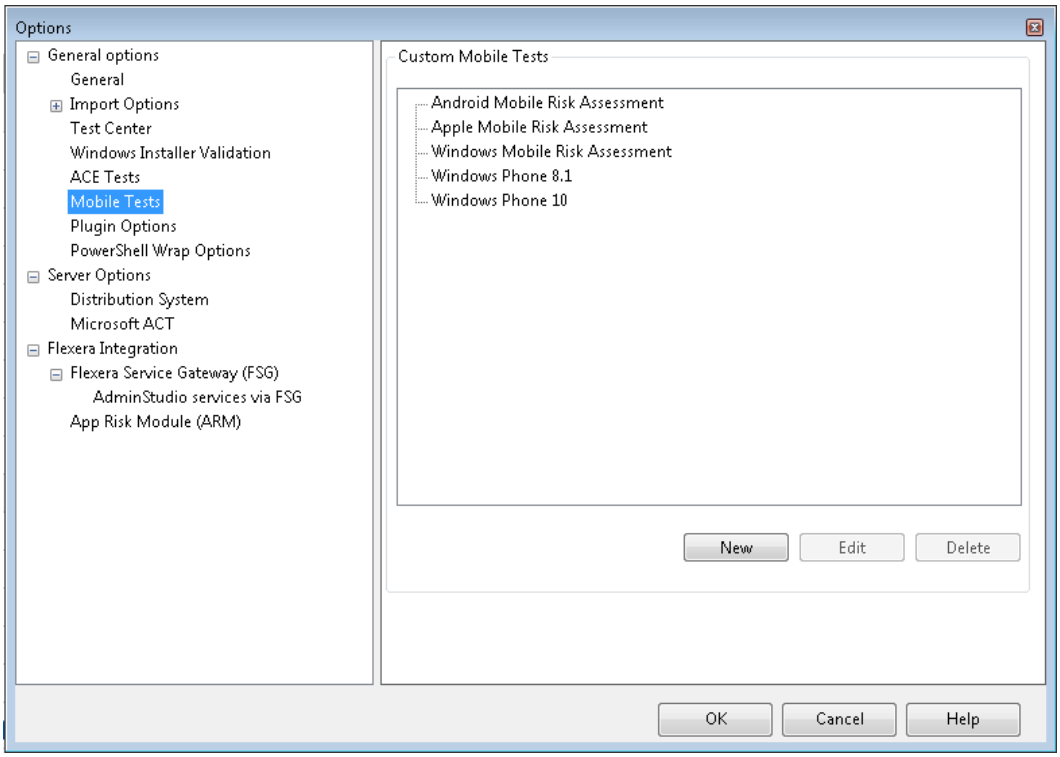


Figure 7-52: Mobile Tests Tab / Options Dialog Box

The **Mobile Tests Tab** includes the following properties.

Table 7-70 • Mobile Tests Tab Properties

Property	Description
Custom Mobile Tests List	List of all custom mobile tests, listed by mobile application test category. The following categories are listed: • Android Mobile Risk Assessment • Apple Mobile Risk Assessment • Windows Mobile Risk Assessment • Windows Phone 8.1 • Windows Phone 10
New	Click to open the **Mobile Test Wizard**, which you can use to create a custom mobile test.
Edit	Click to edit the selected custom mobile test in the Mobile Test Wizard.

Table 7-70 • Mobile Tests Tab Properties

Property	Description
Delete	Click to delete the selected custom mobile test.

Plugin Options Tab

On the **Plugin Options** tab of the Application Manager Options dialog box, you can specify options for AdminStudio plug-ins including the import of Google Android and Apple iOS public store apps, Automated Application Converter, App-V 5 conversion, and Microsoft Intune conversion. The **Plugin Options** tab includes the following subtabs:

- Google Android Link Import Plugin
- Apple iOS Link Import Plugin
- Automated Application Converter Plugin
- App-V 5.x Conversion Plugin
- Microsoft Intune App Conversion Plugin

Google Android Link Import Plugin

On the **Google Android Link Import Plugin** subtab, you can specify the location in your network of Google Android apps that you have already downloaded from the Google Play Store.

If you import deep link to a Google Play Store app that has also already been downloaded to this specified location, AdminStudio will analyze the downloaded binary's data so that additional Analyze tests can be executed for that app. This will result in more accurate test results, and a more successful deployment of the deep link to Workspace ONE or System Center 2012 Configuration Manager.

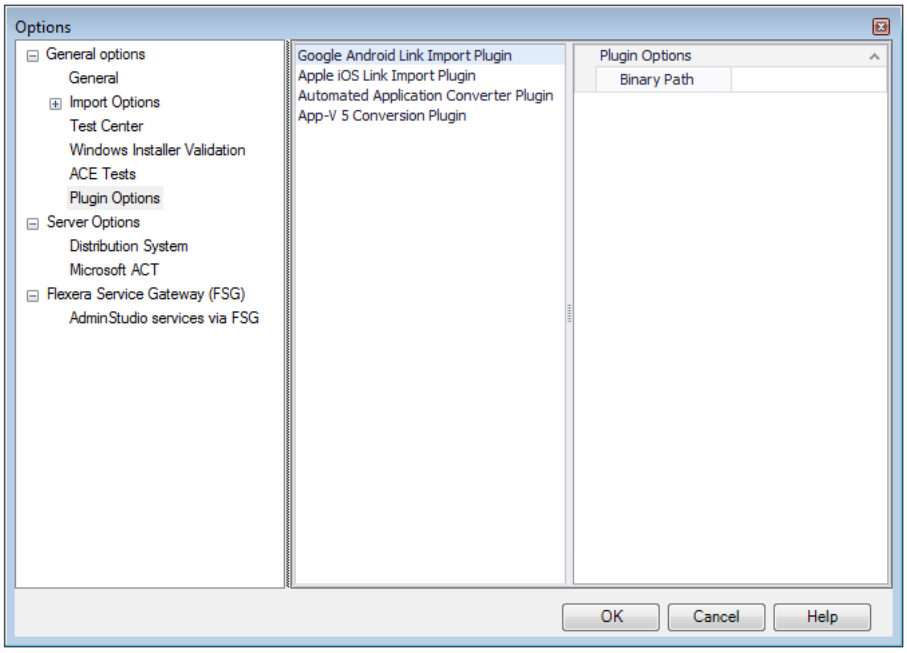


Figure 7-53: Options Dialog Box / Plugin Options Tab / Google Android Link Import Plugin

Apple iOS Link Import Plugin

On the **Apple iOS Link Import Plugin** subtab, you can specify the location in your network of your iTunes Library.

If you import a deep link to an Apple iOS app that has also already been downloaded to this specified iTunes Library, AdminStudio will analyze the downloaded binary's data so that additional Analyze tests can be executed for that app. This will result in more accurate test results, and a more successful deployment of the deep link to Workspace ONE or System Center 2012 Configuration Manager.

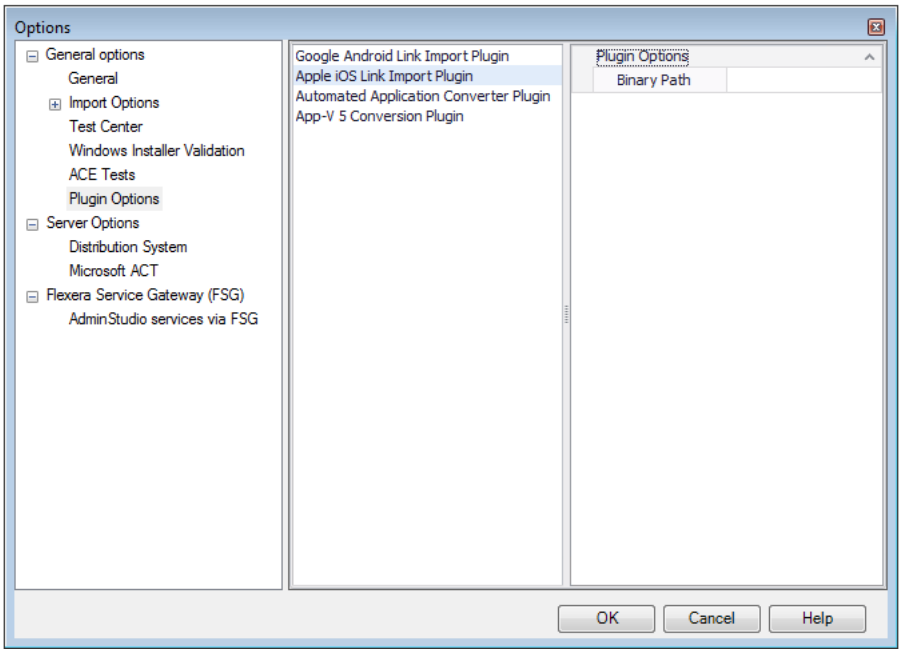


Figure 7-54: Options Dialog Box / Plugin Options Tab / Apple iOS Link Import Plugin

Automated Application Converter Plugin

You can use the Conversion Wizard to convert a Windows Installer package, or multiple packages, to a virtual package in Microsoft App-V (version 4 or 5), Citrix XenApp, or VMware ThinApp format, or to perform repackaging on a virtual machine. Before performing this type of conversion, you need to enter Automatic Application Converter settings on the **Plugins Options > Automated Application Converter Plugin** tab of the **Options** dialog box.

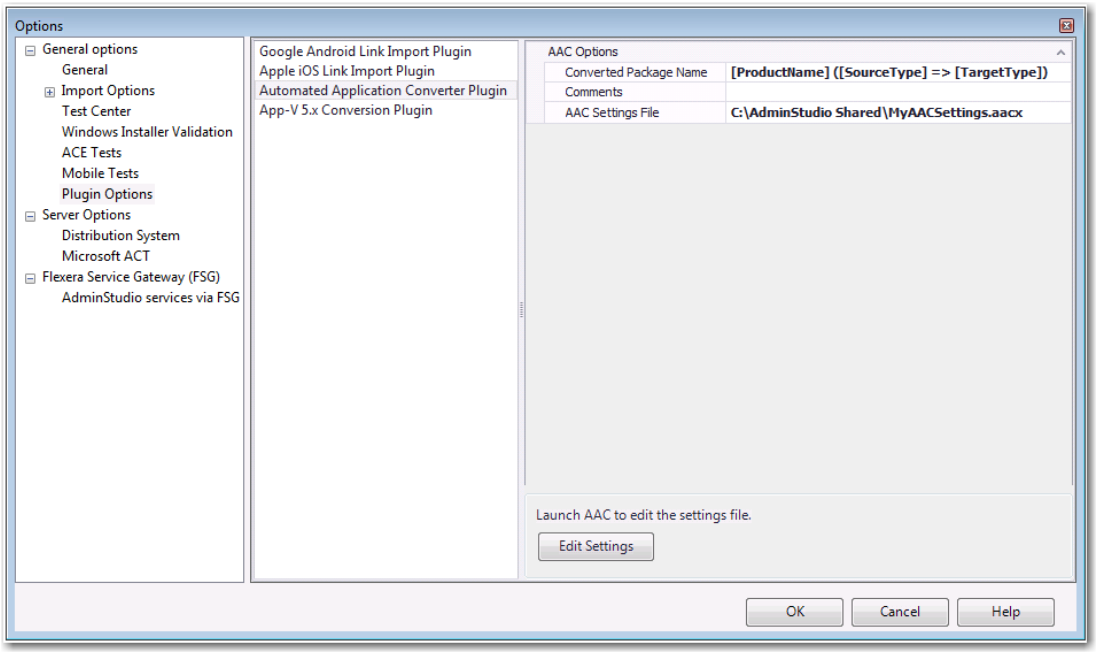


Figure 7-55: Options Dialog Box / Plugin Options Tab / Automated Application Converter Plugin

The **Plugin Options / Automated Application Converter Plugin** tab of the **Options** dialog box includes the following properties:

Table 7-71 • Plugin Options / Automated Application Converter Plugin Properties

Property	Description
Converted Package Name	Enter a name to differentiate the converted version of the package from the original version. By default, this field will be populated with the original package name `[ProductName]`. For example: • `[ProductName]` • `[Manufacturer]_[ProductName]` • `[ProductName]_v5`
Comments	Enter metadata that you would like to add to each converted package. This text will be displayed in the **Administrator Comments** field on the **Package Information** tab of the **Home Deployment Type View** for each package.
AAC Settings File	Select a Automated Application Converter project file (`.aacx`) that contains connection settings to at least one virtual machine. ***Note*** • *For more information, see Creating an Automated Application Converter Settings File.*
Edit Settings	Click to edit the specified Automated Application Converter project file. See Editing the Default Automated Application Converter Settings File From Application Catalog.

App-V 5.x Conversion Plugin

You can use the Conversion Wizard to upgrade an App-V 4.x package (`.sft`) to App-V 5.0 format (`.appv`). Before performing this type of conversion, you need to enter App-V 5.x conversion plug-in settings on the **Plugins Options** tab of the **Options** dialog box.

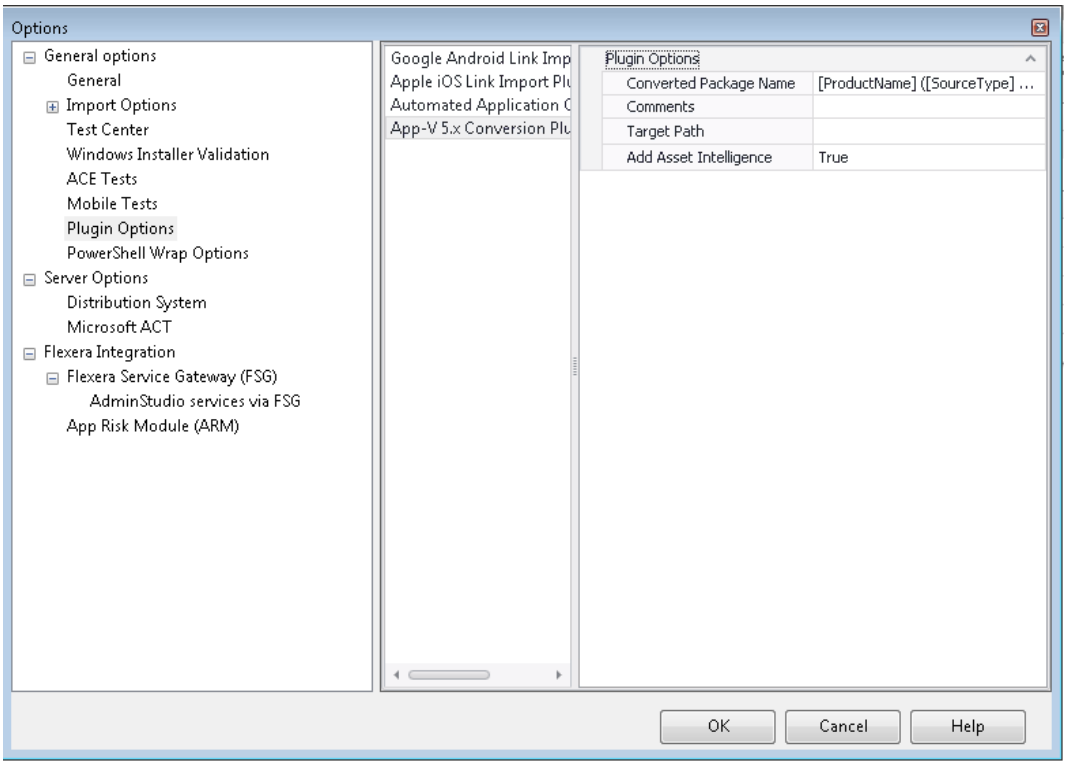


Figure 7-56: Options Dialog Box / Plugin Options Tab / App-V 5.x Conversion Plugin

The **Plugin Options / App-V 5.x Conversion Plugin** tab of the **Options** dialog box includes the following properties:

Table 7-72 • Plugin Options / App-V 5.x Conversion Plugin Properties

Property	Description
New Package Name	Enter a name to differentiate the converted version of the package from the original version. By default, this field will be populated with the original package name `[ProductName]`. For example: • `[ProductName]` • `[ProductName]_v5` • `[Manufacturer]_[ProductName]_v5`
Comments	Enter metadata that you would like to add to each converted package. This text will be displayed in the **Administrator Comments** field on the **Package Information** tab of the **Home Deployment Type View** for each package.
Target Path	Specify the output folder where you want the converted packages to be located.

Table 7-72 • Plugin Options / App-V 5.x Conversion Plugin Properties

Property	Description
Asset Intelligence	Asset intelligence is used to enhance the inventory capabilities of Microsoft System Center 2012 Configuration Manager by extending hardware inventory and adding license management functionality. The asset intelligence features can report application data such as digital PID, MSI product codes, and publisher names for each virtual application registered on a client computer. To add asset intelligence information to a converted App-V 5.x package, set this option to **True**.

Microsoft Intune App Conversion Plugin

On the **Microsoft Intune App Conversion Plugin** subtab, you specify Intune app conversion options.

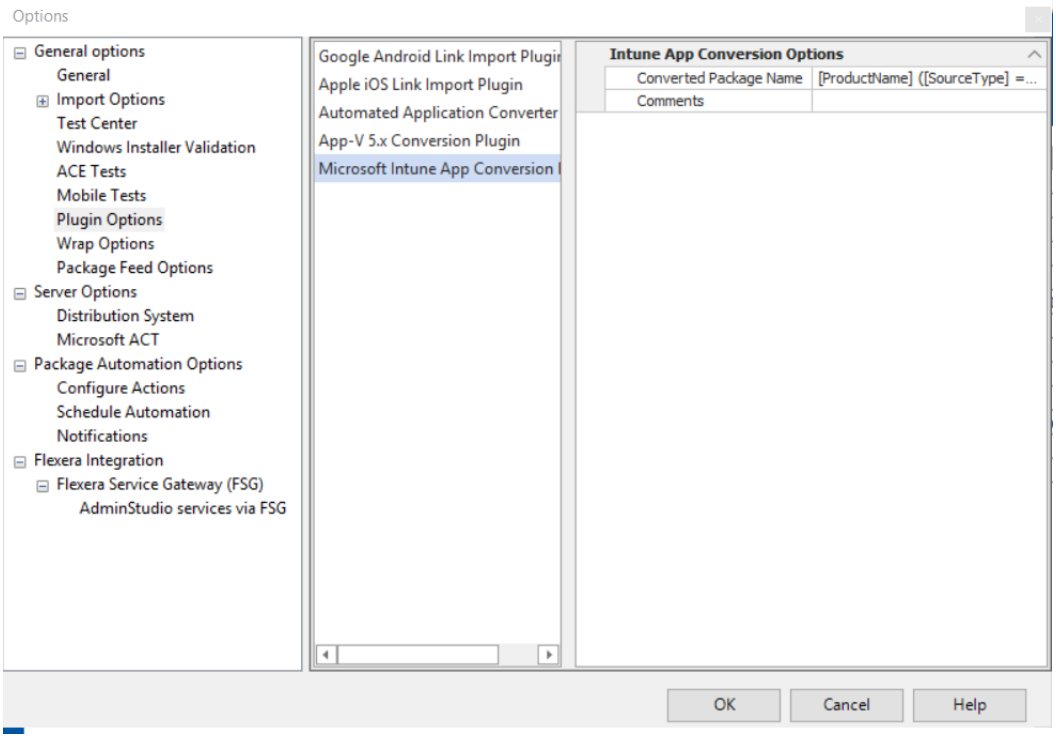


Figure 7-57: Options Dialog Box / Plugin Options Tab / Microsoft Intune App Conversion Options

The **Microsoft Intune App Conversion** subtab has the following options.

Table 7-73 • Microsoft Intune App Conversion Options

Option	Description
Converted Package Name	Specify the format of the converted package name to differentiate it from the original package name. By default, the format is: `[ProductName] ([SourceType] => [TargetType])`
Comments	Specify any comments you want to add to the converted package.

Package Feed Options Tab

Edition •

On the **Package Feed Options** tab, you can specify the folder path to download the setup files from the Package Feed Module.

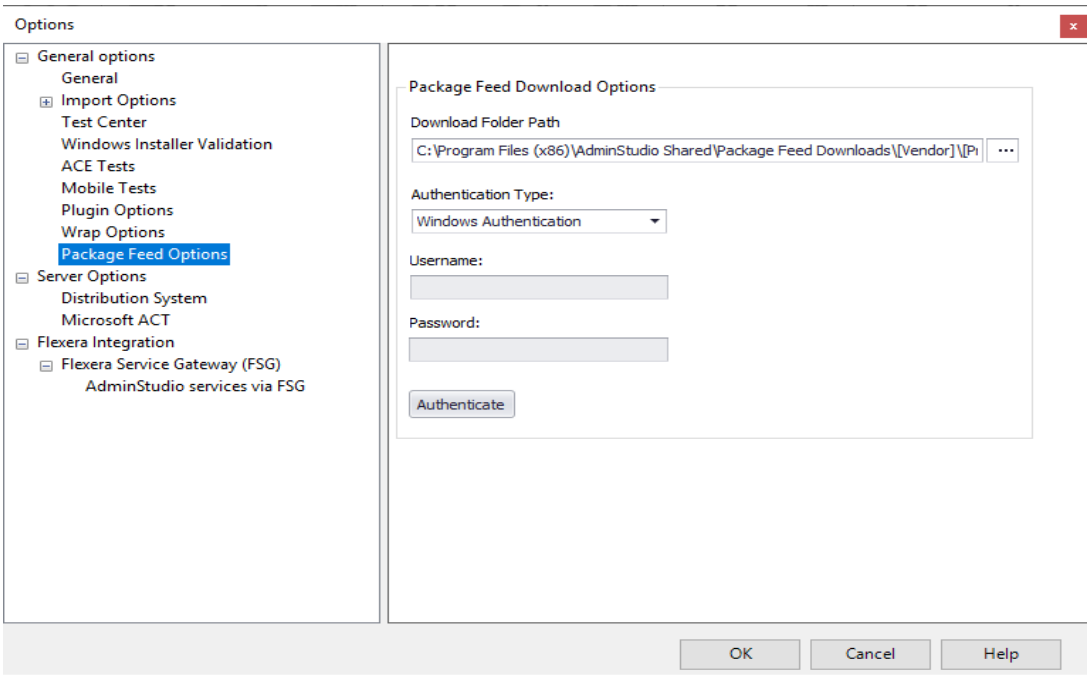


Figure 7-58: Options Dialog Box / Package Feed Options

The **Package Feed Options** includes the following properties:

Table 7-74 • Package Feed Options Tab

Option	Description
Download Folder Path	Packages will get downloaded to this location, You can also edit the default download path: • **Default path** - C:\Program Files (x86)\AdminStudio Shared\Package Feed Downloads • **Default Place Holders** - [Vendor]\[Product Name]\[Version] • **Other Valid Place Holders** - [Platform]\[Setup Type]\[Language] For example the default download folder path is **C:\Program Files (x86)\AdminStudio Shared\Package Feed Downloads\[Vendor]\[Product Name]\[Version]**. ***Note*** *• You can define the folder path of your choice followed by the valid place holders.*
Authentication Type	Choose one of the following options to identify the authentication type you are going to use to access the download folder path: • **Server Authentication**—Choose this option if you want to use server login identification to access the server where the folder path is located. Then enter the appropriate **Username** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to access the server where the folder path is located.

See Also
Importing an Application Using The Package Feed Module

Wrap Options Tab

On the **Wrap Options** tab, you can specify the location of the PowerShell and Exe template, the output directory for PowerShell wrapped and Exe packages, and set an option to automatically wrap a package during import.

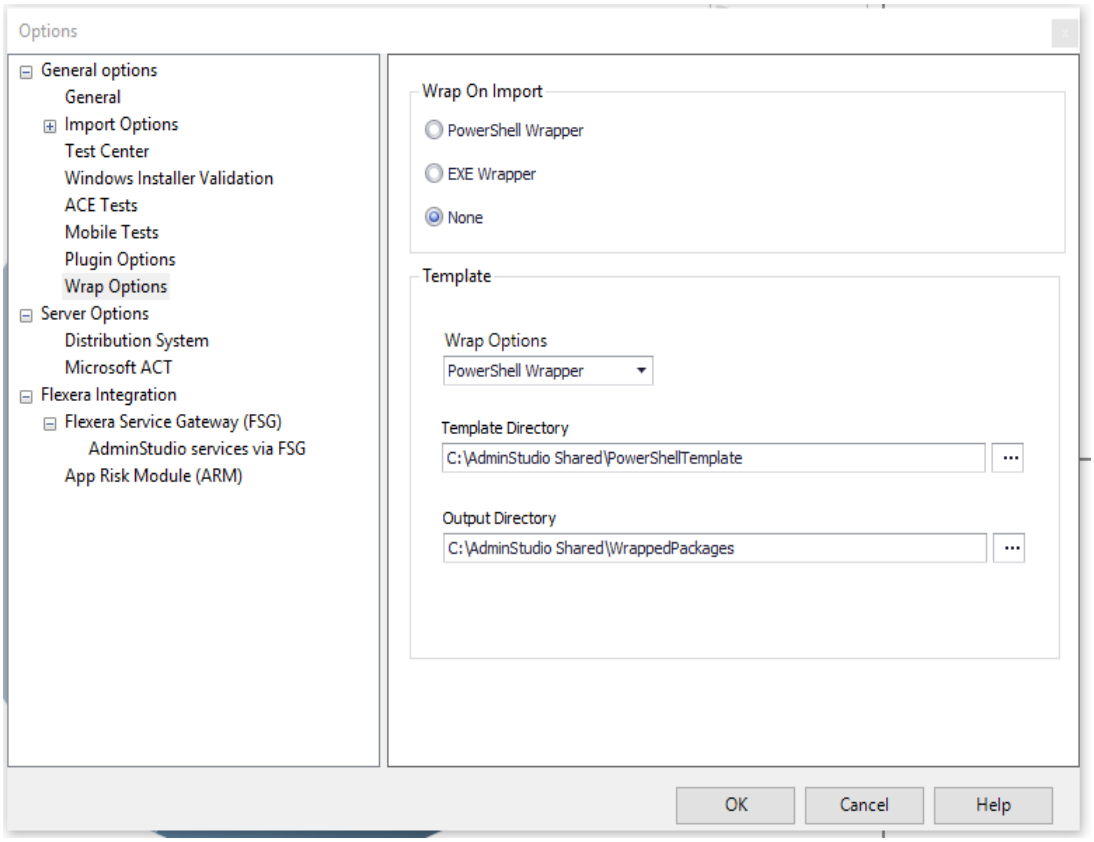


Figure 8: Wrap Options Tab of Options Dialog Box

The **Wrap Options** tab includes the following properties:

Table 7-75 • Wrap Options Tab

Option	Description
Template Directory	Specify the location of the template that you want to use to create PowerShell and exe script files. A default template, `Deploy-Application.ps1`, along with other necessary files are provided in the following location: `C:\Program Files (x86)\AdminStudio Shared\Template\`
Output Directory	Specify the location where the PowerShell-wrapped packages that you create using AdminStudio will be stored. By default, the location is: `C:\Program Files (x86)\AdminStudio Shared\WrappedPackages\` You can click the browse button and select a different directory.

Table 7-75 • Wrap Options Tab

Option	Description
Wrap on Import	Select this option if you want to automatically convert Windows Installer packages (`.msi`) and complex installation packages (`.exe`) to PowerShell wrapped packages (PowerShell script `.ps1` files) upon import into the Application Catalog. By default, this option is not selected.

Server Options / Distribution System Tab

On the **Distribution System** tab of the Application Manager Options dialog box, you can define multiple named connections to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, Microsoft App-V Server, Symantec Altiris Server, Microsoft Intune, Custom Distribution System, and Workspace ONE Server distribution systems. This enables you to both have multiple connections easily available during import and distribution, and to refer to those connection settings by name in AdminStudio PowerShell Cmdlets commands.

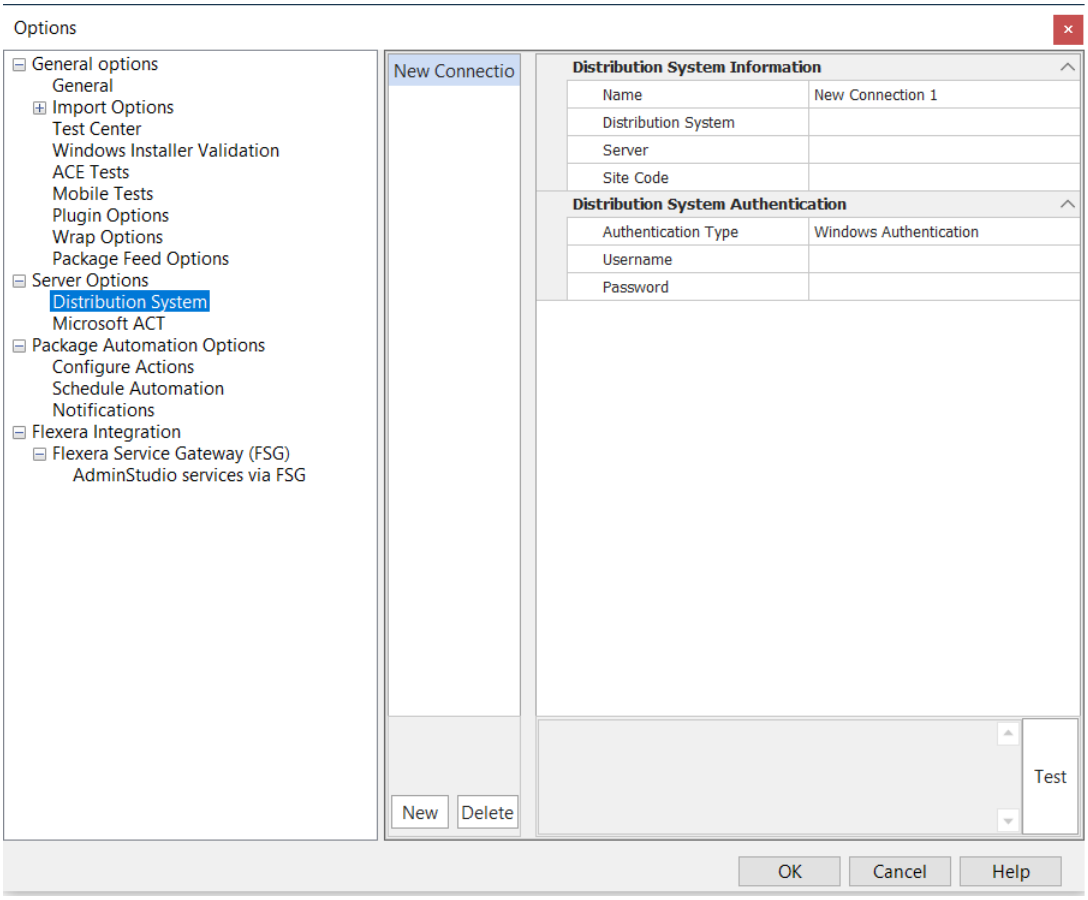


Figure 7-1: Options Dialog Box / Distribution System Tab

The **Server Options / Distribution System** tab of the **Options** dialog box includes the following properties:

Table 7-76 • Server Options / Distribution System Tab Properties

Property	Description
Name	Enter a name to identify this named connection to a distribution system.
Distribution System	Select one of the following to identify the distribution system technology of this named connection: • **Workspace ONE Distribution Plugin** • **Altiris Distribution Plugin** • **App-V Distribution Plugin** • **Casper Distribution Plugin** • **Custom Distribution Plugin** • **Intune Distribution Plugin** • **ConfigMgr Deployment Plugin** • **XenApp Deployment Plugin**
Server	Enter the name of your distribution system server.
Site Code	Enter the code that identifies your distribution system site. ***Note*** *• If you are creating a named connection to a Microsoft App-V, Citrix XenApp or Altiris server, leave the* ***Site Code*** *field blank. This field is not displayed when creating a named connection to a Casper Server.*

Table 7-76 • Server Options / Distribution System Tab Properties

Property	Description
Distribution System Authentication / Authentication Type	Choose one of the following options to identify the authentication type you are going to use to access the specified distribution system: • **Server Authentication**—Choose this option if you want to use server login identification to log into this server. Then enter the appropriate **Username** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this server. • **Global Environment**—Choose this option if you want to set azure environment as a global environment. • **Government Environment**—Choose this option if you want to set azure environment as a government environment. • **User Account**—Choose this option if you want to use user account login identification to log into this server. Then enter the appropriate **Client Id** and **Tenant ID/Name**. • **Client Secret**—Choose this option if you want to client secret login identification to log into this server. Then enter the appropriate **Client Id**, **Tenant ID/Name, and Client Secret**.
Distribution Point	(Casper only) Enter the Casper distribution point you want to distribute packages to. ***Note*** *• Casper supports multiple server infrastructures, but AdminStudio only supports the File Share Distribution Points infrastructure, and copies packages to a UNC File Share Distribution Point in Casper. AdminStudio currently does not support copying packages to JDS Instances, Cloud Distribution Points, Software Update Servers, or NetBoot Servers.*
Publish Location Information / Authentication Type	Choose one of the following options to identify the authentication type you are going to use to access the shared location where you will be publishing packages during distribution: • **Server Authentication**—Choose this option if you want to use server login identification to log into this server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this server. ***Note*** *• The fields in the* ***Publish Location Information*** *section are not required when setting up a connection to Workspace ONE Server. Applications are published directly to the Workspace ONE Server, not to a shared location.*

Table 7-76 • Server Options / Distribution System Tab Properties

Property	Description
Script Information	(Custom Distribution Plugin only) Enter information about the script file that you wish to be executed after a package is copied to the path specified in the **Location to Publish Packages** field. • **Script File**—Click Browse and select the script file (`.ps1`) that you want to add. This is an optional field. ***Note*** *• It is required that the specified PowerShell script file must be present in the AdminStudio installed machine.* ***Note*** *• The specified PowerShell script file will be executed locally on the AdminStudio installed machine during publish.* • **Script Parameters**—Enter the parameters used in the PowerShell script with the appropriate AdminStudio placeholders. This is an optional field. The **Script Parameters** field must be used to pass the AdminStudio placeholders value to the parameters in the PowerShell script file. During the execution of the script, AdminStudio will replace the specified parameters with the value of the placeholders for the package being published. • [ProductName] • [Vendor] • [Version] • [PackagePublishPath] • [InstallCmdLine] • [UnInstallCmdLine] • [Platform] • [RepairCommandLine] • [Language] • [PackagePublishFolder] • [TransformFilePath] • [IconPath] • [SetupType] ***Note*** *• Enter the script parameters in comma-separated format with no gap:* *For Example:* `-ProductName [ProductName],-Vendor [Vendor],-Version [Version]` ***Note*** *• If you use any other delimiter than a comma, an error message will be generated.*

Table 7-76 • Server Options / Distribution System Tab Properties

Property	Description
Location to Publish Packages	Enter or browse to the shared location where you will be publishing packages during distribution. ***Note*** *• The fields in the* ***Publish Location Information*** *section are not required when setting up a connection to Workspace ONE Server. Applications are published directly to the Workspace ONE Server, not to a shared location.* ***Note*** *• When publishing to a Casper server, AdminStudio publishes packages to a directory named Packages.* ***Note*** *• You can add the following placeholders in the shared location path to publish packages. These placeholders are applicable only for* ***Custom Distribution Plugin****.* • *[ProductName]* • *[Vendor]* • *[Version]* • *[Platform]* • *[Setuptype]* • *[Language]* For Example: When publishing 7-Zip v19.00, if the following shared path is specified in the Location to Publish Package field: \\10.XXX.XXX.XXX\PackageRepo\[Vendor]\[ProductName]\[Version] then the placeholders will be resolved to create the shared path as shown below: \\10.XXX.XXX.XXX\PackageRepo\Igor Pavlov\7-Zip\19.00
New	Click to create a new named connection. A set of empty connection setting fields are displayed.
Delete	Click to delete the selected named connection.
Test	Click to test the connection settings of the selected named connection. Messages will be displayed in the Test area to confirm the connection.

Microsoft App-V Server Distribution Requirements

In order for you to distribute packages to a Microsoft App-V Server, the WinRM service must be running, and the App-V Server must be in the list of trusted hosts. Both of these can be accomplished from PowerShell by running the following command:

```
set-item wsman:\localhost\Client\TrustedHosts –value <Machine Name>
```

The following image is an example of starting the WinRM service in PowerShell.

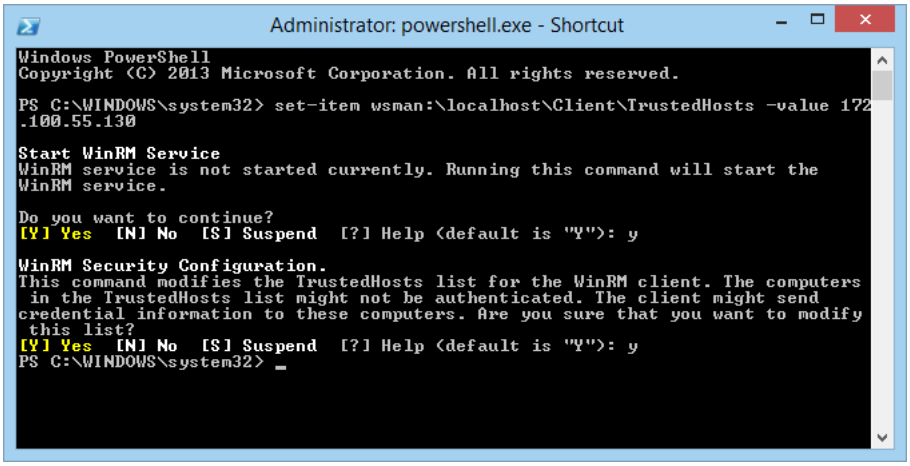


Figure 7-2: Starting the WinRM Service in PowerShell

Server Options / Microsoft ACT Tab

On the **Server Options / Microsoft ACT** tab of the Application Manager Options dialog box, enter connection settings for your Microsoft ACT (Application Compatibility Toolkit) database. This will enable AdminStudio to display data from the ACT database in Application Manager Analyze views and reports.

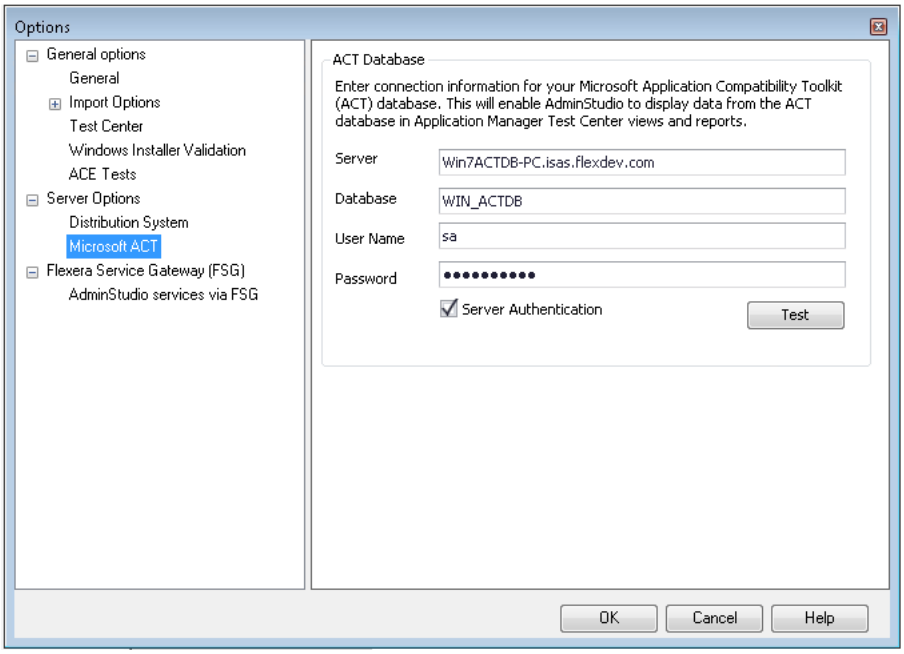


Figure 7-3: Options Dialog Box / Microsoft ACT Tab

The **Server Options / Microsoft ACT** tab includes the following properties:

Table 7-77 • Server Options / Microsoft ACT Tab Properties

Property	Description
Server	Enter the name of the server that contains your ACT database.
Database	Enter the name of your ACT database.
Server Authentication	Do one of the following: • **Server Authentication**—Select this option if you want to use database server login identification to log into this server. Then enter the appropriate **User Name** and **Password**. • **Windows Authentication**—If you want to use Windows network authentication (your network login ID) to log into this database server, leave the **Server Authentication** field unselected and leave the **User Name** and **Password** fields blank.

Note • *If you create a new Application Catalog, you will need to reenter this connection information.*

Package Automation Options / Monitored Directory

The following properties are listed on the **Monitored Directory** tab.

Table 7-78 • Monitored Directory Tab

Properties	Description
Shared Directory	Enter the path or click the browse button (**...**) to browse to the path.
Destination Group	By clicking the browse button (**...**), the **Select Destination Group** dialog box opens. Enter name of the group and click **Ok**. ***Note*** • *You can create multiple groups by clicking on the **New Group** button.* Click **Create subgroup based on source folder structure** check box, if you want to create subgroups in the folder structure.

Table 7-78 • Monitored Directory Tab

Properties	Description
Authentication Type	Select one of the following type: • **Server Authentication**— Choose to use server login identification for authentication. • **Windows Authentication**— Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Username and Password**— If you choose Server Authentication, enter the appropriate Username and Password.
Test Connection	Click to validate the Monitored Directory connection information.
Execute	Click to execute the packages in the Monitored Directory.

Note *• For detailed information on monitored directory, see Monitored Directory for Package Automation.*

Package Automation Options / Configure Actions

The following properties are listed on the **Configure Actions** tab.

Table 7-79 • Configure Actions Tab

Properties	Description
Select action to configure	Select the automation action to configure: • Import • Test • Wrap • Convert • Publish
Select Wrapper	Select the desired wrapper format: • PowerShell Wrapper • EXE Wrapper ***Note*** *• This field is only visible when* ***Wrap*** *is selected in* ***Select action to configure*** *list.*

Table 7-79 • Configure Actions Tab

Properties	Description
Target package format	Select one of the following target package formats: • Microsoft App-V version 4.x • Microsoft App-V version 4.x • Citrix XenApp • VMWare ThinApp • Microsoft Windows Installer • Microsoft MSIX • Microsoft Intune Windows App ***Note*** *• This field is only visible when* ***Convert*** *is selected in* ***Select action to configure*** *list.*
Distribution System	Select one of the following distribution systems to use for automated package distribution: • Workspace ONE Distribution Plugin • Intune Distribution Plugin • ConfigMgr Deployment Plugin ***Note*** *• This field is only visible when* ***Publish*** *is selected in* ***Select action to configure*** *list.*
Destination Group	When **Import** is selected in the **Select action to configure** list, you can arrange the placeholders in this field. The default placeholders are: `[Vendor]\[Product Name]\[Version]` When **Publish** is selected in the **Select action to configure** list, specify the group in the specified distribution system into which you want to publish packages. ***Note*** *• This field is only visible when* ***Import*** *or* ***Publish*** *is selected in* ***Select action to configure*** *list.*
Add this action to automation	Select this option to enable automation of the action selected in the **Select action to configure** list field.

Table 7-79 • Configure Actions Tab

Properties	Description
Automatically create a custom transform file	Selecting **Automatically create a custom transform file** will automatically create a transform (.mst) file based on the selections made in the customization wizard of the subscribed packages and imported into the catalog along with the MSI package into the catalog. This check box will be selected by default.
Include Monitored Directory for Scheduled Automation	Selecting **Include Monitored Directory for Scheduled Automation** will execute packages that are in Monitored Directory through Scheduled Package Automation.

Note *• For detailed information on scheduling automated processes, see Configuring Actions for Automation.*

Package Automation Options / Schedule Automation

On the **Schedule Automation** tab, you can specify the scheduling of automated tasks.

Table 7-80 • Schedule Automation Tab

Properties	Description
Frequency	Select the frequency when you want to automated tasks to occur. • Once • Daily • Weekly • Monthly
Recur every x days	When **Daily** is selected as the **Frequency**, enter a numeric value to indicate how frequently (in days) you want the automated task to occur.
On Date	When **Monthly** is selected as the **Frequency**, select this option and select the first date that you want the monthly automated task to occur. It will repeat on the same day each subsequent month.
Start Time	Select the time of day when you want the automated task to occur.
Schedule	Click to start the scheduled automation process.
Stop	Click to stop the scheduled automation process.

Note *• For detailed information on scheduling automated processes, see Scheduling Automation.*

Package Automation Options / Notifications

On the **Notifications** tab, you can configure notification settings for automated tasks.

Table 7-81 • Notifications Tab

Properties	Description
SMTP Server	Enter the SMTP Server details.
Authentication	Select one of the following option to configure the notifications: • Server Authentication • Anonymous
Domain	Enter the domain name.
User Name	Enter the user name.
Password	Enter the password.
From E-mail ID	Enter the email address of the sender.
To E-mail ID(s)	Enter the email addresses of the recipient.
SMTP Server Ports	Enter SMTP server ports.
Use SSL	Select SSL option.
Clear	Click to clear the selected options.
Send Test E-mail	Click to verify the email addresses.

Note *• For detailed information on configuring notifications for automated tasks, see Specifying Notification Settings for Automation.*

Software Repository Tab

On the **Software Repository** tab of the Application Manager Options dialog box, you can view and edit the **Software Repository Location** for the connected Application Catalog and the **Proxy Account** to access that location.

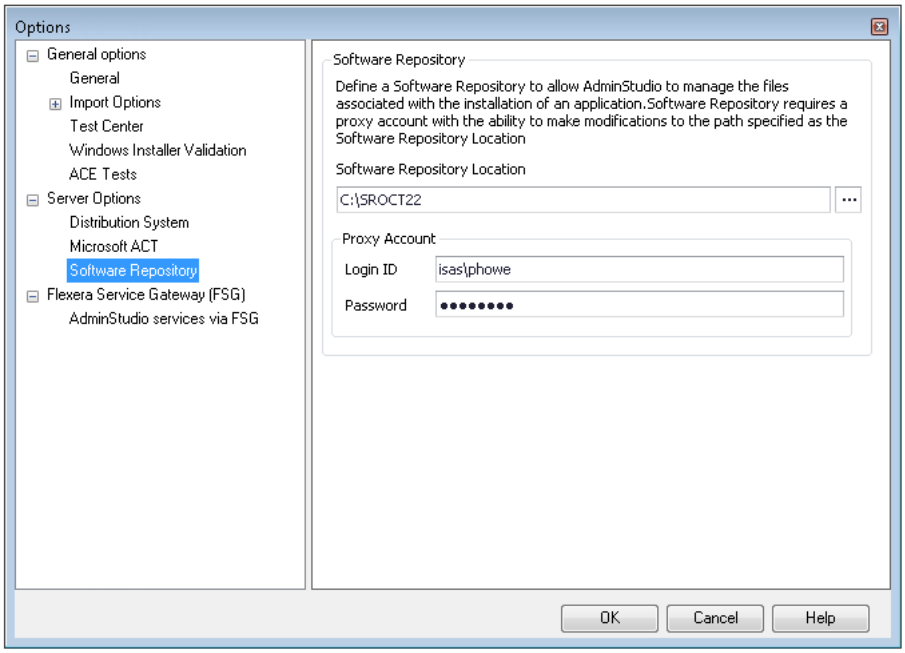


Figure 7-4: Options Dialog Box / Software Repository Tab

Note • *This tab is only displayed if you enabled the Software Repository when creating the connected Application Catalog (by selecting the **Enable Software Repository** option on the **Select Software Repository Location** panel of the Application Catalog Wizard).*

The **Software Repository** tab includes the following options:

Table 7-82 • Software Repository Tab Properties

Option	Description
Software Repository Location	Enter or select the directory location of the Software Repository for this Application Catalog. All of the files associated with a package that is imported into the Application Catalog are copied to this location. This allows you to manage those files, preventing them from getting modified or lost.

Table 7-82 • Software Repository Tab Properties (cont.)

Option	Description
Proxy Account	Specify the **Login ID** and **Password** for a **Proxy Account** that AdminStudio can use to access and modify the specified **Software Repository Location** folder. ***Note*** *• You cannot use Windows Authentication for this Proxy Account.* ***Important*** *• The Proxy Account needs full control on the* ***Software Repository Location*** *folder at the directory level as well as at the sharing level. Only such accounts can be used as a Proxy Account to access the* ***Software Repository Location*** *directory.*

Flexera Integration / Flexera Service Gateway (FSG) Tab

If you have also purchased Flexnet Manager Suite (FNMS), App Portal and/or Workflow Manager, you can connect to the Flexera Service Gateway and communicate with those applications.

On the **Flexera Integration/Flexera Service Gateway (FSG)** tab of the Application Manager Options dialog box, you can configure connection to Flexera Service Gateway and FlexNet Manager Suite in the following tabs:

- FSG Configuration
- FNMS Configuration

FSG Configuration

You enter the login credentials for your Flexera Service Gateway server on the **FSG Configuration** tab.

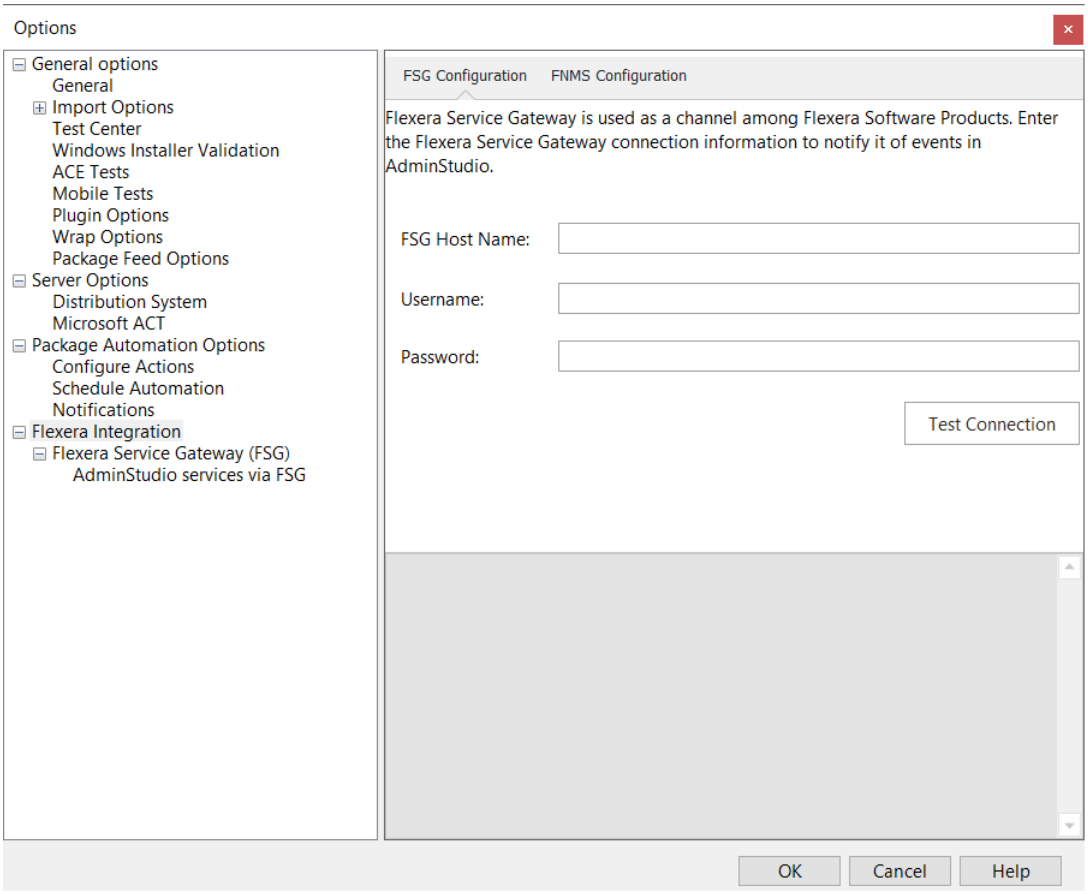


Figure 7-5: Flexera Integration / Flexera Service Gateway (FSG) Tab

Note • *For a detailed description of the benefits of communicating with FlexNet Manager Suite and App Portal, see Integrating with Other Flexera Applications via the Flexera Service Gateway.*

The **FSG Configuration** tab includes the following properties:

Table 7-83 • FSG Configuration Tab Properties

Property	Description
FSG Host Name	Enter the name or URL of your Flexera Service Gateway server. • **Port number**—If your System Administrator has installed Flexera Service Gateway using a different port than the default port, enter the appropriate port number at the end of the URL, preceded by a colon, such as: `172.300.40.501:8484` • **DNS name vs. IP address**—You can use a DNS name or an IP address. You should specify a DNS name if all clients are on the same domain and can resolve it; otherwise, use an IP address. • **HTTPS**—You should always use `https`. ***Note*** *• The Flexera Service Gateway installer is downloaded from the Flexera Product & License Center.*
User Name	Enter the FSG user name. ***Note*** *• Unless your System Administrator has provided you with a specific User Name to use, enter the default value of* **`admin`**.
Password	Enter the valid password. ***Note*** *• Unless your System Administrator has provided you with a specific User Name to use, enter the default value of* **`admin`**.
Test Connection	Click to validate the Flexera Service Gateway connection information.

FNMS Configuration

On the **FNMS Configuration** tab you can provide your FlexNet Manager Suite details to establish a successful connection.

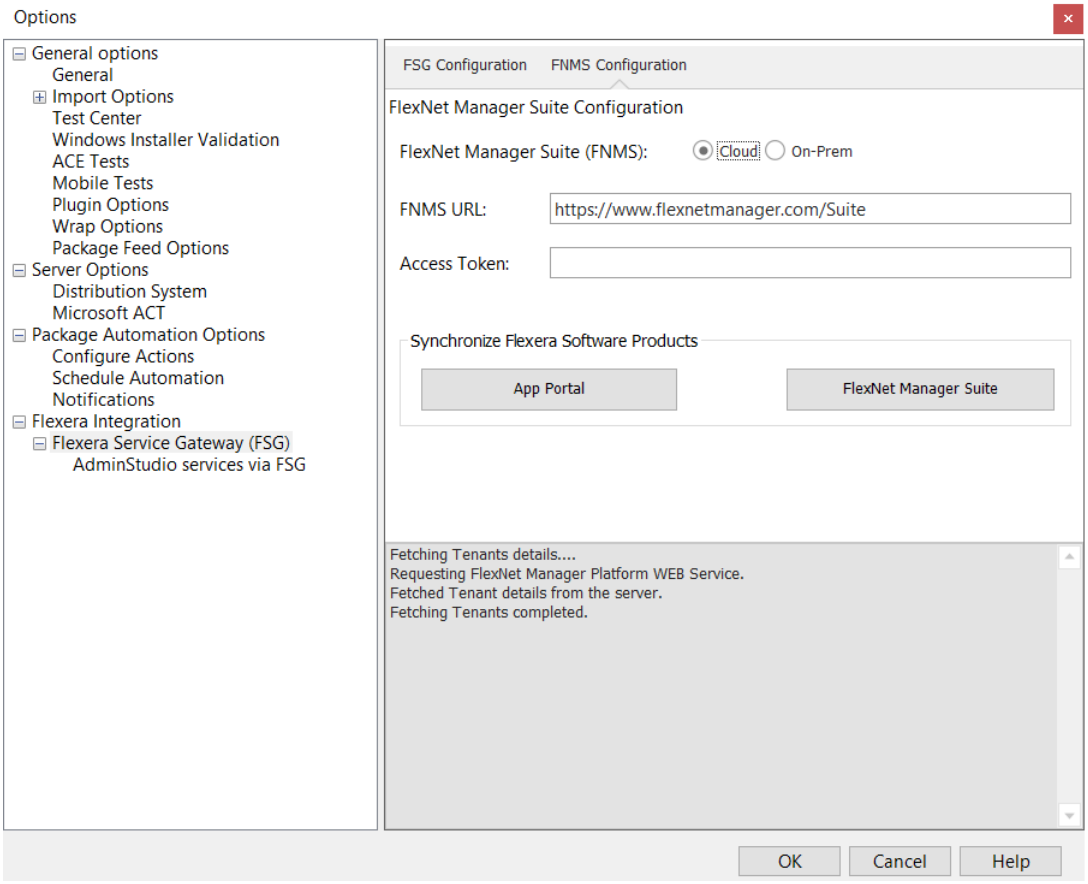


Figure 7-6: Flexera Integration / FNMS Configuration Tab (Cloud)

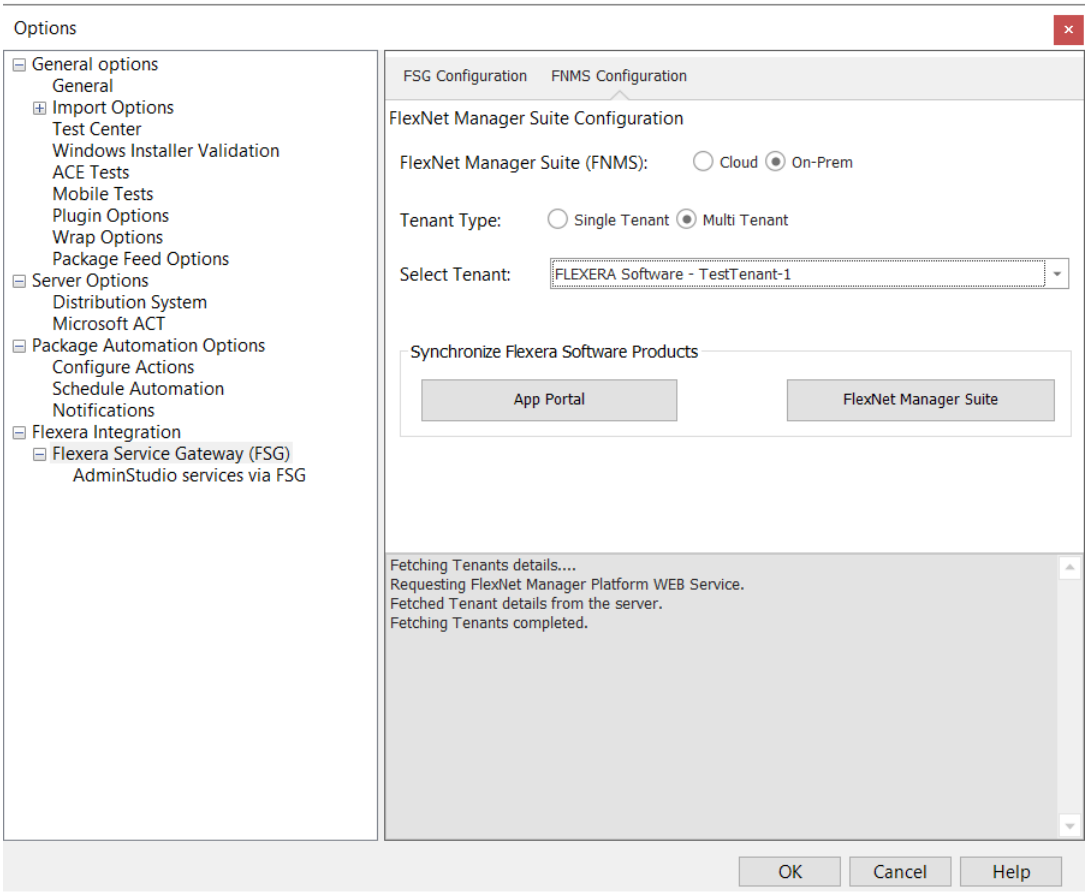


Figure 7-7: Flexera Integration / FNMS Configuration Tab (On-Prem)

Note • *For a detailed description of the benefits of communicating with FlexNet Manager Suite and App Portal, see Integrating with Other Flexera Applications via the Flexera Service Gateway.*

The **FNMS Configuration** tab includes the following properties:

Table 7-84 • FNMS Configuration Tab Properties

Property	Description
FlexNet Manager Suite (FNMS)	Specify the type of FlexNet Manager Suite instance that you wish to integrate with AdminStudio. • Cloud • On-Premises ***Note*** • *FNMS Cloud will be selected by default.*

Table 7-84 • FNMS Configuration Tab Properties

Property	Description
FNMS URL	This property is applicable to FNMS Cloud instance only. By default this property will display: https://www.flexnetmanager.com/suite.
Access Token	Enter the access token provided by your system administrator. Access token can be obtained from the FNMS Cloud instance.
Tenant Type	Select the tenant type of the on-prem FNMS set up in your environment: • Single Tenant • Multi Tenant
Select Tenant	When connecting to a Multi-tenant On-Prem FNMS, click on the drop down to see the list of tenants from your Multi-tenant On-Prem FNMS instance. Select the desired tenant from the list. For Single Tenant On-Prem FNMS instance, leave this property as none.
App Portal	Click to create a catalog item in App Portal for all of the applications in the Application Catalog that were published to ConfigMgr (Formerly called as System Center Configuration Manager) before the Flexera Service Gateway connection information was entered. • If, when you click this button, valid System Center Configuration Manager connection information is not entered on the **Distribution System** tab of the **Options** dialog box, a message will appear prompting you to enter connection information. • If you are not connected to the Flexera Service Gateway or if the Flexera Service Gateway is not available, an error message will be displayed stating that the sync has failed. ***Note*** *• After valid Flexera Service Gateway connection information is entered, each time you publish an application to Configuration Manager, a catalog item for that application will automatically be created in App Portal.*
FlexNet Manager Suite	Click to search the FlexNet Manager Suite Application Recognition Library (ARL) to locate and obtain the **Flexera Identifier** for the Application Catalog's existing applications. ***Note*** *• After valid Flexera Service Gateway connection information is entered, each time you import an application into the Application Catalog, the* ***Flexera Identifier*** *for that application will be obtained from FlexNet Manager Suite.*

AdminStudio Services via FSG Tab

In order for AdminStudio to communicate and share package data with another application, you need to identify a shared Application Catalog database that both products can access.

You identify the shared Application Catalog on the **AdminStudio Services via FSG** tab of the Application Manager Options dialog box:

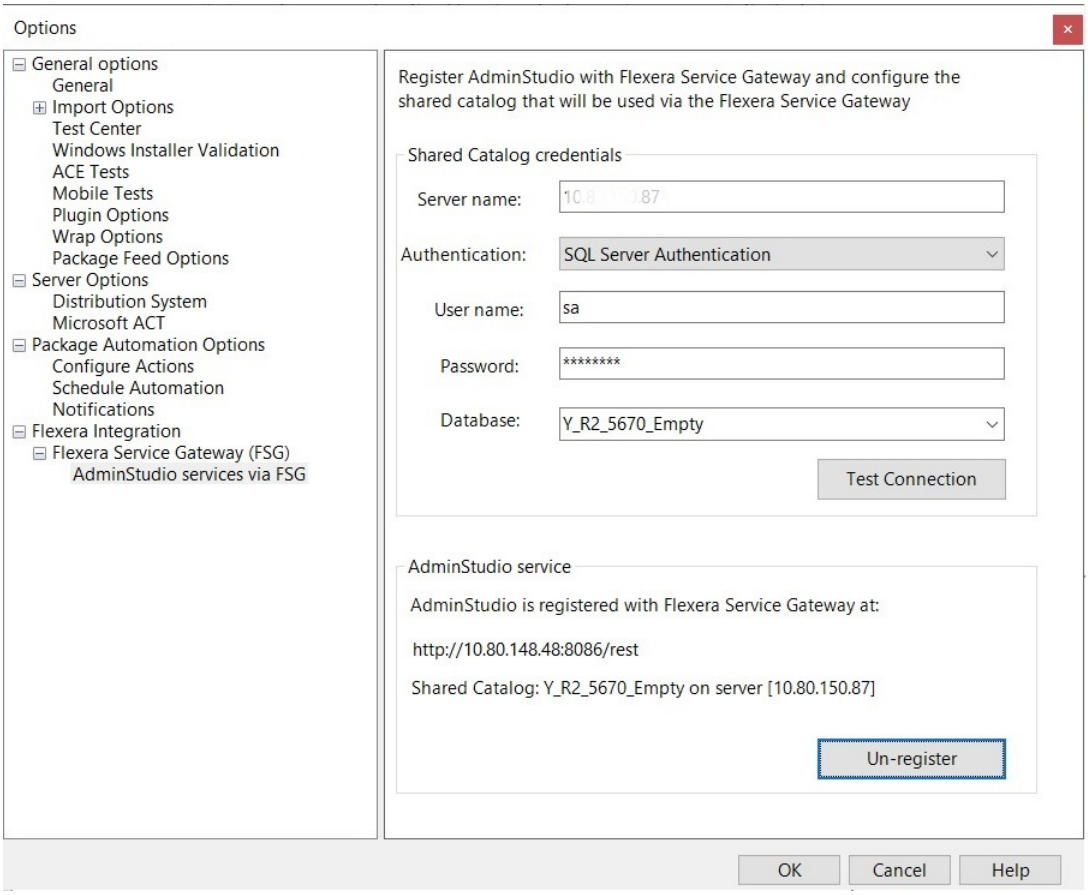


Figure 7-8: AdminStudio Services via FSG Tab of Options Dialog Box

The **AdminStudio Services via FSG** tab includes the following properties:

Table 7-85 • AdminStudio Services via FSG Tab

Property	Description
Server name	Lists the name of the server that contains the shared Application Catalog database that has been registered with the Flexera Service Gateway.
Authentication	Specifies the type of authentication to access the shared Application Catalog database as either **Windows Authentication** or **SQL Server Authentication**. If **SQL Server Authentication** is selected, the **User name** and **Password** credentials must also be entered.
Database	Name of the shared Application Catalog database that has been registered with the Flexera Service Gateway.
Test Connection	Click to test the connection to the shared Application Catalog.

Table 7-85 • AdminStudio Services via FSG Tab

Property	Description
Un-register	Click to unregister the specified Application Catalog with the Flexera Service Gateway.

References Dialog Box

The **References** dialog box, which opens when you right-click on a condition on the **Global Conditions** dialog box and select **References** from the shortcut menu, lists any applications or other global conditions that reference the selected global condition.

The **References** dialog box includes the following properties:

Table 7-86 • References Dialog Box

Property	Description
Referring Applications list	List of all applications that have been assigned a requirement that uses the selected global condition.
Referring Global Conditions list	List of all other global conditions which use the selected global condition.

SCCM Server Environment Dialog Box

The SCCM Server Environment dialog box, which is opened by clicking **App-V Virtual Environments** in the Application Manager ribbon and then selecting **SCCM Server Environment**, lists all existing defined SCCM server environments.

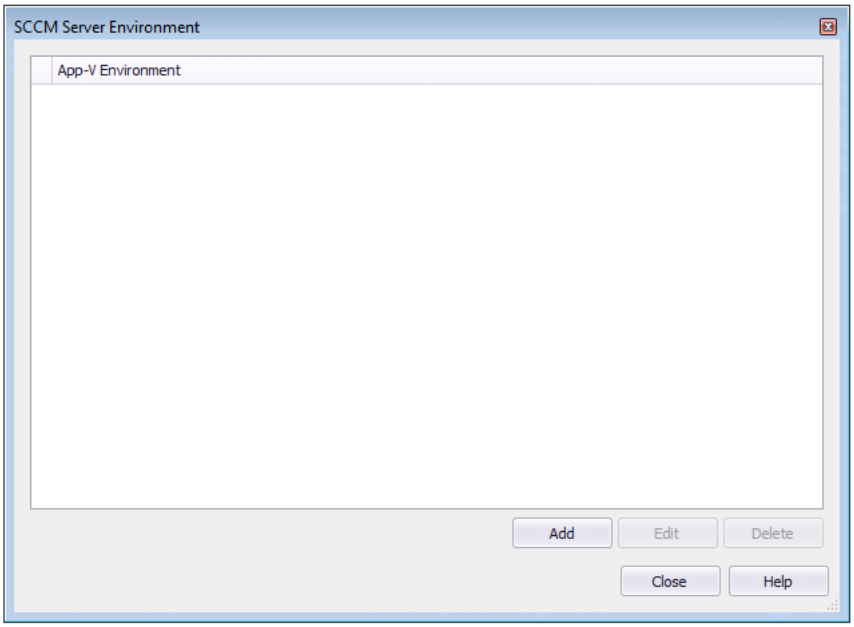


Figure 7-9: SCCM Server Environment Dialog Box

On the **SCCM Server Environment** dialog box, you can perform the following tasks:

- **Adding a new virtual environment**—Click **Add** to open the **Create Virtual Environment** dialog box and create a new virtual environment.
- **Editing an existing virtual environment**—Select a virtual environment and click **Edit** to edit an existing virtual environment.
- **Deleting a virtual environment**—Select a virtual environment and click **Delete** to delete an existing virtual environment.

Select Application Catalog Dialog Box

The **Select Application Catalog** dialog box opens when you are attempting to connect to an existing Standalone Application Catalog that requires database authentication by selecting it from a list of recently used Application Catalogs.

- **Enterprise Server**—Select this tab to open the AdminStudio Enterprise Server Application Catalog database. See Enterprise Server Tab.
- **Standalone**—Select this tab to open an Application Catalog other than the AdminStudio Enterprise Server database. See Standalone Tab / Specify Database Information.
- **Recent**—Provides a list of recently opened Application Catalogs. When you select an Application Catalog and click **OK**, either the Application Catalog opens or you are prompted for login information (if you need authentication to the Application Catalog). See Recent Tab.

Select AdminStudio Enterprise Server URL Dialog Box

If you click the HTTP link on an AdminStudio Enterprise Server Login dialog, this dialog box opens prompting you to identify the AdminStudio Enterprise Server URL that you would like to connect to.

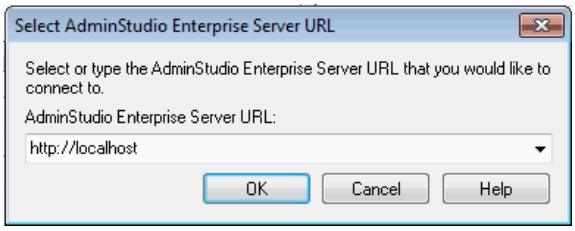


Figure 7-10: Select AdminStudio Enterprise Server URL Dialog Box

Select Substitute Package Dialog Box

The **Substitute Package** field on the **Casper Deployment Data > Limitations** subtab of the **Home Deployment Type View** specifies the package to deploy to computers that do not have the required architecture type.

If you click on the **Substitute Package** field (which, by default, is set to **None**), the **Select Substitute Package** dialog box opens, prompting you to select a substitute package from either the Casper Server or the Application Catalog.

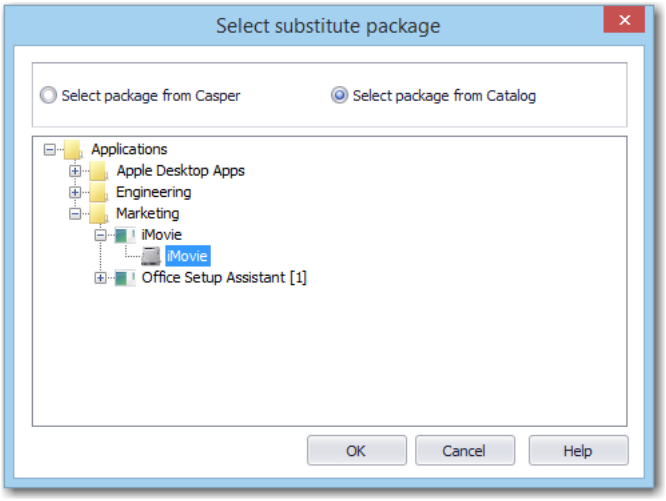


Figure 7-11: Select Substitute Package Dialog Box

Servers Dialog Box

The **Servers** dialog box opens when you click the browse link in the **Server names** field on the **XenApp Information** subtab of the **XenApp Deployment Data** tab of the **Home Deployment Type View** for a Citrix XenApp profile.

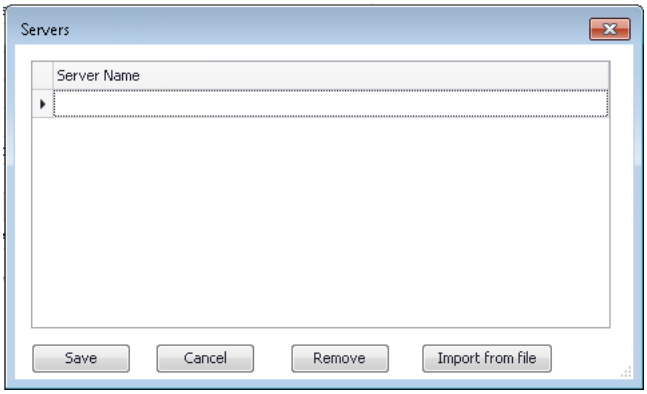


Figure 7-12: Servers Dialog Box

Enter server names in the list; click **Enter** to create a new row in the list. You also have the option to click **Import from file** to import a list of servers from an application server list file (`*.asl`).

Specify Applications Dialog Box

On the **Specify Applications** dialog box, which is opened when you click **Add** on the **Add Applications** dialog box, you can use a tree structure to select an App-V 5.0 application to add to an App-V 5.0 SCCM Server virtual environment.

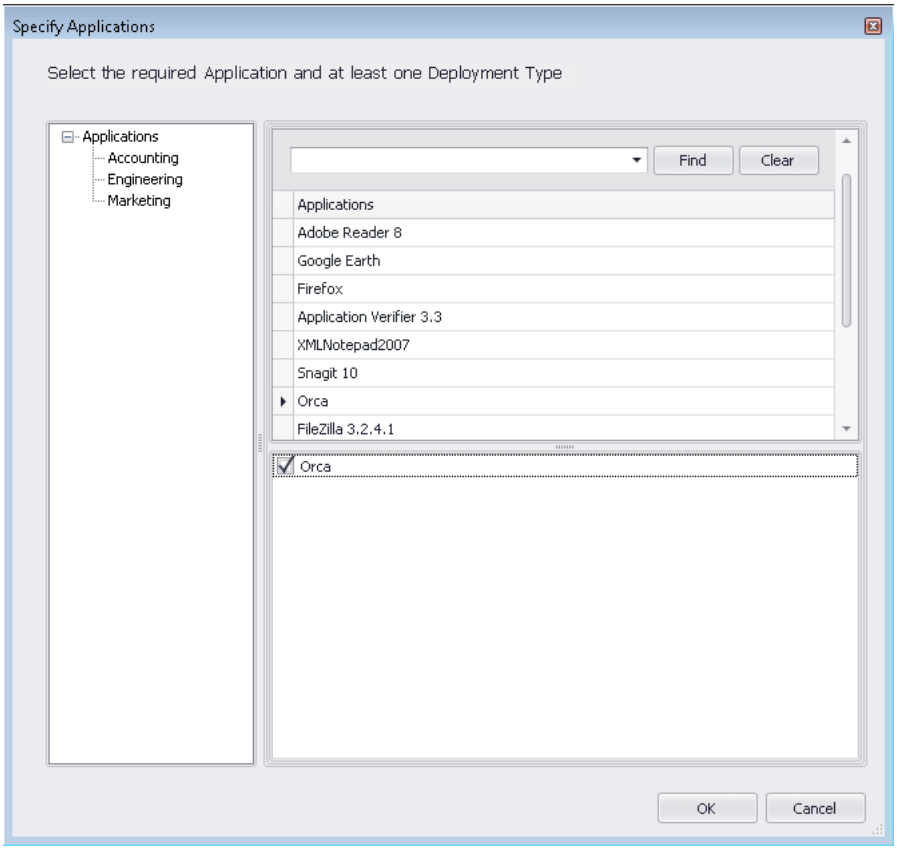


Figure 7-13: Specify Applications Dialog Box

The **Specify Applications** dialog box includes the following properties:

Table 7-87 • Specify Applications Dialog Box

Property	Description
Group tree list	Select the group that contains the App-V 5.0 package that you want to add.
Applications list	When a group is selected in the group tree list, the names of the applications in that group are listed in the Applications pane. Select the application in the list that contains the App-V 5.0 package that you want to add to the virtual environment.
Deployment type list	When an application is selected in the **Applications** list, that application's App-V 5.0 deployment type is listed in the lower pane. Select the App-V 5.0 deployment type that you want to add to the virtual environment and click OK. ***Note*** • *If the selected application does not contain any App-V 5.0 packages, nothing is listed in the Deployment type list.*

Users Dialog Box

The **Users** dialog box opens when you click the browse link in the **Accounts** field on the **XenApp Information** subtab of the **XenApp Deployment Data** tab of the **Home Deployment Type View** for a Citrix XenApp profile.

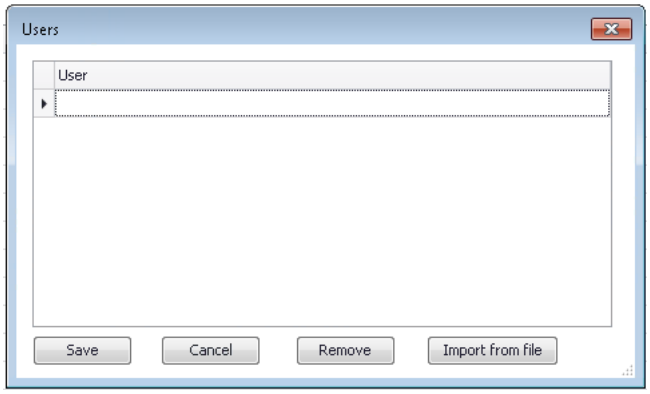


Figure 7-14: Users Dialog Box

Enter the user accounts that you want to have access to this XenApp profile to the list; click **Enter** to create a new row in the list. You also have the option to click **Import from file** to import a list of user accounts from an application user list file (`*.aul`).

Virtual Package Association Dialog Box

You can choose to use the **Associate Package** function in Application Manager to manually associate a virtual package with its source Windows Installer package after both packages have been imported into the Application Catalog.

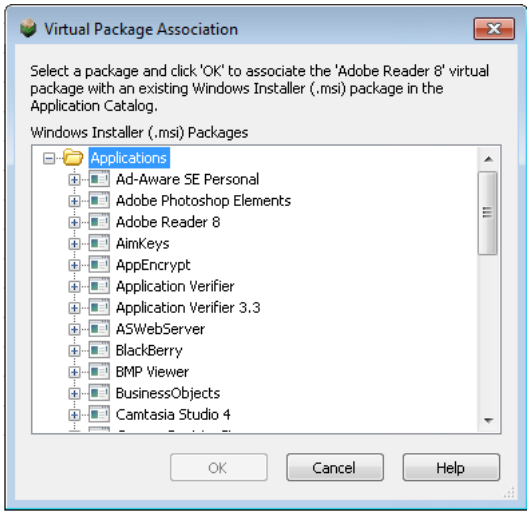


Figure 7-15: Virtual Package Association Dialog Box

If you right-click on a virtual package in the Application Catalog tree and then select **Associate Package** from the shortcut menu, the **Virtual Package Association** dialog box opens, listing all of the Windows Installer packages in the Application Catalog. Select the virtual package's source Windows Installer package and click **OK**. The Windows Installer package will now be listed in the **Associations** field on the **Package Information** tab of the **Home Deployment Type View** for the virtual package.

Important • *After you have imported a virtual package into the Application Catalog, you can use the **Associate Package** function to associate it with any Windows Installer package in the Application Catalog, even one that is not its source package. Therefore, it is preferable to use the Import Wizard to import both the Windows Installer and virtual packages at the same time so that AdminStudio can create the proper associations.*

XML Namespaces Dialog Box

On the **XML Namespaces** dialog box, which is opened by clicking **Namespace** on the Create Global Condition Dialog Box when defining a condition with the **Setting Type** of **XPath query**, you specify the XML namespaces and prefixes that you want to use when this XPath query runs.

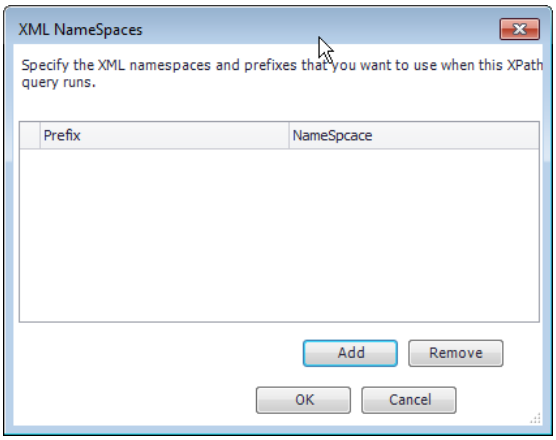


Figure 7-16: XML Namespaces Dialog Box

The **XML Namespaces** dialog box includes the following properties:

Table 7-88 • XML Namespaces Dialog Box Properties

Property	Description
Prefix	Lists the prefix portion of the defined XML namespaces.
Namespace	Lists the namespace portion of the defined XML namespaces.
Add	Click to open the **Add XML Namespace** dialog box, where you are prompted to enter a **Prefix** and **Namespace** for an XML namespace.
Remove	Click to remove a namespace from the list.

Wizards

Application Manager includes the following wizards:

- Application Catalog Wizard
- Conversion Wizard
- Customization Wizard
- Detection Method Wizard
- Detection Rule Wizard of Intune Deployment Data Tab
- Dependency Wizard
- Import Wizard
- Merge Module Import Wizard
- OS Snapshot Wizard
- Wrap Package Wizard
- Requirement Wizard
- Requirement Wizard of Intune Deployment Data Tab
- Return Code Wizard of Intune Deployment Data Tab
- Supersedence Wizard
- Test on Virtual Machine Wizard
- Upgrade Wizard

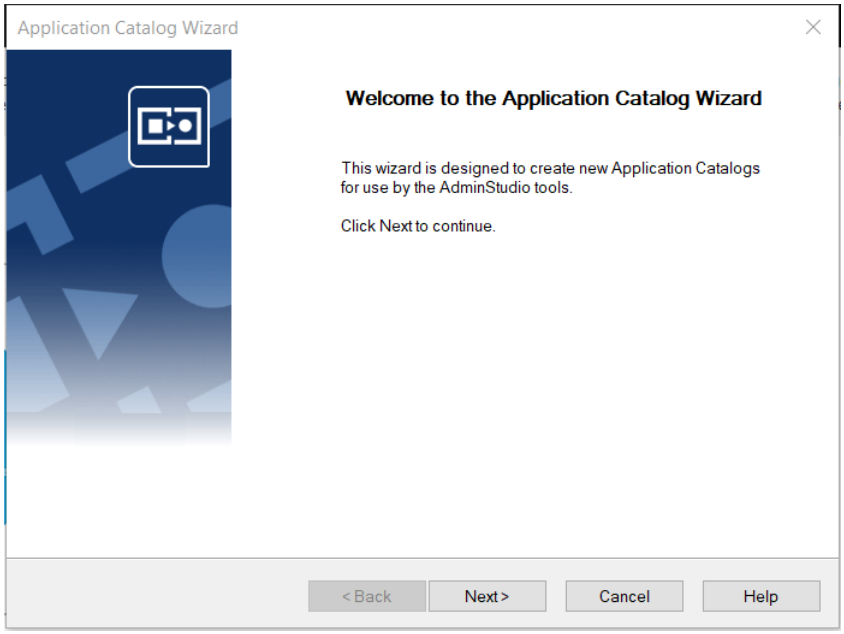
Application Catalog Wizard

You use the Application Catalog Wizard to create a new SQL Server Application Catalog database. This Wizard includes the following panels:

- Welcome Panel
- Specify Database Information Panel
- Select Software Repository Location Panel
- Creating Application Catalog Panel

Welcome Panel

You use the Application Catalog Wizard to create a new SQL / Azure Server Application Catalog database. On the **Welcome** panel, click **Next** to continue.



Specify Database Information Panel

On the **Specify Database Information** panel of the Application Catalog Wizard and the **Standalone** tab of the Connect Application Catalog Dialog Box, you have the following two options:

- Azure SQL
- On-Prem SQL

Azure SQL

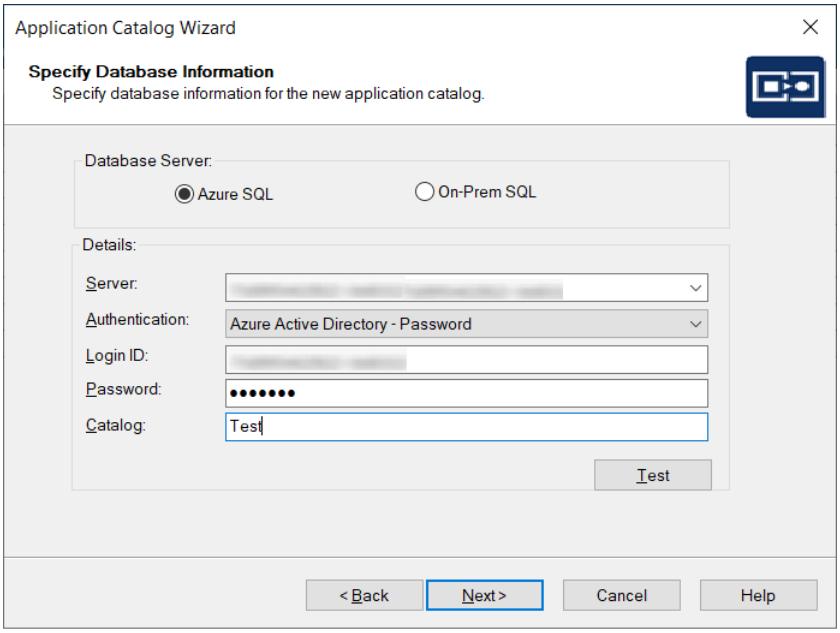
Enter the information required to login to the specified Application Catalog.

Table 7-89 • Application Catalog Wizard / Specify Database Information Panel / Azure SQL Option

Option	Description
Server	Select one of the available Azure SQL Servers on the network from this list. You can also manually enter the name of the server to which you want to connect.
Authentication	Select one of the following options: • **Server Authentication**—Choose to use Azure SQL Server login identification for authentication. If you chose this option, enter the appropriate Login ID and Password. • **Azure Active Directory - Password—**Choose to use Azure Active Directory login identification for authentication. If you chose this option, enter the appropriate Login ID and Password.
Catalog	Select the catalog from those available on the Server.

Table 7-89 • Application Catalog Wizard / Specify Database Information Panel / Azure SQL Option (cont.)

Option	Description
Test	Click this button to test whether a connection can be made to the database.



On-Prem SQL

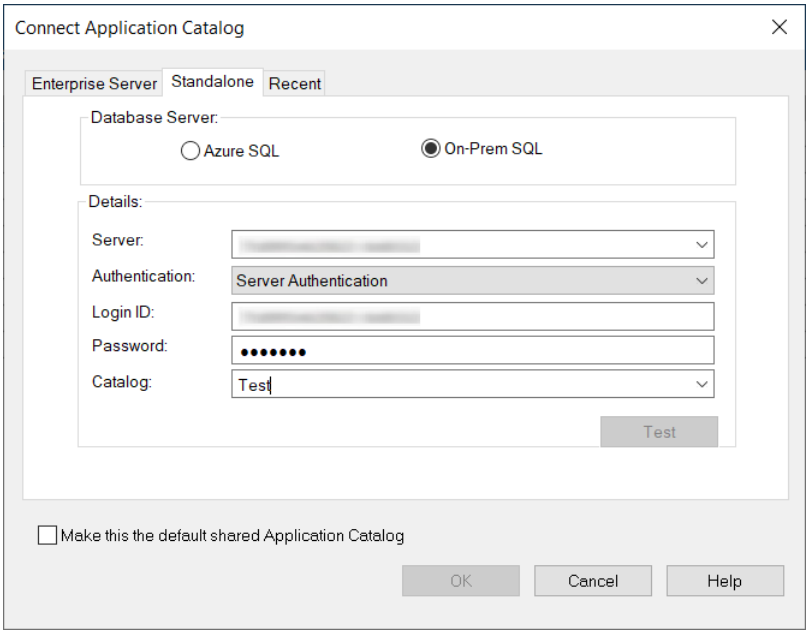
Enter the information required to login to the specified Application Catalog.

Table 7-90 • Application Catalog Wizard / Specify Database Information Panel Options

Option	Description
Server	Select one of the available SQL Servers on the network from this list. You can also manually enter the name of the SQL Server to which you want to connect.
Authentication	Select one of the following options: • **Windows Authentication**—Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Server Authentication**—Choose to use SQL Server login identification for authentication. • **Login ID and Password**—If you chose Server Authentication, enter the appropriate Login ID and Password.
Catalog	Select the catalog from those available on the Server.
Test	Click this button to test whether a connection can be made to the database.

Table 7-90 • Application Catalog Wizard / Specify Database Information Panel Options (cont.)

Option	Description
Make this the default shared Application Catalog	When this option is selected, the Application Catalog you are trying to open or create will become the default Application Catalog (and be recorded as such in the AdminStudio Shared directory).



Select Software Repository Location Panel

A Windows Installer package is made up of many files that are executed when the setup is run. You only import the `.msi` file into the Application Catalog, not all of the files necessary for installation. With the Software Repository, when you import an installation package into the Application Catalog, all of the files associated with that package are copied into the Software Repository location, a directory that you specify. This allows you to manage those files, preventing them from getting modified or lost.

On the **Select Software Repository Location** panel, you can choose to **Enable the Software Repository** for the new Application Catalog, and specify a **Proxy Account** for AdminStudio to use to make modifications to the directory path selected as the **Software Repository Location**.

Important • *You are only permitted to enable the Software Repository when creating an Application Catalog, not after it has already been created.*

The **Select Software Repository Location** panel includes the following options:

Table 7-91 • Select Software Repository Location Panel Options

Option	Description
Enable Software Repository	Select this option to enable the Software Repository feature for this new Application Catalog.
Software Repository Location	Enter or select the directory location of the Software Repository for this Application Catalog. All of the files associated with a package that is imported into the Application Catalog are copied to this location. This allows you to manage those files, preventing them from getting modified or lost.
Proxy Account	Specify the **Login ID** and **Password** for a **Proxy Account** that AdminStudio can use to access and modify the specified **Software Repository Location** folder. ***Note*** • *You cannot use Windows Authentication for this Proxy Account.* ***Important*** • *The Proxy Account needs full control on the **Software Repository Location** folder at the directory level as well as at the sharing level. Only such accounts can be used as a Proxy Account to access the **Software Repository Location** directory.*

Creating Application Catalog Panel

This panel displays the progress while your new Application Catalog is being created. If the Application Catalog cannot be created, an error message will be displayed.

Conversion Wizard

You can use the Application Manager **Conversion Wizard** to perform the following tasks from within Application Manager:

- **Convert an App-V 4.x package to App-V 5.0 format**—See Converting App-V 4.x Packages to App-V 5.0 Format.

  Important • *If AdminStudio is installed on a Windows 7 (x64) machine, you will need to first set the PowerShell execution policy to "unrestricted" before attempting to use the Conversion Wizard to upgrade an App-V 4.x package to App-V 5.0 format. To do this, execute the following command on an elevated Windows PowerShell (x86) utility:*

  ```
  Set-ExecutionPolicy Unrestricted
  ```

***Note** • To perform this upgrade, the Microsoft Application Virtualization Sequencer Version 5.0 must be installed on the same machine as AdminStudio.*

- **Convert one or multiple Windows Installer packages or legacy installers to virtual packages** using default Automated Application Converter settings— See Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging.
- **Convert MSI/EXE packages to Microsoft Intune format**—You can use the Conversion Wizard to convert packages to Microsoft Intune format. For more information, see Converting MSI/EXE Packages to Intunewin Format Using the Conversion Wizard

You open the Conversion Wizard by right-clicking on an application or group of applications in the Application Manager tree and then selecting **Launch Conversion Wizard** from the shortcut menu.

The Conversion Wizard consists of the following panels:

- Target Type Selection Panel
- Package Output Location Panel
- Select the Package(s) to Convert Panel
- Summary Panel
- Converting the Packages Panel

***Note** • Repackager, Automated Application Converter (AAC) and the Conversion Wizard in Application Manager now support Windows Services for MSIX Packages. A Windows Service installed while converting a legacy package format (MSI/EXE) will be captured and packaged into the MSIX package upon conversion.*

Target Type Selection Panel

On the **Target Type Selection** panel of the Conversion Wizard, you specify the type of conversion that you want to perform.

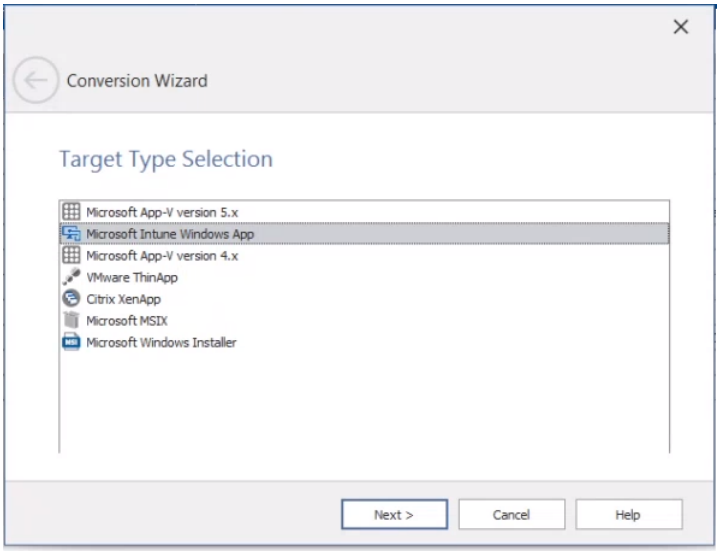


Figure 7-17: Conversion Wizard / Target Type Selection Panel

The following conversion types are available:

Table 7-92 • Conversion Wizard / Target Type Selection Panel

Option	Description
Microsoft App-V version 5.x	One of the following: • Convert the selected Microsoft App-V 4.x virtual package to App-V 5.x format. • Convert the selected Windows Installer or legacy package to a virtual package in Microsoft App-V 5.x format.
Microsoft MSIX Package	Covert the selected msi or exe package to MSIX package.
Microsoft App-V version 4.x	Convert the selected Windows Installer or legacy package to a virtual package in Microsoft App-V 4.x format.
VMware ThinApp	Convert the selected Windows Installer or legacy package to a virtual package in VMware ThinApp format.
Citrix XenApp	Convert the selected Windows Installer or legacy package to a virtual package in Citrix XenApp format.
Microsoft Windows Installer	Convert the selected legacy package to a Windows Installer package.
Microsoft Intune	Convert the selected MSI /EXE package to Microsoft Intune format.

Package Output Location Panel

On the **Package Output Location** panel, which opens when you are performing a conversion to Microsoft Intune format, you are prompted to specify an output location where you want to save the converted Intune package.

If you want to select different output location, click **Browse** and select preferred location (if required), and then click **Next**.

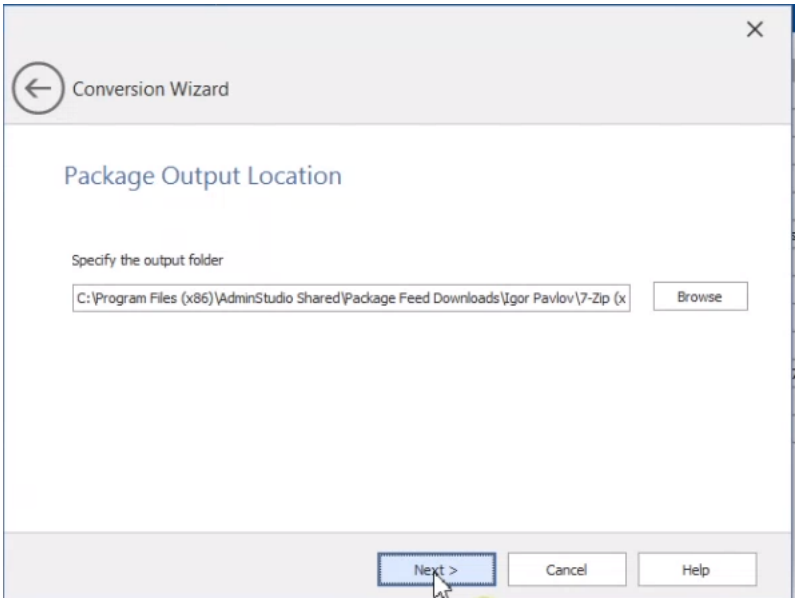


Figure 7-18: Conversion Wizard / Package Output Location Panel

Select the Package(s) to Convert Panel

When using the Conversion Wizard to perform a conversion to a virtual package or to upgrade an App-V 4.x virtual package to App-V 5.0 format, the **Select the Package(s) to Convert** panel opens and prompts you to select the packages you want to convert. By default, the package(s) that were selected when you launched the Conversion Wizard are already selected.

Select or clear the selection of applications that you want to convert and then click **Next** to continue.

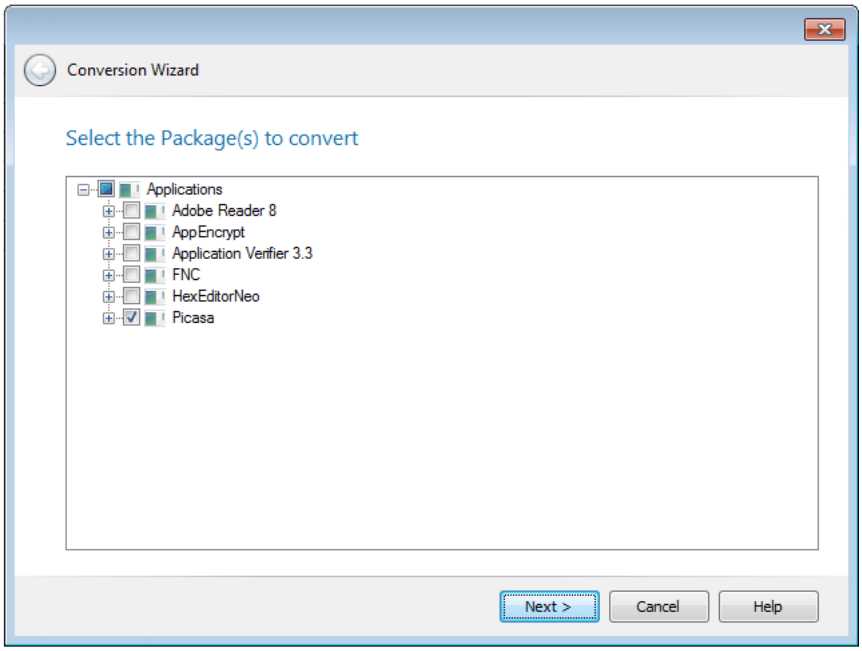


Figure 7-19: Conversion Wizard / Select the Package(s) to Convert Panel

Automated Application Converter Settings Panel

On the Automated Application Converter Settings panel of the Conversion Wizard you can specify the virtual machine platform to use during this conversion run, and can also choose to customize the default Automated Application Converter Settings for this run of the Conversion Wizard.

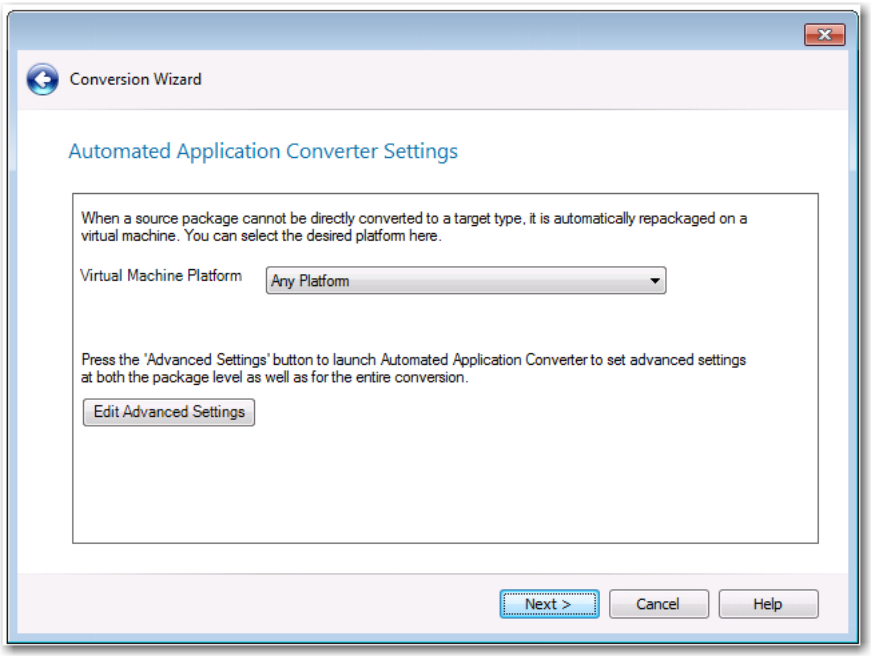


Figure 7-20: Automated Application Converter Settings Panel of Conversion Wizard

The virtual machine platforms defined in the settings file (that is specified on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box) are listed in the **Virtual Machine Platform** list. Select the platform to use for this run of the Conversion Wizard.

If you want to edit additional advanced settings, click the **Edit Advanced Settings** button. A copy of the default conversion settings file is opened, displaying the **Packages** tab of Automated Application Converter. For more information, see Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging.

Important • *Changes that you make to settings by clicking the **Edit Advanced Settings** button on this panel are only used for this run of the Conversion Wizard. To change the default settings, you need to edit the settings file that is specified on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box.*

Summary Panel

The **Summary** panel of the Conversion Wizard lists the selections you have made in the wizard. Click **Next** to begin conversion.

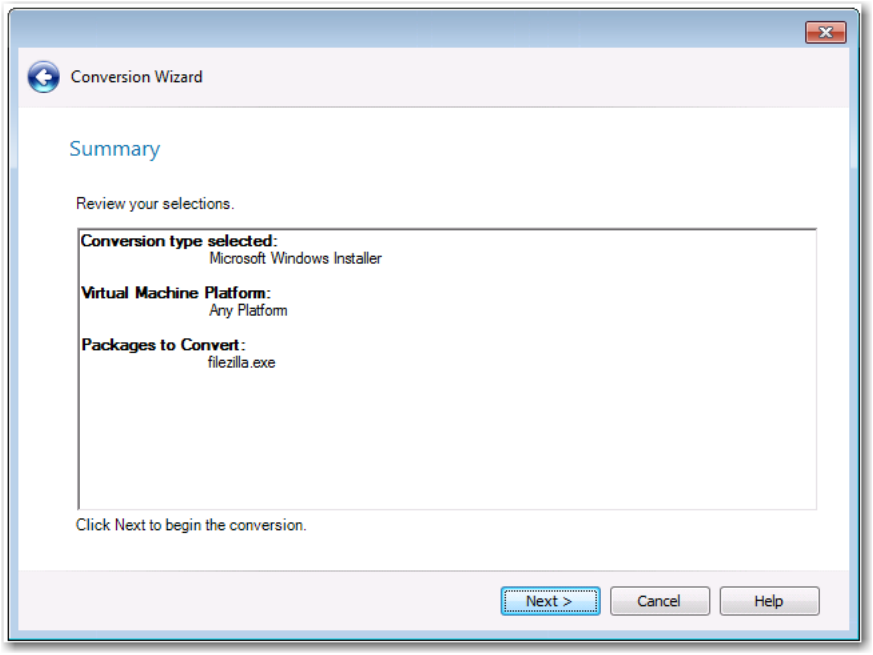


Figure 7-21: Conversion Wizard / Summary Panel

Converting the Packages Panel

During conversion using the Conversion, status messages are displayed on the Converting the Packages panel. When conversion is complete, the results of the conversion are listed.

Click **Finish** to close the wizard.

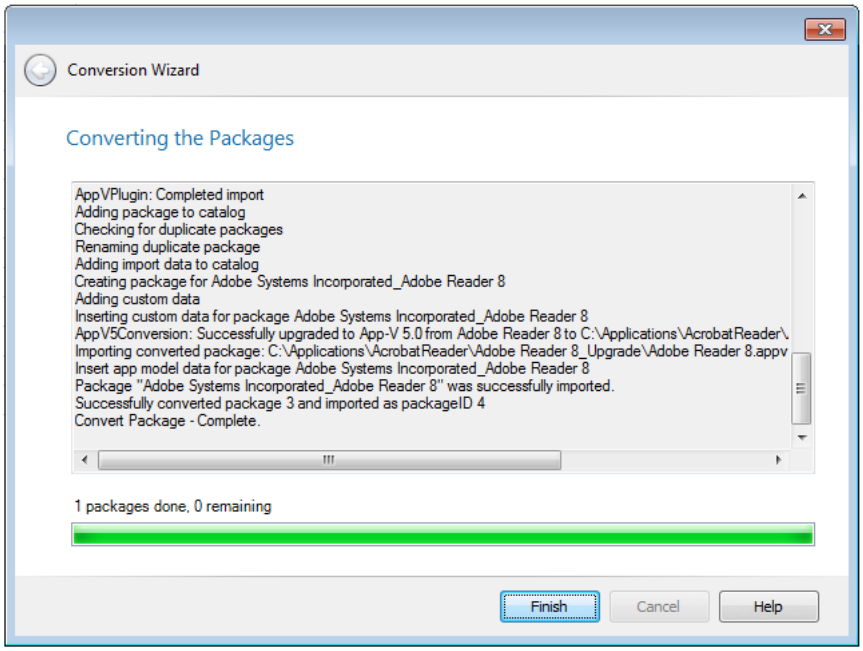


Figure 7-22: Conversion Wizard / Converting the Packages

Customization Wizard

Customization Wizard allows you to select from various customization options for a given package. The panels in the **Customization Wizard** are dynamic depending on the number of customization options available for the selected package. Options checked/unchecked in the **Customization Wizard** will be saved. When a new version of the package is executed manually or during automation, all the saved customization options are used to generate a transform (.mst) file and is imported along with the MSI package into the catalog.

You can launch the Customization Wizard from:

- The **Home** tab of Application Manager - Select an MSI package in the Applications tree and then click **Customize** in the ribbon button. You can also launch by selecting the **Customize** option from the context menu.
- The **Backlog** tab of Application Manager - Select a package request and then click **Customize** in the ribbon button. You can also launch by selecting the **Customize** option from the context menu.

Below is an example of various panels available in the **Customization Wizard** for Google Chrome. Please note that these panels and options are specific to Google Chrome, these will change depending on the customization options available for a package.

The Customization Wizard for Google Chrome consists of the following panels:

- Options Panel
- Homepage Preferences Panel
- Distribution Preferences Panel
- Add or Remove Programs Panel

- Release Notes Panel
- Saving Customization Panel

Options Panel

The following **Options** panel is a custom dynamic panel for the Google Chrome package.

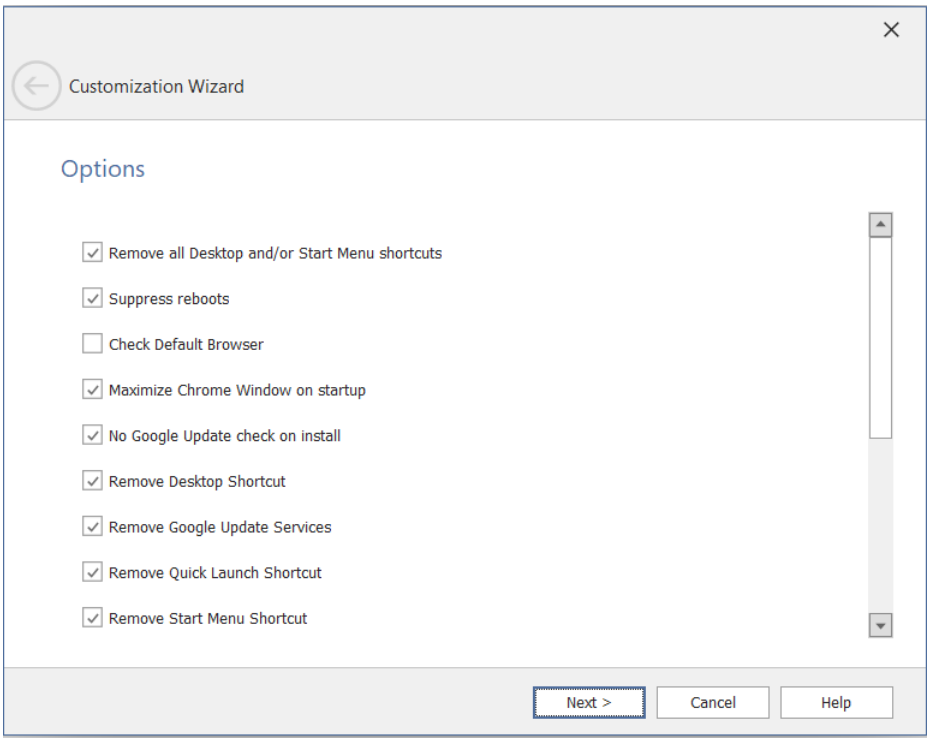


Figure 7-23: Customization Wizard / Options Panel

Homepage Preferences Panel

The following **HomePage Preferences** panel is a custom dynamic panel for the Google Chrome package.

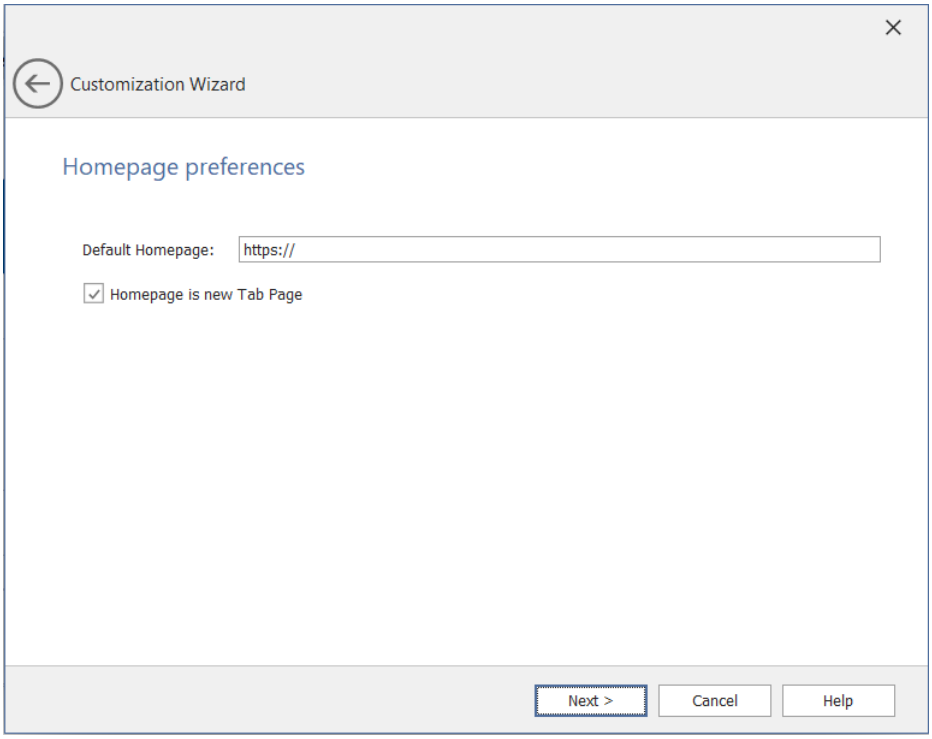


Figure 7-24: Customization Wizard / Homepage Preferences Panel

Distribution Preferences Panel

The following **Distribution Preferences** panel is a custom dynamic panel for the Google Chrome package.

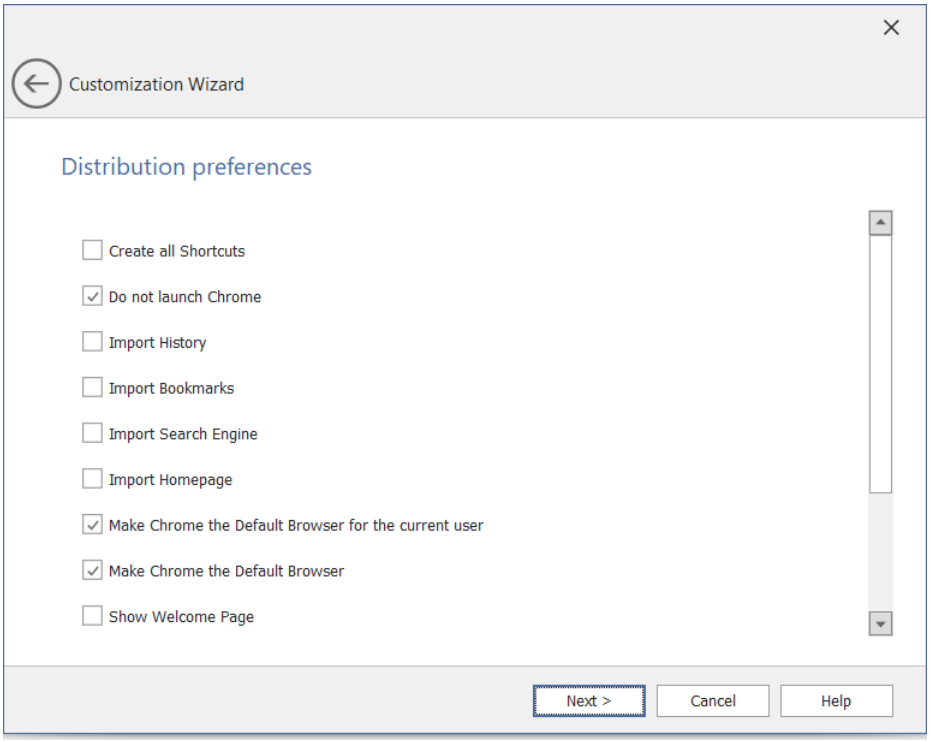


Figure 7-25: Customization Wizard / Distribution Preferences Panel

Add or Remove Programs Panel

The following **Add or remove Programs** panel is a custom dynamic panel for the Google Chrome package.

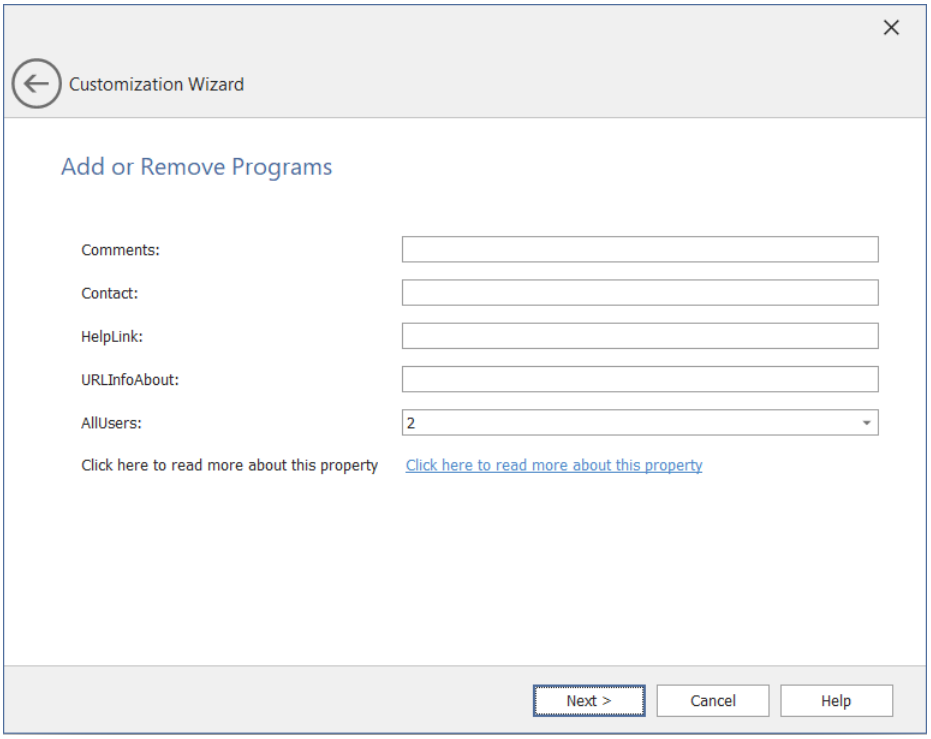


Figure 7-26: Customization Wizard / Add or remove Programs Panel

Release Notes Panel

The following **Release Notes** panel is a custom dynamic panel for the Google Chrome package.

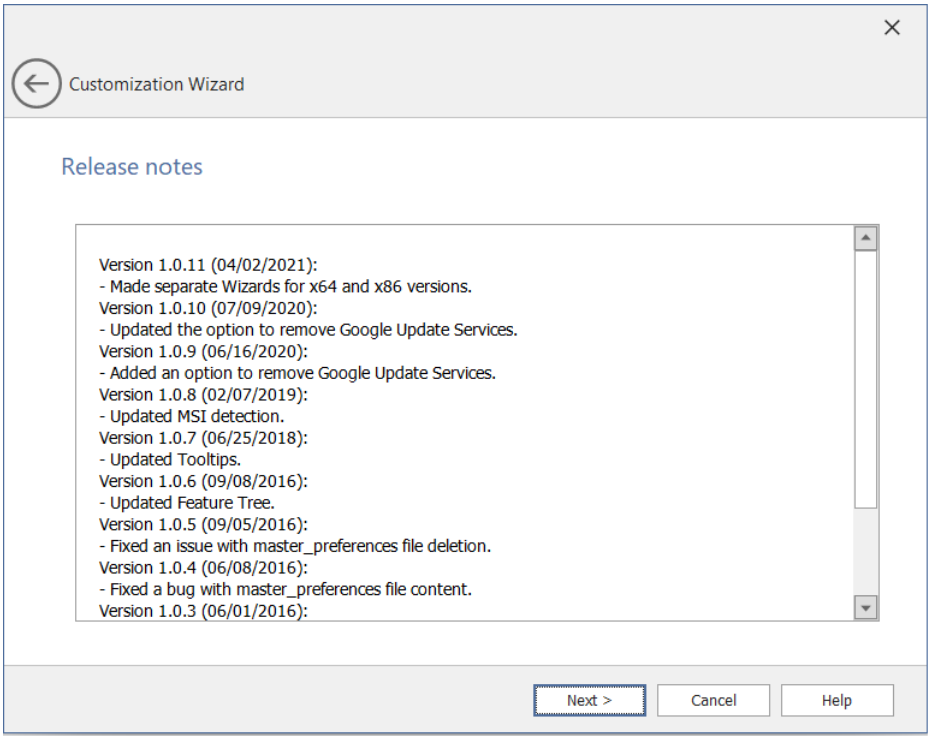


Figure 7-27: Customization Wizard / Release Notes Panel

Saving Customization Panel

The following **Saving Customization** panel is a custom dynamic panel for the Google Chrome package.

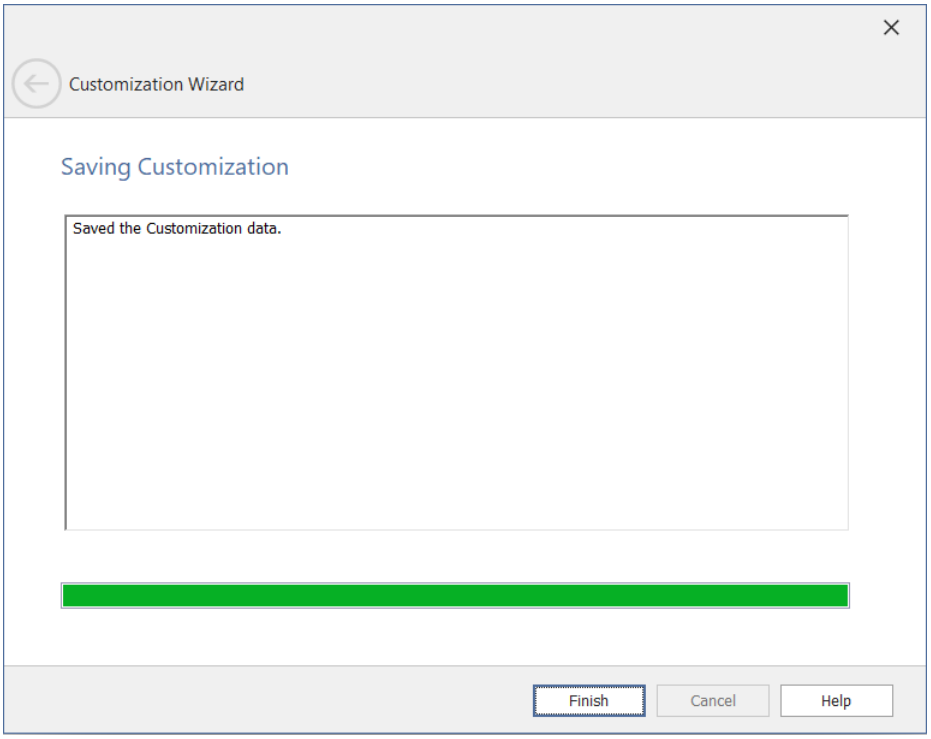


Figure 7-28: Customization Wizard / Saving Customization Panel

Detection Method Wizard

The **Detection Method** subtab of the **Deployment Data** tab of the **Home Deployment Type View** lists methods to detect whether this package is already installed on the target system. You can use the Detection Method Wizard to add detection methods to this list and to edit existing detection methods.

The **Detection Method Wizard** is opened by clicking the **Add Detection Method** or **Edit Detection Method** buttons in the ribbon of the **Detection Method** subtab.

The **Detection Method Wizard** consists of the following panels:

- Welcome Panel
- File System Detection Panel
- Registry Detection Panel
- Windows Installer Detection Panel
- Script Detection Panel
- Summary Panel

Welcome Panel

On the **Welcome** panel, you select the type of the detection method that you are adding: file system, registry, Windows Installer, or script.

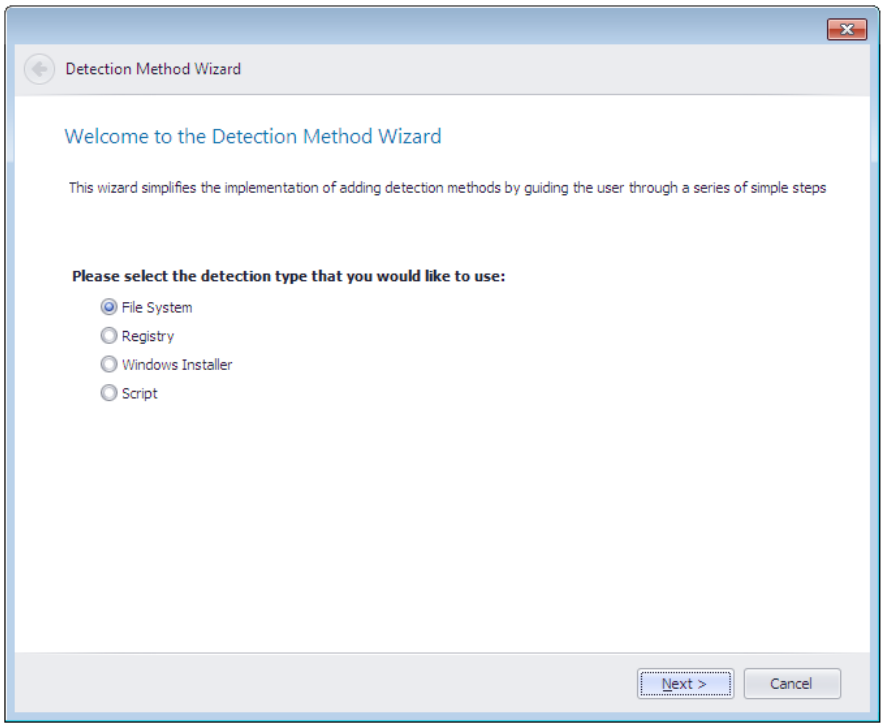


Figure 7-29: Detection Method Wizard / Welcome Panel

The **Welcome** panel includes the following properties:

Table 7-93 • Detection Method Wizard / Welcome Panel

Property	Description
File System	Determine whether an application is installed on a client device by detecting whether a specified file or folder exists on that client device.
Registry	Determine whether an application is installed on a client device by detecting whether a specified registry key or registry value exists on that client device.
Windows Installer	Determine whether an application is installed on a client device by detecting whether a specified Windows Installer file exists on that client device.
Script	Use a script to determine whether an application is installed on a client device.

File System Detection Panel

On the **File System Detection** panel, which opens if you selected **File System** on the **Welcome** panel, you enter the file or folder path information and the conditions to be applied on the selected file.

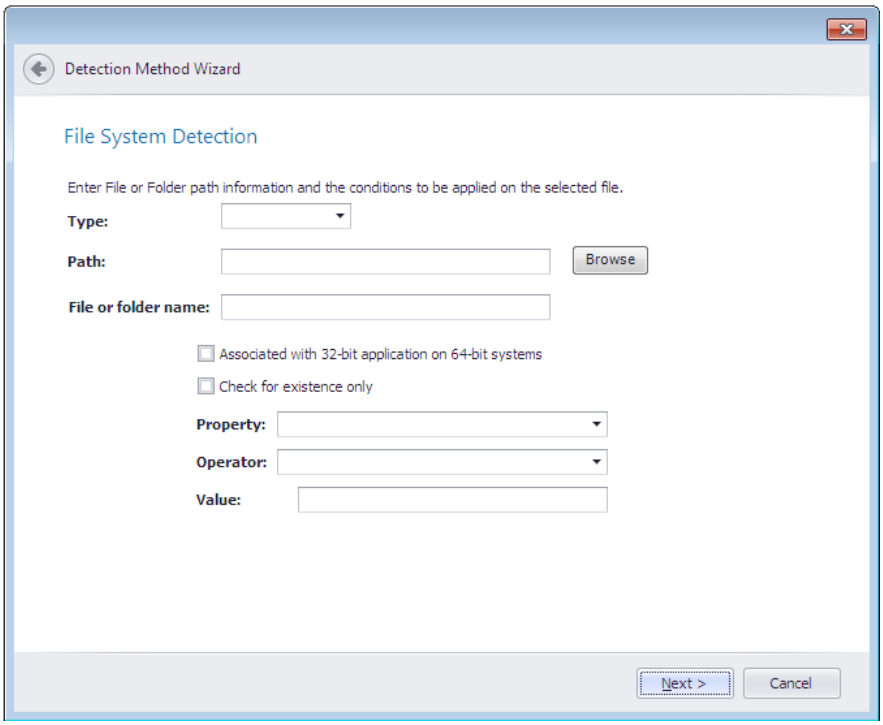


Figure 7-30: Detection Method Wizard / File System Detection Panel

The **File System Detection** panel includes the following properties:

Table 7-94 • Detection Method Wizard / File System Detection Panel

Property	Description
Type	Select either **File** or **Folder**.
Path	Click **Browse** and select the path the location of the file or folder.
File or folder name	Enter the name of the file or folder that you are using in this detection method.
Associated with 32-bit application on 64-bit systems	Select this option if you want to restrict this detection method to a file or folder that is associated with a 32-bit application on a 64-bit system.
Check for existence only	Select this option if you want to just check for the existence of the specified file or folder on the client system, without requiring that it meet any **Date Modified** or **Date Created** condition. If you select this option, the **Property**, **Operator**, and **Value** fields are disabled.

Table 7-94 • Detection Method Wizard / File System Detection Panel

Property	Description
Property **Operator** **Value**	Use these fields to define a condition, using the **Date Modified** or **Date Created** property, that the selected file or folder has to meet in order to be successfully detected. Available operators are: **Equals**, **Not equal to**, **Greater than or equal to**, **Greater than**, **Less than**, **Less than or equal to**, or **Between**. In the **Value** field, enter a date in the following format: 4/17/2012 4:47:50 PM

Registry Detection Panel

On the **Registry Detection** panel, which opens if you selected **Registry** on the **Welcome** panel, you enter the Windows Registry information and the conditions to be applied on those values.

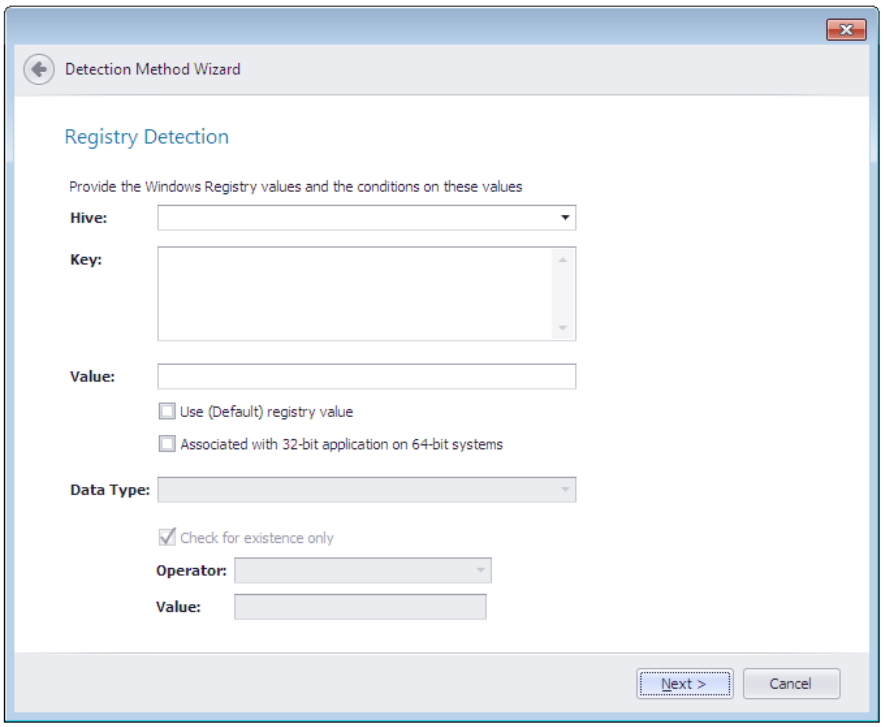


Figure 7-31: Detection Method Wizard / Registry Detection Panel

The **Registry Detection** panel includes the following properties:

Table 7-95 • Detection Method Wizard / Registry Detection Panel

Property	Description
Hive	Select the registry hive of the registry key or value that you are using in this detection method.
Key	Enter the registry key that you are using in this detection method.
Value	Enter the registry value that you are using in this detection method. This field is disabled if the **Use (Default) registry value** option is selected.
Use (Default) registry value	Select this option to search for a default registry value. If you select this option, the **Value** field is disabled and the **Data Type** field is enabled.
Associated with 32-bit application on 64-bit systems	Select this option if you want to restrict this detection method to a registry key or value that is associated with a 32-bit application on a 64-bit system.
Data Type	Select the data type of the default registry value that you are using in this detection method. Available data types are: **String**, **Integer**, or **Version**. This field is only enabled when the **Use (Default) registry value** option is selected.
Check for existence only	Select this option if you want to just check for the existence of the specified registry key or value on the client system, without requiring that it meet any conditions. If you select this option, the **Operator** and **Value** fields are disabled.
Operator **Value**	Use these fields to define a condition that the specified registry key or value has to meet in order to be successfully detected. Available operators are: **Equals**, **Not equal to**, **Greater than or equal to**, **Greater than**, **Less than**, **Less than or equal to**, or **Between**. In the **Value** field, enter a value for the registry key or value to define the condition.

Windows Installer Detection Panel

On the **Windows Installer Detection** panel, which opens if you selected **Windows Installer** on the **Welcome** panel, you provide the Windows Installer file information and the conditions that need to be applied on that file.

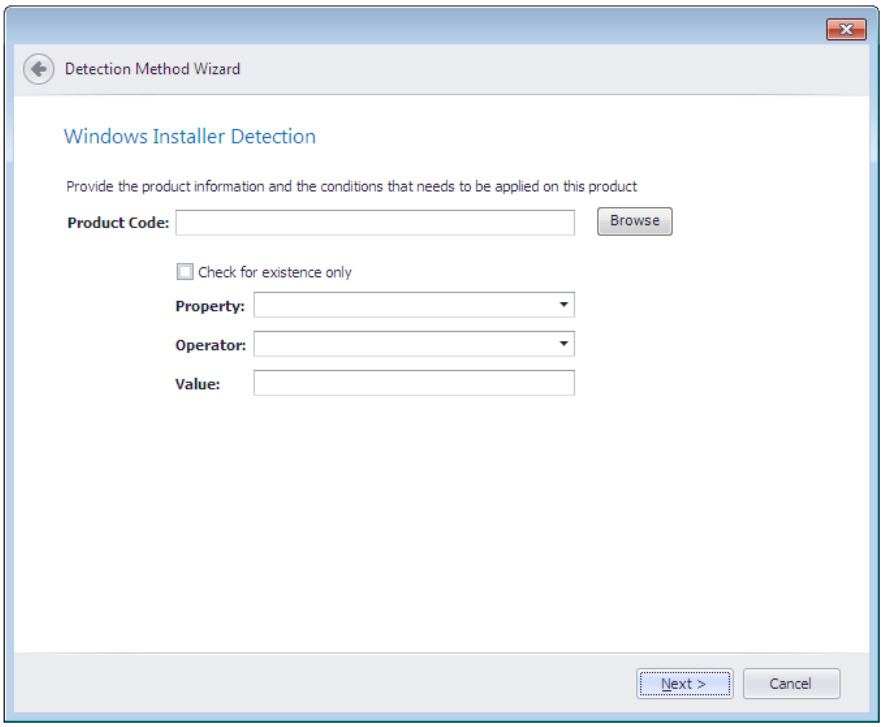


Figure 7-32: Detection Method Wizard / Windows Installer Detection Panel

The **Windows Installer Detection** panel includes the following properties:

Table 7-96 • Detection Method Wizard / Windows Installer Detection Panel

Property	Description
Product Code	Click **Browse** and select the Windows Installer `.msi` file that you want to use in this detection method. The Windows Installer file's **Product Code** will then be listed this field. ***Note*** *• You also have the option of entering the product code.*
Check for existence only	Select this option if you want to just check for the existence of the specified Windows Installer file on the client system, without requiring that it meet any conditions. If you select this option, the **Property**, **Operator** and **Value** fields are disabled.

Table 7-96 • Detection Method Wizard / Windows Installer Detection Panel

Property	Description
Property **Operator** **Value**	Use these fields to define a condition, using the **Version** or **Upgrade Code** property, that the specified Windows Installer file has to meet in order to be successfully detected. After you select a **Property**, the **Value** field is populated with a **Version** or **Upgrade Code** value from the selected Windows Installer file. Available operators are: **Equals**, **Not equal to**, **Greater than or equal to**, **Greater than**, **Less than**, or **Less than or equal to**. ***Note*** **•** *If you attempt to edit a Windows Installer Detection detection method, and you attempt to change the detection method property (from **Upgrade Code** to **Version** or vice versa), you may be required to required to browse to the Windows Installer file again to retrieve the new property value.*

Script Detection Panel

On the **Script Detection** panel, which opens if you selected **Script** on the **Welcome** panel, you specify the script information that you want to use to determine whether an application is installed on a client device.

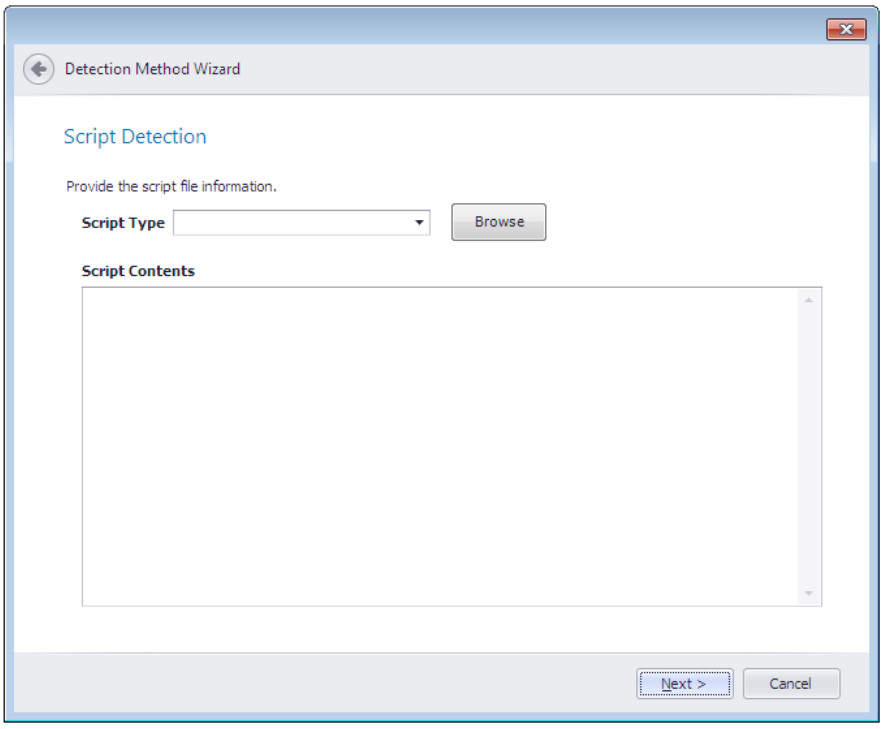


Figure 7-33: Detection Method Wizard / Script Detection Panel

Important • *If you already have some non-script rules defined for this deployment type, if you then add a script rule, the non-script rule(s) will not be pushed to ConfigMgr (Formerly called as System Center Configuration Manager) when you publish this application.*

The **Script** panel includes the following properties:

Table 7-97 • Detection Method Wizard / Script Panel

Property	Description
Script Type	First, select one of the following options: **PowerShell**, **VBScript**, or **JScript**. Next, click the **Browse** button and select the script that you want to use for this detection method. You also have the option of entering or pasting the script code directly into the **Script Contents** box.
Script Contents	Displays the contents of the specified script file. You are permitted edit the script in this text box.

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the Detection Method to the list.

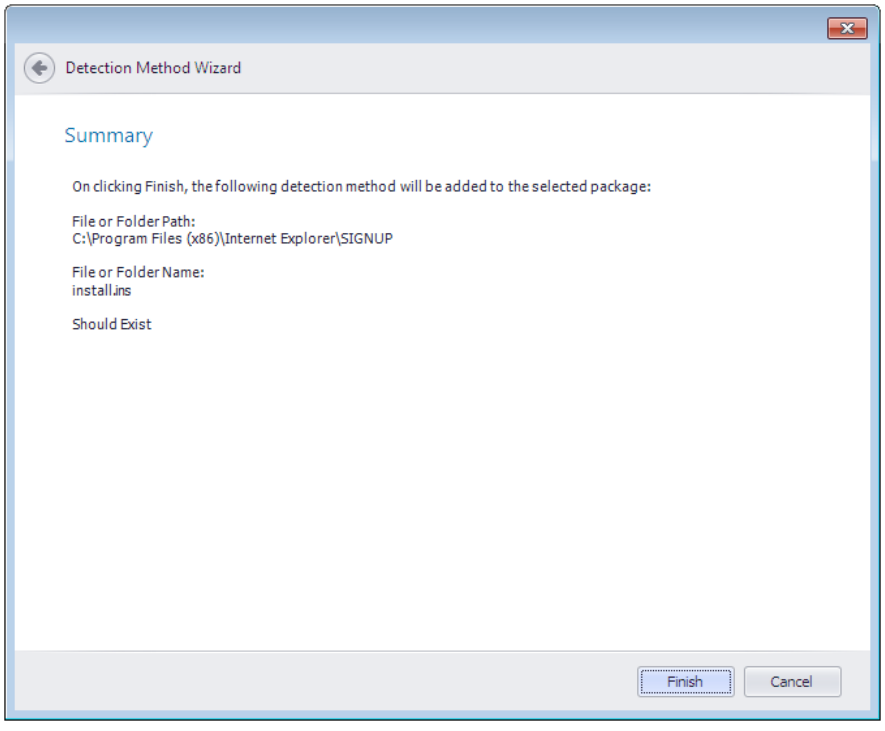


Figure 7-34: Detection Method Wizard / Summary Panel

Detection Rule Wizard of Intune Deployment Data Tab

The **Detection rules** subtab of the **Intune Deployment Data** tab of the **Home Deployment Type View** lists methods to detect whether this package is already installed on the target system. You can use the Detection rules Wizard to add detection rules to this list and to edit existing detection rules.

The Detection rules Wizard is opened by clicking the **Add Detection Method** or **Edit Detection Method** buttons in the ribbon of the **Detection rules** subtab.

The **Detection rules Wizard** consists of the following panels:

- Welcome Panel
- Configure Manual Detection Rule Panel
- Custom Detection Script Panel
- Summary Panel

Welcome Panel

On the **Welcome** panel, you select the type of the detection rule that you are adding.

- Manual Detection Rule
- Custom Detection Script

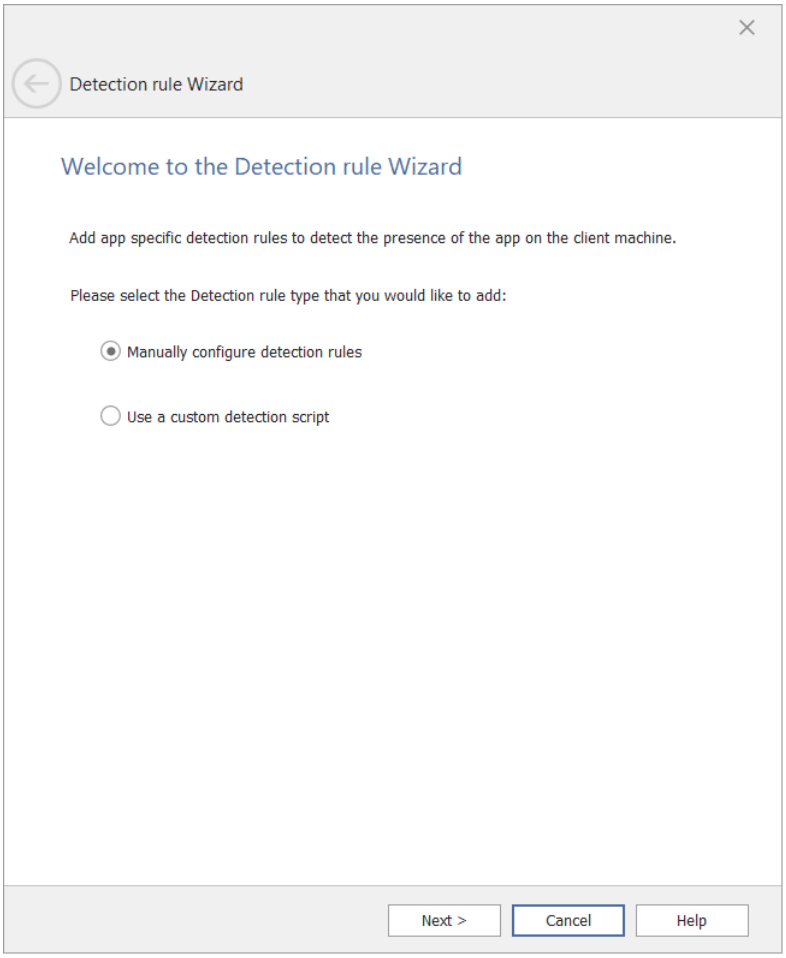


Figure 7-35: Detection Rule Wizard / Welcome Panel

The **Welcome** panel includes the following options:

Table 7-98 • Detection Rule Wizard / Welcome Panel

Option	Description
Manual configuration detection rules	Choose to manually configure the presence of the application.
Use a custom detection script	Use a script to determine whether an application is installed on a client device.

Configure Manual Detection Rule Panel

On the **Configure Manual Detection Rule** panel, select and specify the detection rule type.

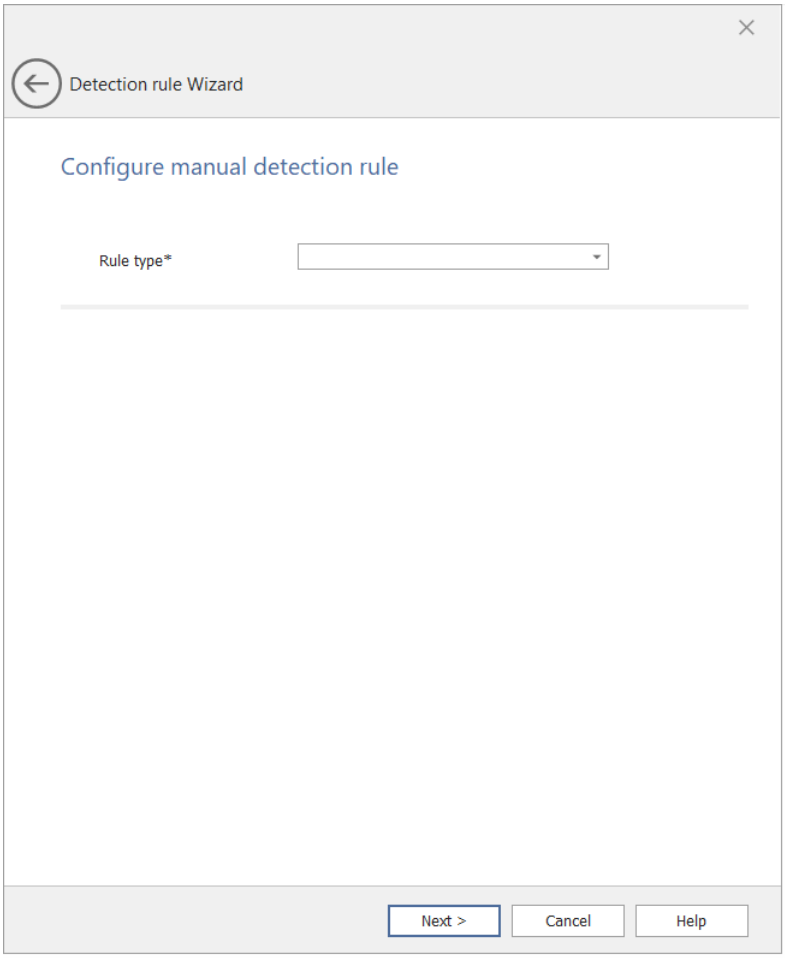


Figure 7-36: Detection Rule Wizard / Configure Manual Detection Rule Panel

The Configure Manual Detection Rule panel includes the following Properties:

Table 7-99 • Detection Rule Wizard / Configure Manual Detection Rule Panel

Properties	Description
Rule type	Select one of the following Requirement type: • MSI • File • Registry
MSI product code	Enter the valid Product code for the selected **MSI** rule type.
MSI product version check	Select this option if you want to verify the MSI product version in addition to the MSI product code. By default this option will be unselected.
Path	Enter the path of the selected **File**.

Table 7-99 • Detection Rule Wizard / Configure Manual Detection Rule Panel

Properties	Description
File or folder	Enter the name of the file or folder that you are using in this method.
Detection Method	Select one of the following method for the selected **File**: • File or folder exists • Date modified • Date created • String (version) • Size in MB
Key Path	Enter the path of the selected **Registry** type.
Value Name	Enter the register value in the text box.
Detection Method	Select one of the following method for the selected **Registry**: • Key exists • Key does not exits • String comparison • Version comparison • Integer comparison
Operator	Select one of the following operators: • Equals • Not equal to • Greater than or equal to • Greater than • Less than or equal to • Less than
Value	Enter the value to define the condition.

Table 7-99 • Detection Rule Wizard / Configure Manual Detection Rule Panel

Properties	Description
Associated with a 32-bit app on 64-bit clients	Select this option to expand any path environment variables in the 32-bit context on 64-bit clients. Unselect this option to expand any path variables in the 64-bit context on 64-bit clients. 32-bit clients will always use the 32-bit context. By default this option will be unselected. ***Note*** *• For the Registry Requirement Type:* • *Select this option to search the 32-bit registry on 64-bit clients.* • *Unselect this option to search the 64-bit registry on 64-bit clients. 32-bit clients will always search the 32-bit registry. By default this option will be unselected.*

Custom Detection Script Panel

On the Custom detection script panel, you select the script file.

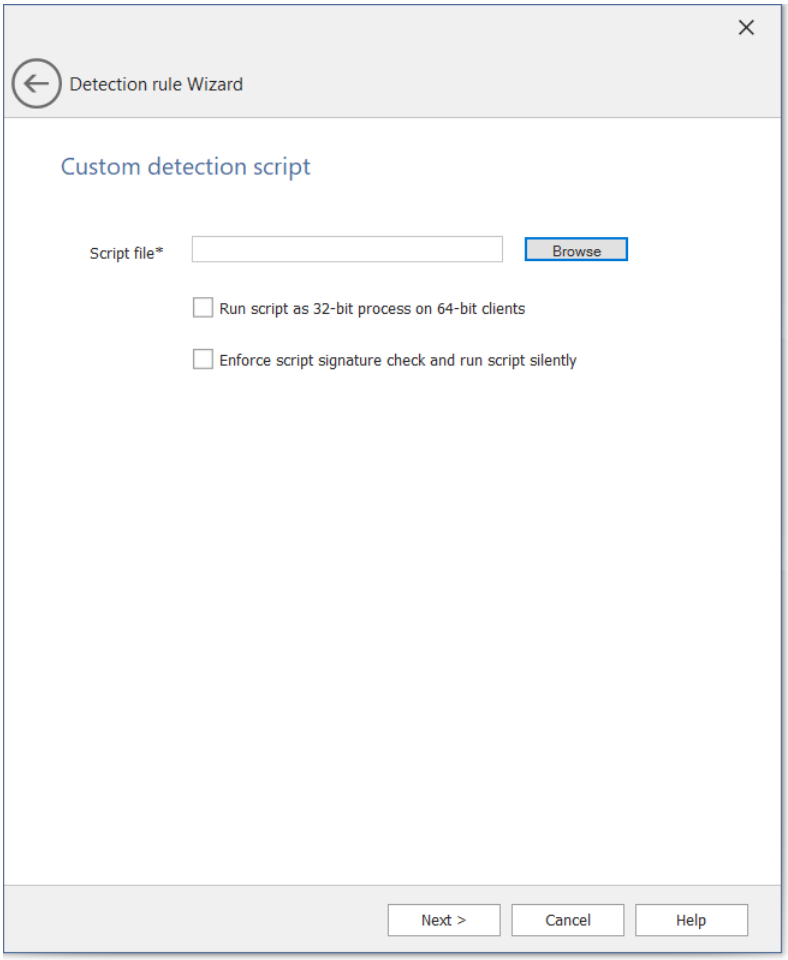


Figure 7-37: Detection Rule Wizard / Custom Detection Script Panel

The Custom detection script panel includes the following Properties:

Table 7-100 • Detection Rule Wizard / Custom Detection Script Panel

Properties	Description
Script file	Click the **Browse** button and select the script that you want to add for this rule. ***Note*** • *The application will be detected when the script both returns a **0** value exit code and writes a string value to STDOUT.*
Run script as 32-bit process on 64-bit clients	Select this option if you want to run the script in a 32-bit process on 64-bit clients. Unselect to run the script in a 64-bit process on 64-bit clients. 32-bit clients run the script in a 32-bit process. By default this option will be unselected.

Table 7-100 • Detection Rule Wizard / Custom Detection Script Panel

Properties	Description
Enforce script signature check and run script silently	Select this option if you want to verify that the script is signed by a trusted publisher, which will allow the script to run with no warnings or prompts displayed. The script will run unblocked. Unselect to run the script with end-user confirmation without signature verification. By default this option will be unselected.

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the Detection rule to the list.

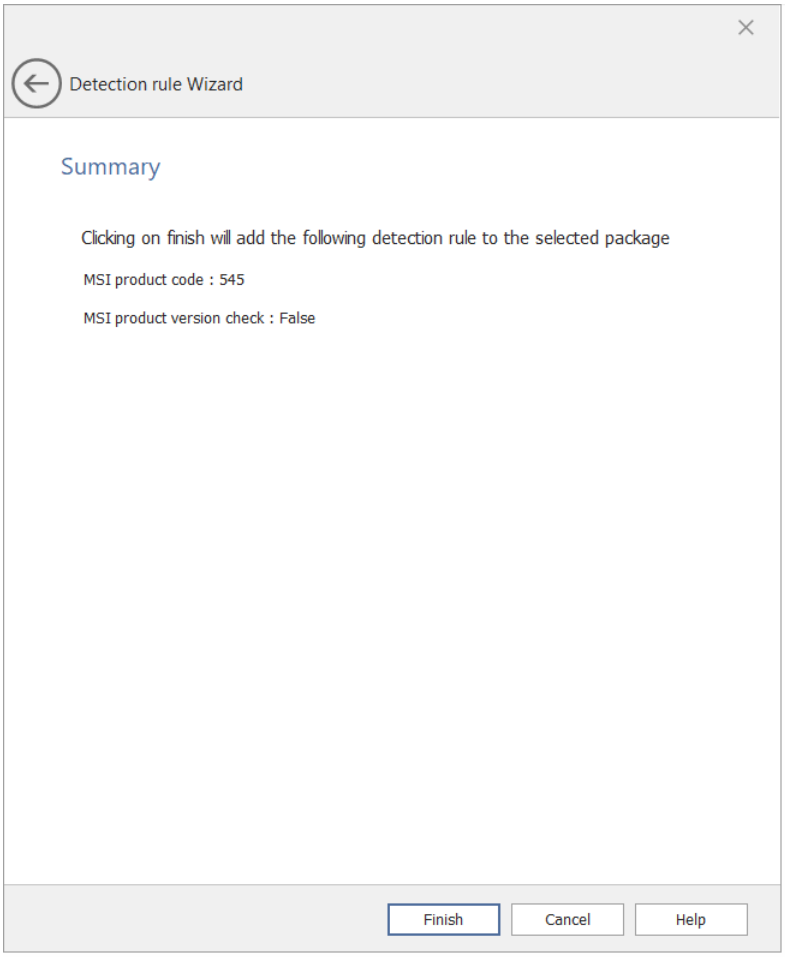


Figure 7-38: Detection Rule Wizard / Summary Panel

Dependency Wizard

You can use the **Dependencies** subtab to view or edit a list of other packages in the Application Catalog that must also be deployed by ConfigMgr (Formerly called as System Center Configuration Manager) with this package onto the target machine in order for this package to successfully operate. You can use the Dependency Wizard to add dependencies to this list and to edit existing dependencies.

The **Dependency Wizard** is opened by clicking the **Add Dependency** or **Edit Dependency** buttons in the ribbon of the **Dependencies** subtab.

The **Dependency Wizard** consists of the following panels:

- Welcome Panel
- Deployment Types in Application Catalog Panel
- Configuration Manager Credentials Panel
- Deployment Types in Configuration Manager 2012 Panel
- Auto Detect Dependencies Panel
- Scanning Progress Panel
- Auto Scan Results Panel
- System Requirements Panel
- Summary Panel

Welcome Panel

On the **Welcome** panel, choose the method that you would like to use to add dependencies:

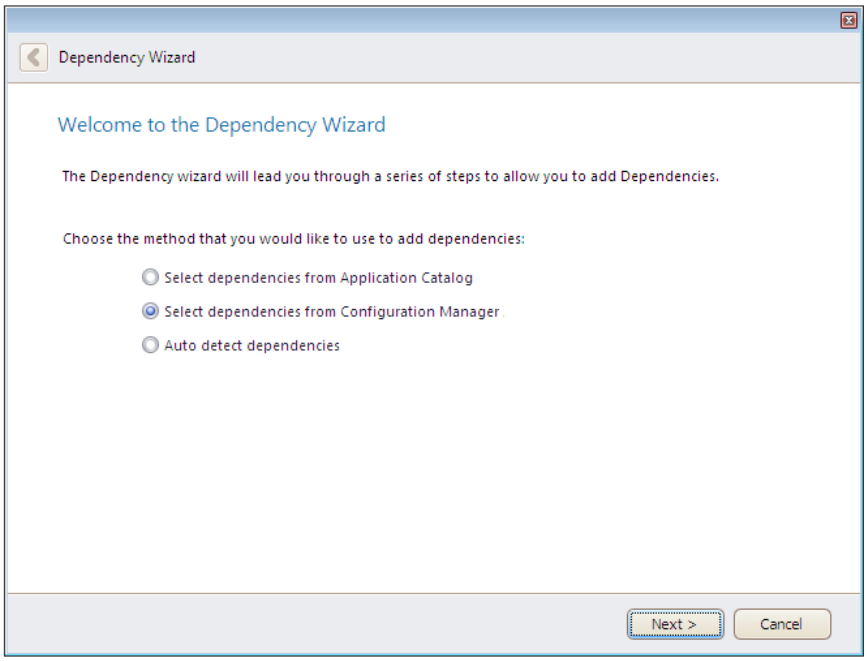


Figure 7-39: Dependency Wizard / Welcome Panel

The **Welcome** panel includes the following options:

Table 7-101 • Dependency Wizard / Welcome Panel

Option	Description
Select dependencies from Application Catalog	Select this option if you want to select a dependent application from those in the Application Catalog.
Select dependencies from Configuration Manager	Select this option if you want to select a dependent application from those in ConfigMgr (Formerly called as System Center Configuration Manager).
Auto-detect dependencies	Select this option if you want Application Manager to automatically detect dependent packages. To do this, Application Manager will automatically scan the file headers of Windows Installer packages to determine if any dependencies exist.

Deployment Types in Application Catalog Panel

On the **Deployment Types in Application Catalog** panel, you select dependent applications from the Application Catalog.

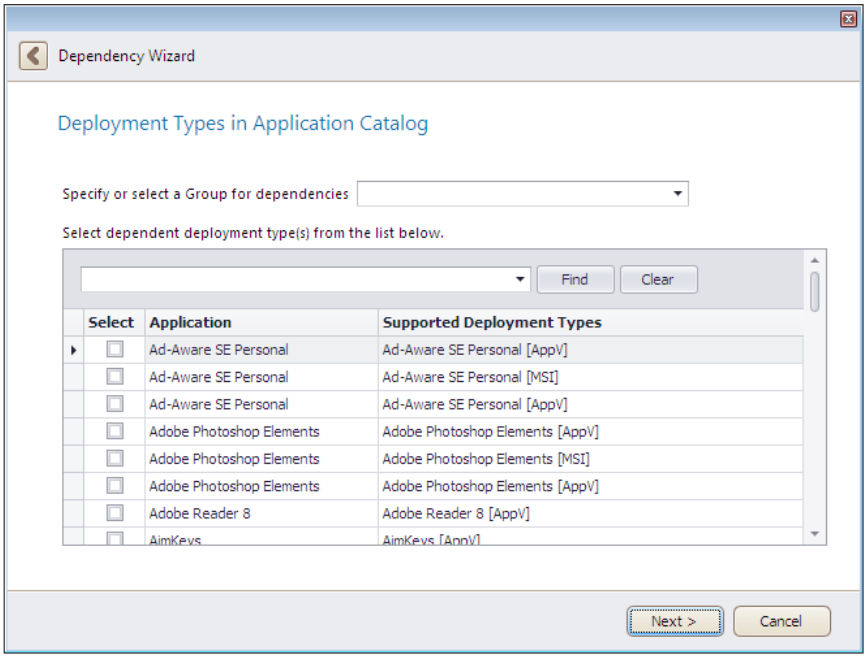


Figure 7-40: Dependency Wizard / Deployment Types in Application Catalog Panel

The Deployment Types in Application Catalog panel includes the following options:

Table 7-102 • Dependency Wizard / Deployment Types in Application Catalog Panel

Option	Description
Specify or select a Group for dependencies	Select a group name from the list or enter the name of a new group.

Table 7-102 • Dependency Wizard / Deployment Types in Application Catalog Panel

Option	Description
Search box	Use to filter the list of packages.
Application	Name of available applications.
Supported Deployment Types	Deployment type of available applications.

Configuration Manager Credentials Panel

On the **Configuration Manager Credentials** panel, you enter connection information for ConfigMgr (Formerly called as System Center Configuration Manager).

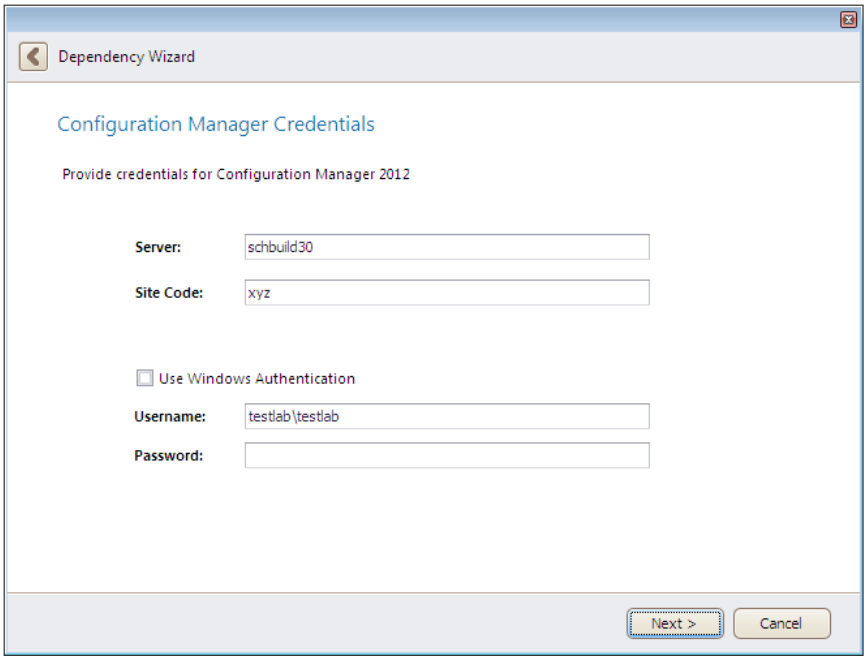


Figure 7-41: Dependency Wizard / Configuration Manager Credentials Panel

The **Configuration Manager Credentials** panel includes the following properties:

Table 7-103 • Dependency Wizard / Configuration Manager Credentials Panel

Property	Description
Server	Enter the name of the Configuration Manager Server you want to connect to.
Site Code	Enter the code that identifies the Configuration Manager site you want to connect to.
Use Windows Authentication	Select this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server.

Table 7-103 • Dependency Wizard / Configuration Manager Credentials Panel

Property	Description
Username and Password	If using server authentication, enter the **Username** and **Password** of that server.

Deployment Types in Configuration Manager 2012 Panel

On the **Deployment Types in Configuration Manager 2012** panel, you select dependent deployment types from the list.

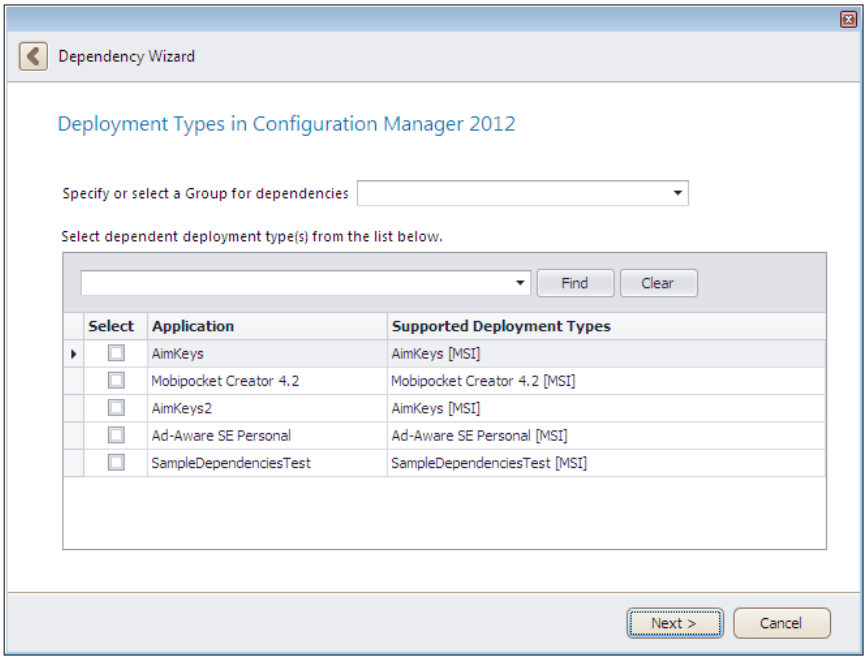


Figure 7-42: Dependency Wizard / Deployment Types in Configuration Manager 2012 Panel

The **Deployment Types in Configuration Manager 2012** panel includes the following properties:

Table 7-104 • Dependency Wizard / Deployment Types in Configuration Manager 2012 Panel

Property	Description
Specify or select a Group for dependencies	Either select an existing group from the list or enter the name for a new group.
Search box	Use to filter the list of packages.
Application	Name of available applications
Supported Deployment Types	Deployment types of available applications.

Auto Detect Dependencies Panel

The **Auto Detect Dependencies** panel opens when you select **Auto-detect dependencies** from the **Welcome** panel of the Dependency Wizard.

Click **Next** to begin scanning. Application Manager will begin to automatically scan the file headers of Windows Installer packages to determine if any dependencies exist.

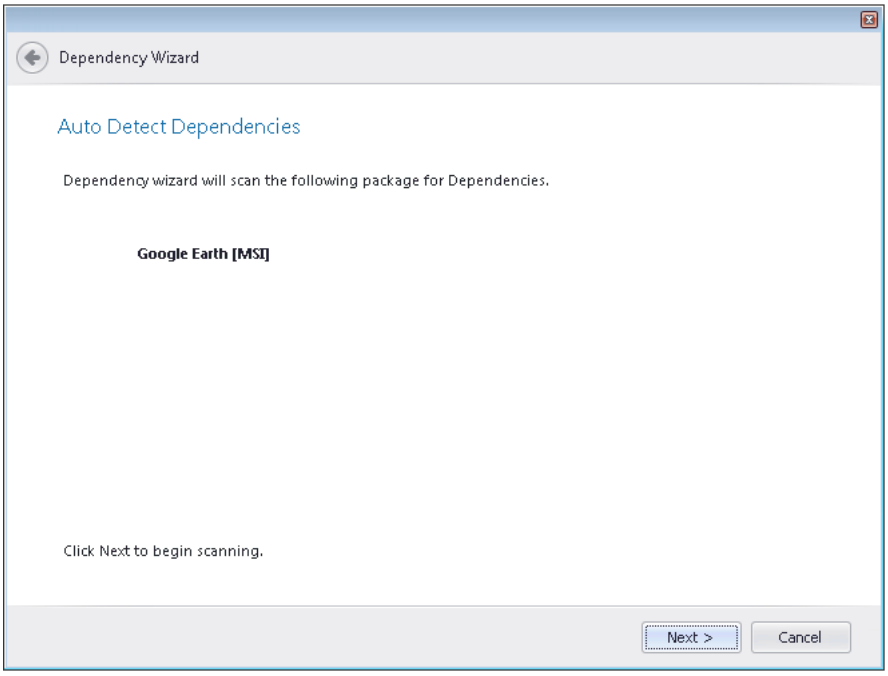


Figure 7-43: Dependency Wizard / Auto Detect Dependencies Panel

When you scan a Windows Installer package for dependencies, using the **Auto detect dependencies** option of the **Dependency Wizard**, two types of dependencies are detected:

- **Package-level dependencies**—The **Auto detect dependencies** option of the **Dependency Wizard** detects other packages that the current package is dependent upon. You can then choose to add them to the package's **Dependencies** subtab of the **Deployment Data** tab on the **Home Deployment Type View**.
- **File-level dependencies**—The **Auto detect dependencies** option of the **Dependency Wizard** populates the a Windows Installer package's file-level Dependencies View.

Scanning Progress Panel

The **Scanning Progress** panel opens when you begin a dependency scan and shows the progress. When scanning is complete, you are prompted to click **Next** to see the results of the dependency scan.

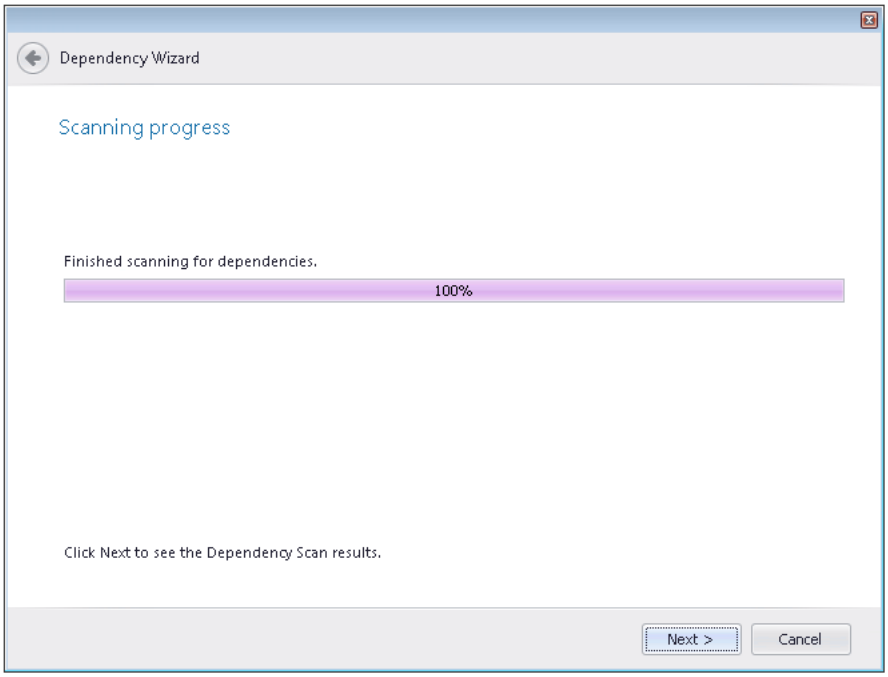


Figure 7-44: Dependency Wizard / Scanning Progress Panel

Auto Scan Results Panel

If the Dependency Wizard detected some dependencies during the dependency scan, the dependencies are listed.

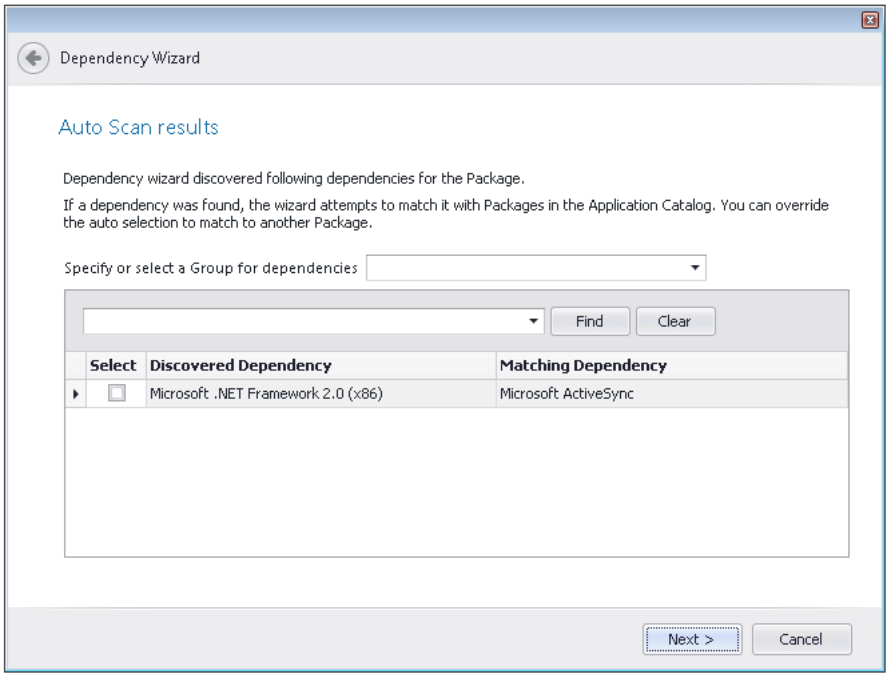


Figure 7-45: Dependency Wizard / Auto Scan Results Panel

The **Auto Scan Results** panel includes the following properties when dependencies are detected:

Table 7-105 • Auto Scan Results Panel (Dependencies Found) / Dependence Wizard

Property	Description
Specify or select a Group for dependencies	Either select an existing group from the list or enter the name for a new group.
Search box	Use to filter the list of packages.
Discovered Dependency	List of discovered dependencies. Select the dependencies you want to add to the **Dependencies** tab.
Matching Dependency	Identifies other packages that are also dependent upon the listed dependency.

Below is an example of the Auto Scan Results panel when no dependencies are found:

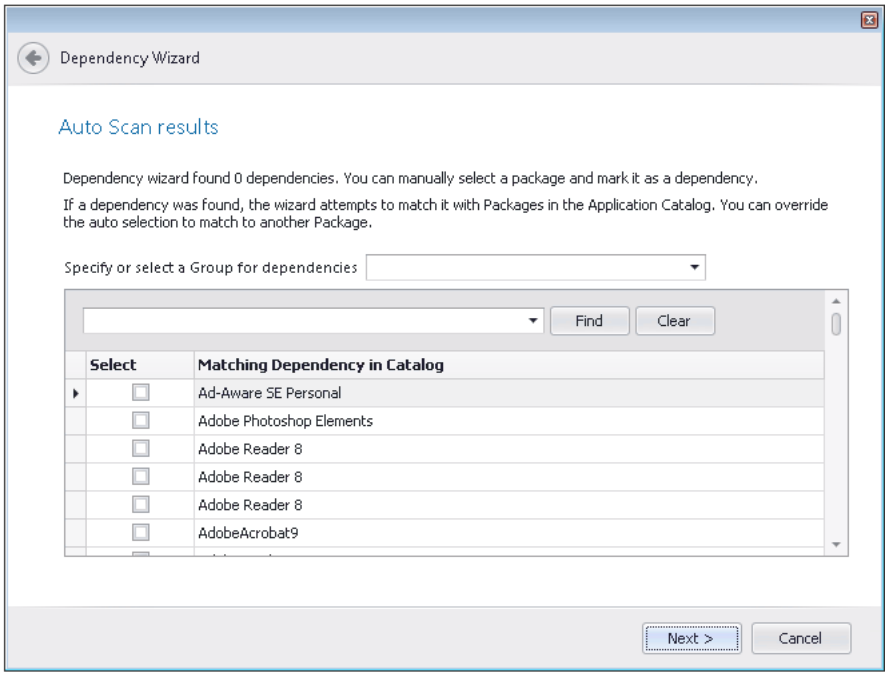


Figure 7-46: Dependency Wizard / Auto Scan Results Panel / No Dependencies Found

The **Auto Scan Results** panel includes the following properties when no dependencies are detected:

Table 7-106 • Auto Scan Results Panel (Dependencies Found) / Dependence Wizard

Property	Description
Specify or select a Group for dependencies	Either select an existing group from the list or enter the name for a new group.
Search box	Use to filter the list of packages.

Table 7-106 • Auto Scan Results Panel (Dependencies Found) / Dependence Wizard

Property	Description
Matching Dependency in Catalog	Select the packages that you want to add to the **Dependency** tab for the selected package.

System Requirements Panel

If you are performing a dependency scan, the **System Requirements** panel opens and lists any system requirements that were detected for the selected package.

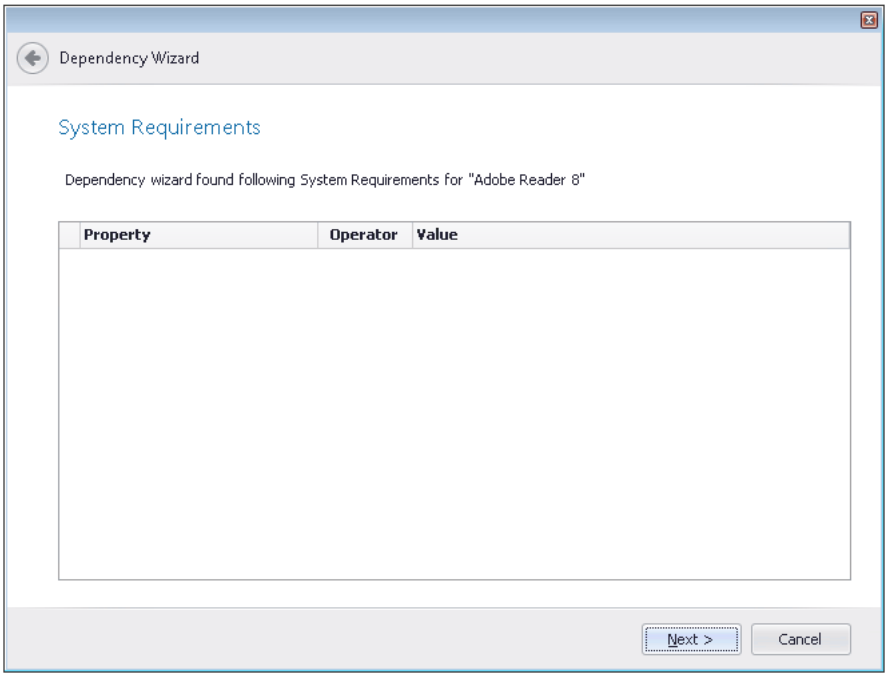


Figure 7-47: Dependency Wizard / System Requirements Panel

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the dependencies to the list.

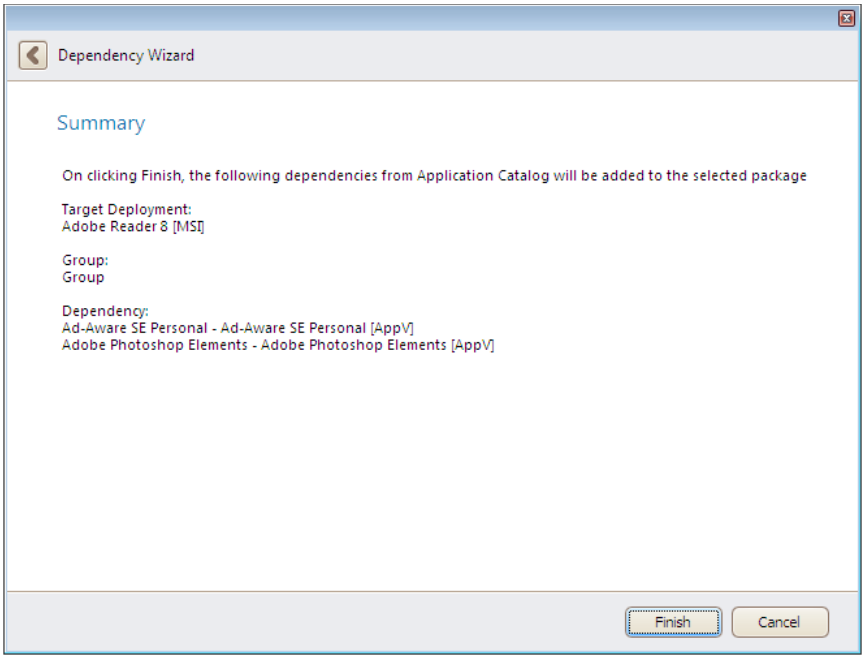


Figure 7-48: Dependency Wizard / Summary Panel

Import Apps List Wizard

You use the import Apps List Wizard to import a CSV file, exported from an external inventory system (such as Configuration Manager, FlexNet Manager Suite, or another CMDB system), that contains a list of all of your organization's applications. After import, AdminStudio can compare that list of imported packages against the Package Feed module data to determine which package tasks you can automate. AdminStudio's Package Feed Module includes information on package updates and the details needed to deploy them.

Note • *The apps list is expected to be in the .csv format. This file is expected to have the list of apps deployed in your organization across all the end point devices. Such a file is generally obtained (or exported) from an inventory system like ConfigMgr, Flexera's FlexNet Manager Suite (FNMS) etc.*

The Import Apps List Wizard panel contains the following panels:

- Select Apps List Source
- Import Apps List from CSV
- Summary
- Running the Import

Select Apps List Source

On the **Select Apps List Source** panel, **Import from csv** is automatically selected from the **Select source** list.

Note • *In future releases, additional source options will be added.*

Click **Next**. The **Import Apps List from CSV** panel opens.

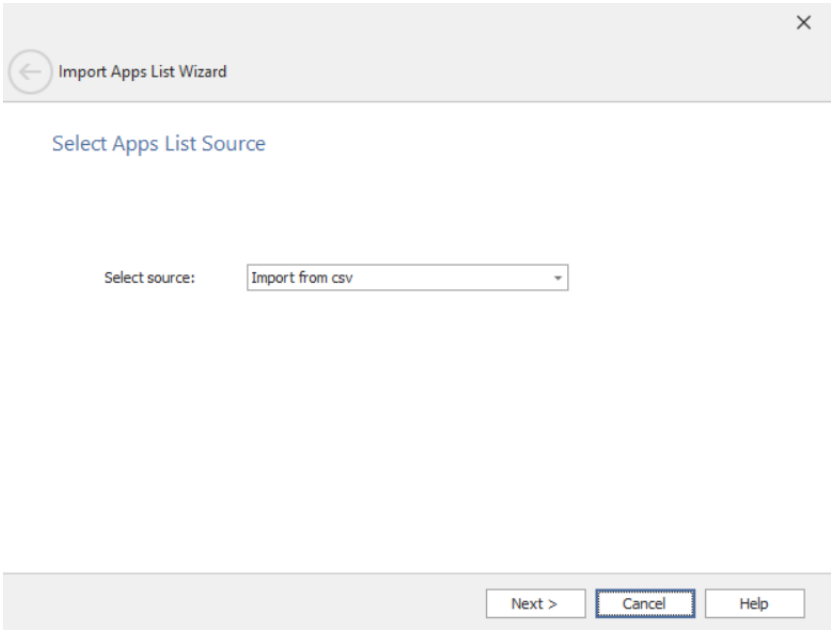


Figure 7-49: Import Appls List Wizard / Select Apps List Source Panel

Import Apps List from CSV

On the **Import Apps List from CSV** panel, you select the CSV file you want to import.

The CSV file is expected to contain a list of apps deployed in your organization across all the end point devices. Such a file is generally obtained (or exported) from an inventory system like ConfigMgr, Flexera's FlexNet Manager Suite (FNMS) etc.

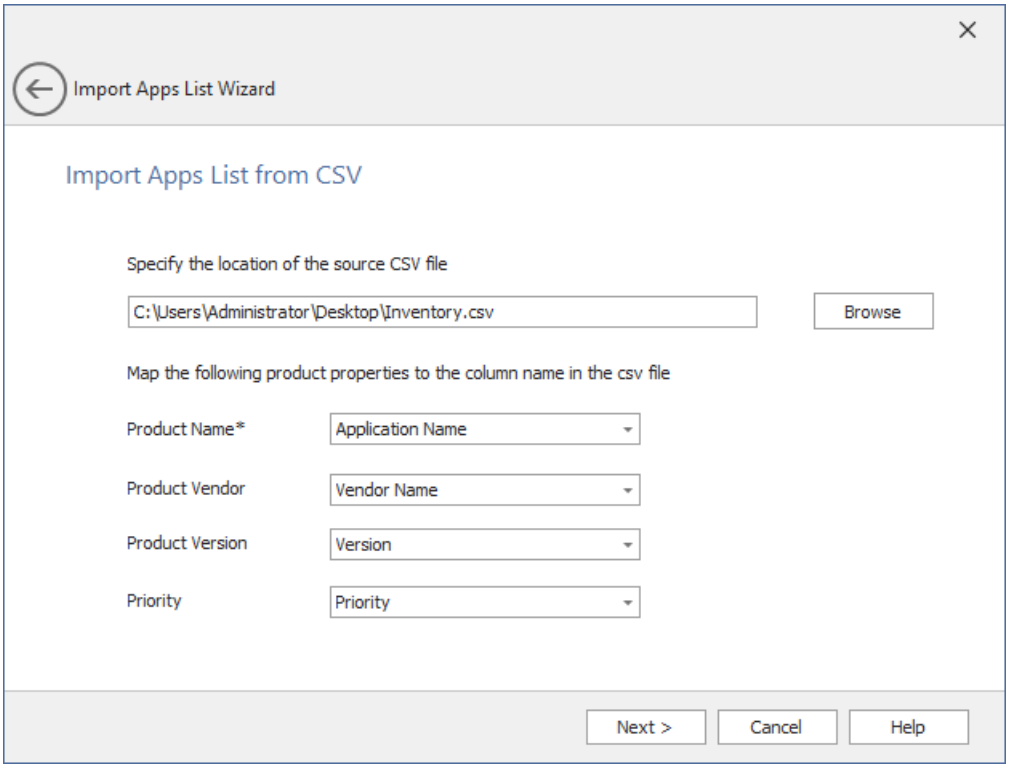


Figure 7-50: Import Apps List Wizard / Import Apps List from CSV

The Import Apps List from CSV panel has the following options:

Table 7-107 • Import Apps List from CSV Panel

Option	Description
Specify the location of the source CSV file	Click **Browse** and select the csv file which has a list of apps deployed in your organization. ***Important*** • *The CSV file can have any number of columns,. However it is mandatory that it has columns representing* ***Product Name***, ***Product Vendor***, ***Product Version***, *and* ***Priority****. These three columns will be imported into the AdminStudio Application Catalog.*

Table 7-107 • Import Apps List from CSV Panel

Option	Description
Product Name **Product Vendor** **Product Version** **Priority**	AdminStudio will automatically map the columns from the CSV file to Product Name, Product Vendor and Product Version, based on how the columns are named. If the mapping is not done automatically or if it is found to be incorrect, then select the appropriate column name from drop down to map to **Product Name**, **Product Vendor**, **Product Version**, and **Priority**. For example: • **Product Name** - Software Title, Application Name, Product, etc. • **Vendor Name** - Manufacturer, Publisher, or Vendor • **Product Version** - Software Versions or Version • **Priority** - Priority ***Note*** • *Providing **Product**, **Vendor**, **Version**, and **Priority** details improves the package suggestions provided in the **Version in Package Feed** and **Version in Catalog** columns of the AdminStudio **Backlog** tab list.* ***Note*** • *Any row in the CSV file with blank or empty value for the **Product Name** will be considered as an invalid record and will not be imported into AdminStudio. Product Vendor and Product Version can be blank or empty.*

Summary

When you click **Next** on the **Import Apps List from CSV** panel of the **Import Apps List Wizard**, the **Summary** panel opens. Review the specified options and click **Next** to begin the import.

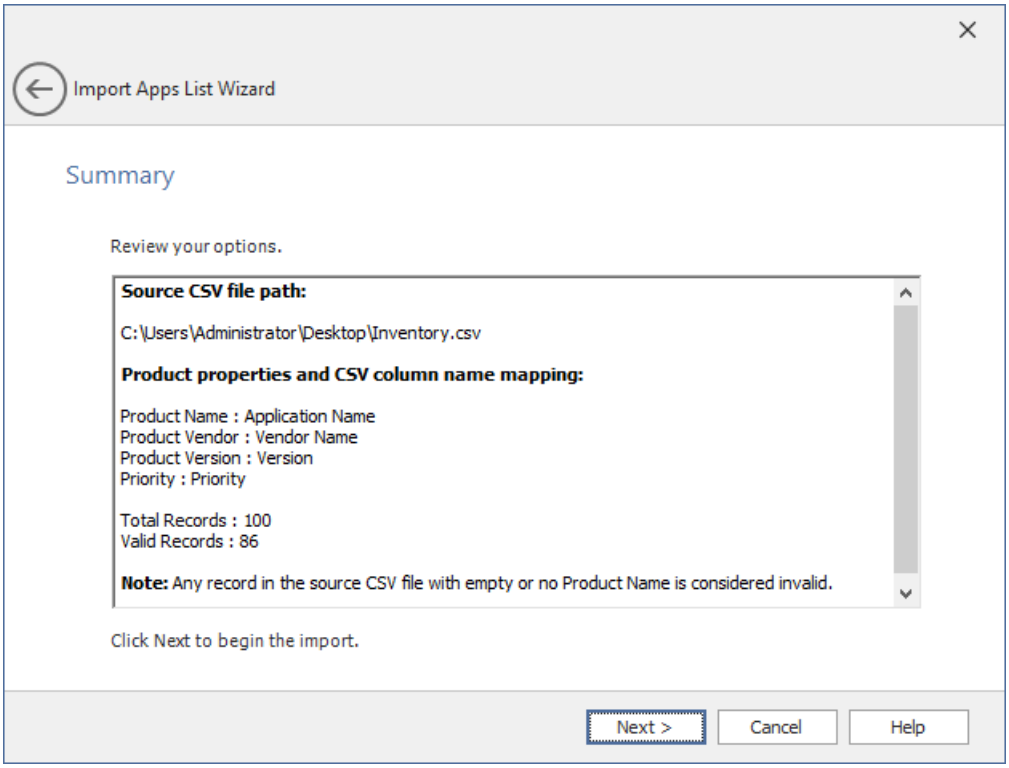


Figure 7-51: Import Apps List Wizard / Summary

Running the Import

When you click **Next** on the **Summary** panel of the **Import Apps List Wizard**, the **Running the Import** panel opens.

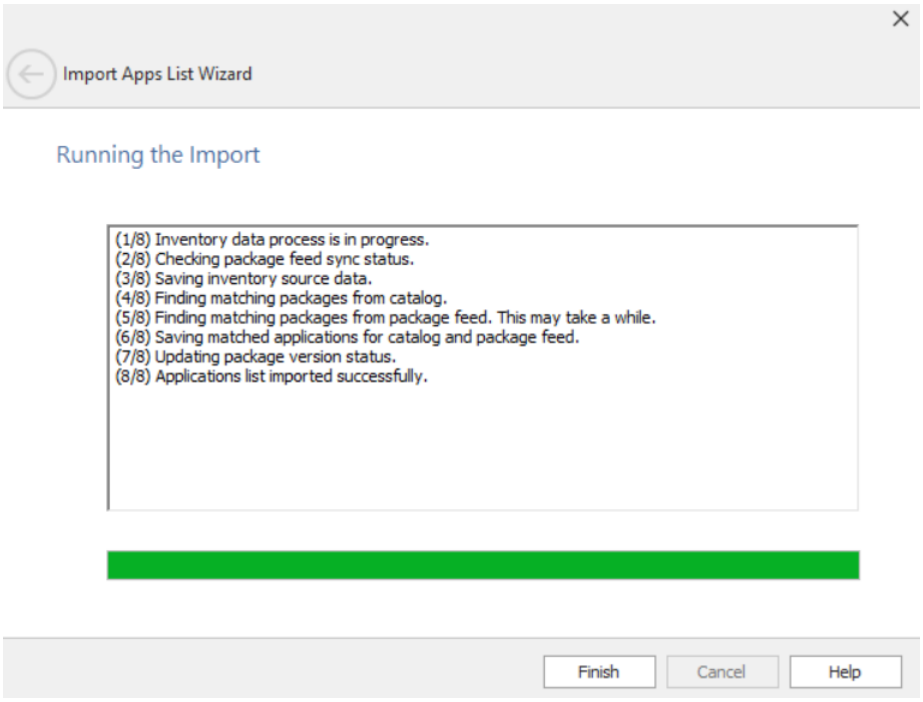


Figure 7-52: Import Apps List Wizard / Running the Import

After successful import, click **Finish**. Upon successful import, all app entries present in the CSV file will be displayed on the **Backlog** tab. For more information, see:

- Matching the Applications
- Configuring Actions for Automation
- Scheduling Automation

Import Wizard

The Import Wizard allows you to import the following installation package types into the Application Catalog: Windows Installer packages and associated transform and patch files, Microsoft UWP app packages, macOS desktop applications, virtual packages (Microsoft App-V, VMware ThinApp, and Citrix), mobile apps (Apple iOS, Google Android, and Windows Store), merge modules, OS snapshots, and other non-MSI setup formats (such as InstallShield Professional or ISMP installations).

The Import Wizard consists of the following panels:

- Source Panel
- Search for an Application in Package Feed Panel
- Provide Vendor Setup File Panel
- Silent Command Line Arguments
- Package Feed Module - Customization Wizard Panels

- Package Type Selection Panel (Single Application)
- Package Type Selection Panel (Folder of Multiple Applications)
- Enterprise Policy File Selection Panel
- Security Patch File Selection Panel
- OS Snapshot Selection Panel
- Public Store Selection Panel
- Store Application Selection Panel
- Source Server Details Panel
- Package File Selection Panel
- Package Folder Selection Panel
- Select Applications (Folder of Multiple Applications) Panel
- Select Applications/Packages Panel
- Package Support Files Panel
- Destination Group Panel
- Summary Panel
- Running the Import Panel

When an import is being performed, Application Manager displays its progress messages in the **Import** tab of the **Output Window**.

Source Panel

On the **Source** panel of the Import Wizard, which is opened by clicking the **Import** button on the **Home** tab of the Application Manager ribbon, specify the type of import that you want to perform.

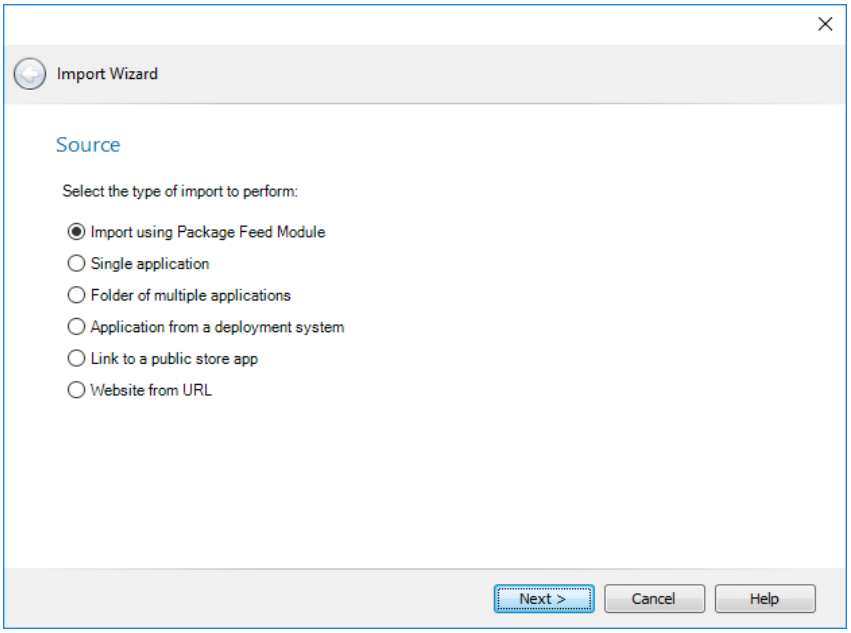


Figure 7-53: Import Wizard / Source Panel

On the **Source** panel, select one of the following options:

Table 7-108 • Source Panel

Option	Description
Import using Package Feed Module	Select this option to download and import an application into the Application catalog by using the Package Feed Module.
Single application	Select this option to import a single application into the Application Catalog.
Folder of multiple applications	Select this option to import a directory of applications into the Application Catalog. You can import packages of multiple deployment types using this option. You will be prompted to select which deployment types you wish to import. All of the installer packages of the selected deployment types that are located in the selected folder or its subfolders will be imported. You will also be given the option to mimic the directory structure of the selected directory.
Application from a deployment system	Select this option to import applications or packages from the following deployment systems: • Microsoft System Center 2012 Configuration Manager • Microsoft System Center 2007 Configuration Manager ***Note*** • *Prior to being able to import packages from one of these deployment systems, you first need to set up a connection, as described in Creating Multiple Named Connections to Distribution Systems.*

Table 7-108 • Source Panel

Option	Description
Link to a public store app	Select this option to import a deep link to a mobile app in a public store: Apple Store, Google Play Store, or Microsoft Windows Store.

Search for an Application in Package Feed Panel

Note *• To download and import using the Package Feed Module, see Importing an Application Using The Package Feed Module*

On the **Search for an application in Package Feed** panel, which opens when you select **Import using Package Feed Module** on the **Source** panel, you are prompted to search for the application that you want to import.

You can search for an application using:

- Product Name
- Vendor Name

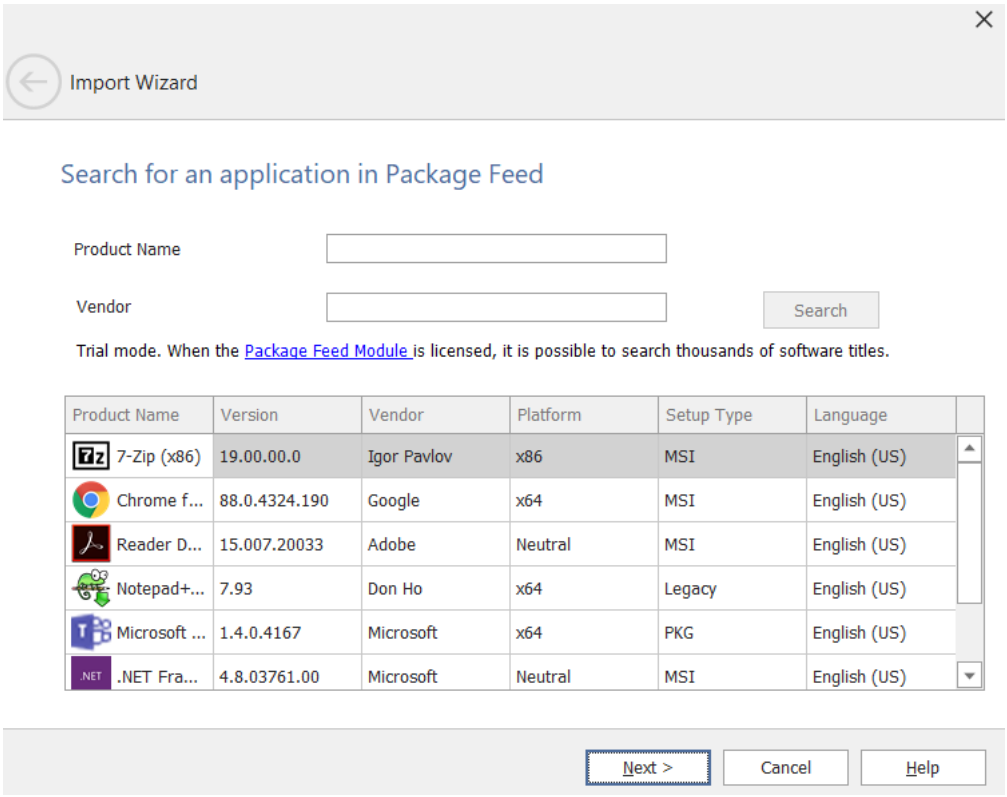


Figure 7-54: Import Wizard / Search for an Application in Package Feed

Provide Vendor Setup File Panel

Note • *To download and import using the Package Feed Module, see Importing an Application Using The Package Feed Module*

On the **Provide Vendor Setup file** panel, which opens when you select **Import using Package Feed Module** on the **Source** panel, you are prompted to browse or download the application that you want to import.

Provide Vendor Setup file panel also gives the below details:

- Product Name
- Version
- Vendor Name
- Setup Type
- Language
- Platform

In the **Choose setup file option** field, you can use the **Browse** option to select the directory that contains the packages that you want to import into the Application Catalog.

Click Download button to initialize the download.

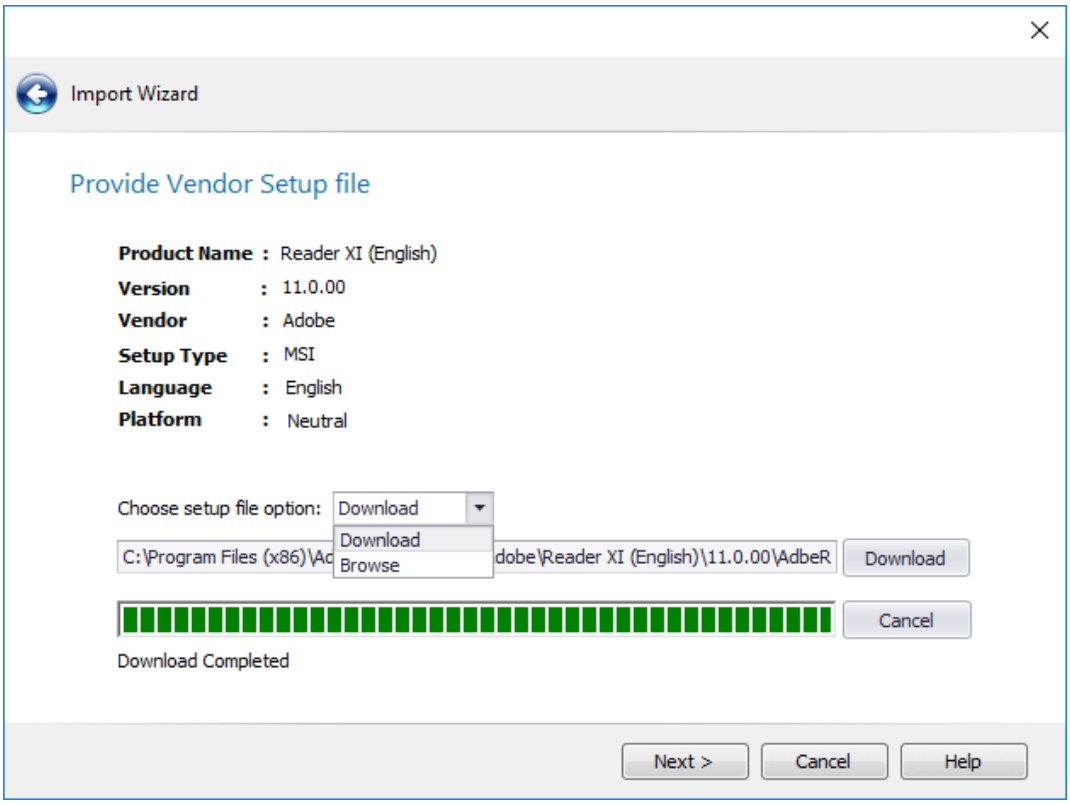
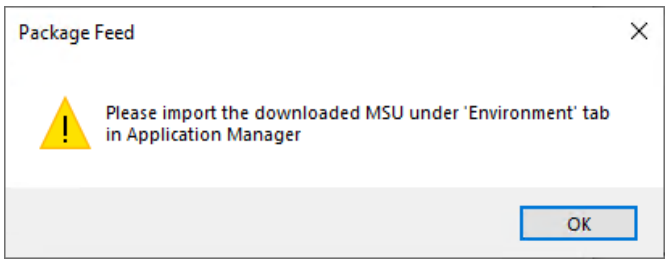


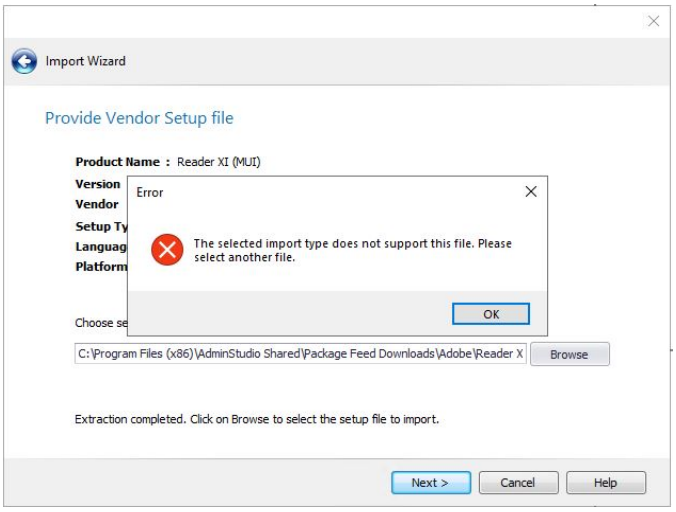
Figure 7-55: Import Wizard / Provide Vendor Setup file in Package Feed

Note • *Following message appears for the below package types:*

- *When the package type is MSU the below message appears, see Security Patch File Selection Panel for further instructions.*



- *You need to **Extract** the downloaded zip file and **Browse** for the extracted setup file. The below error message appears, when you click **Next** without browsing for the extracted set up files.*



Silent Command Line Arguments

Note • *To download and import using the Package Feed Module, see Importing an Application Using The Package Feed Module*

On the **Silent Command Line Arguments** panel, which opens when you select **Import using Package Feed Module** on the **Source** panel, you are provided with the Install Command Line and Uninstall Command Line fetched from the package feed module of the package. If package can be customized then **Create transform file mst file with additional custom installation options** will be enabled and checked by default if deployment type is msi.

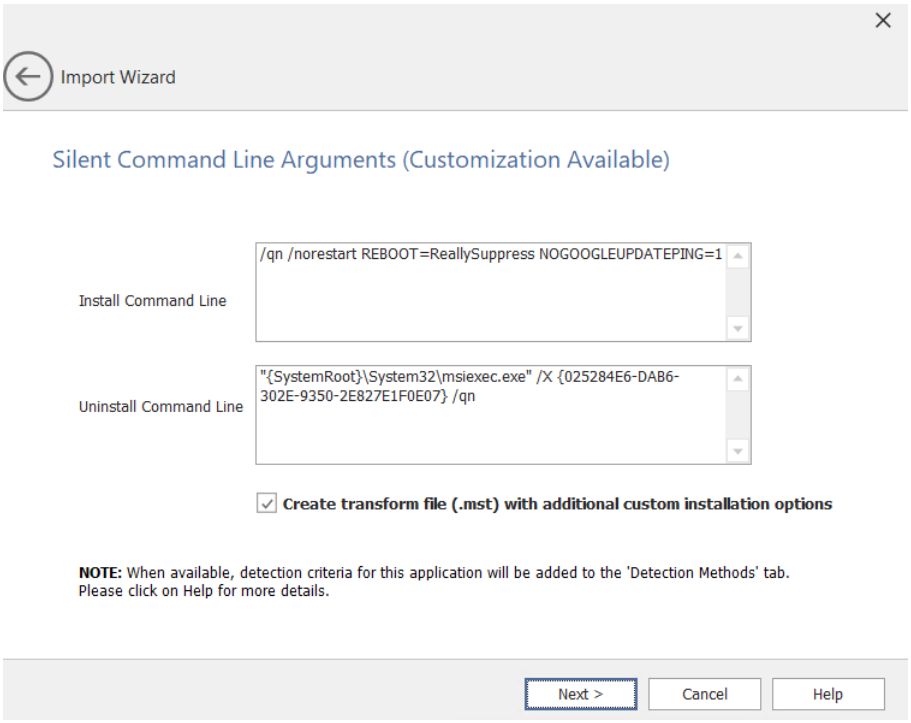


Figure 7-56: Import Wizard / Silent Command Line Arguments (Customization Available)

Important • *AdminStudio adds **Detection Methods** while importing an application from the Package Feed Module, If the application that is being imported from the Package Feed Module is of **Neutral** architecture, then AdminStudio will treat it as a **32-bit** application on a 64-bit system.*

See Also
Detection Method Wizard

Package Feed Module - Customization Wizard Panels

Note • *To download and import using the Package Feed Module, see Importing an Application Using The Package Feed Module.*

The **Customization Wizard** allows you to select from various customization options for a given package. The Customization panels appear in the **Import Wizard** and are dynamic depending on the number of customization options available for the selected package.

Below are examples of four custom panels that would appear if you were using the Package Feed Module to import Google Chrome.

For more information on Customization Wizard, see Customization Wizard.

Options Panel

The following **Options** panel is a custom dynamic panel for the Google Chrome package.

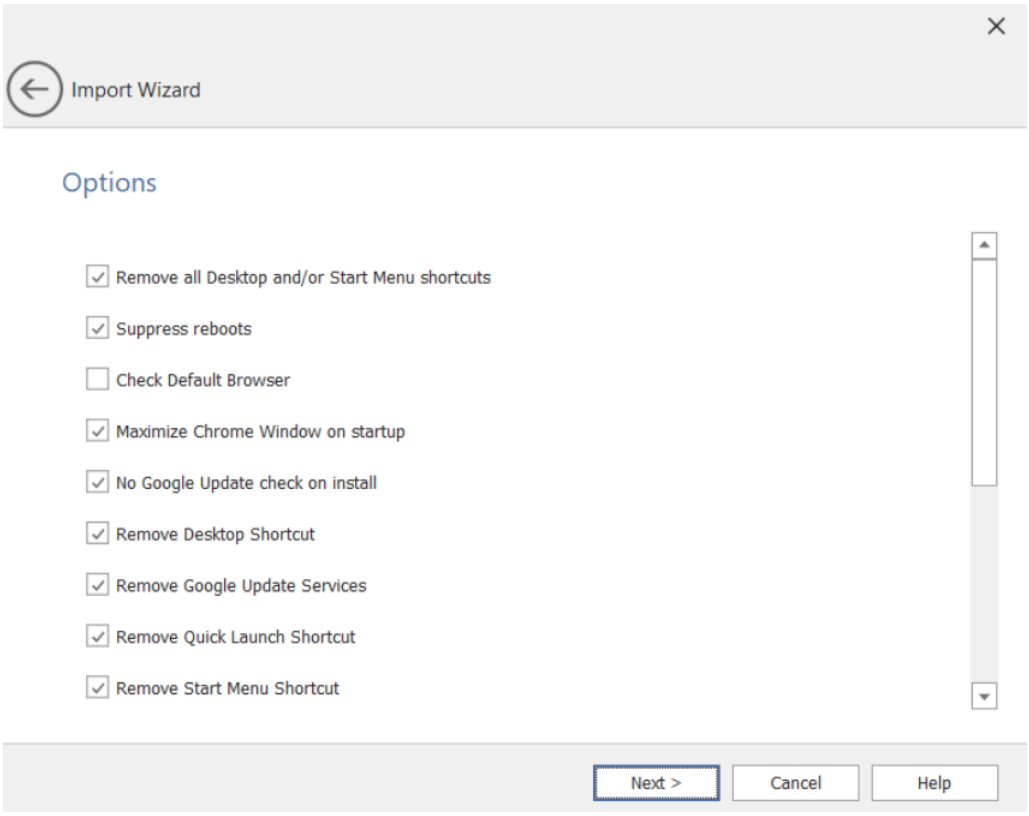


Figure 7-57: Dynamic Panel: Options

Home Page Preferences Panel

The following **Home Page Preferences** panel is a custom dynamic panel for the Google Chrome package.

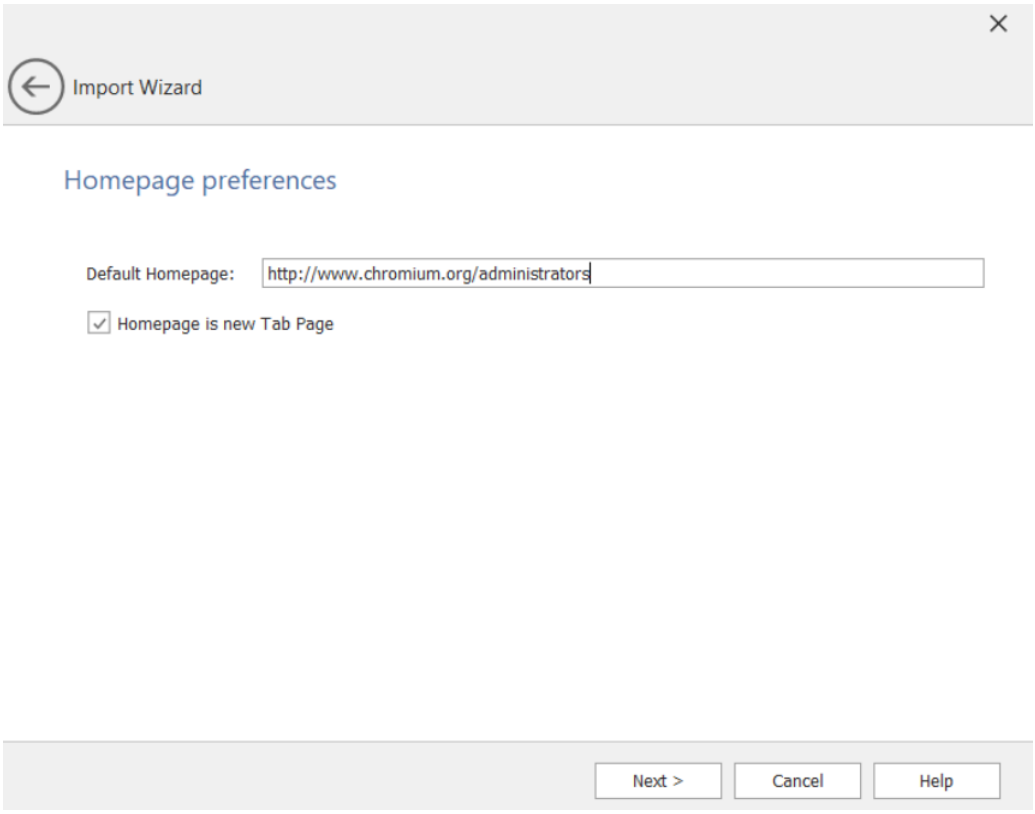


Figure 7-58: Dynamic Panel: Home Page Preferences

Distribution Preferences Panel

The following **Distribution Preferences** panel is a custom dynamic panel for the Google Chrome package.

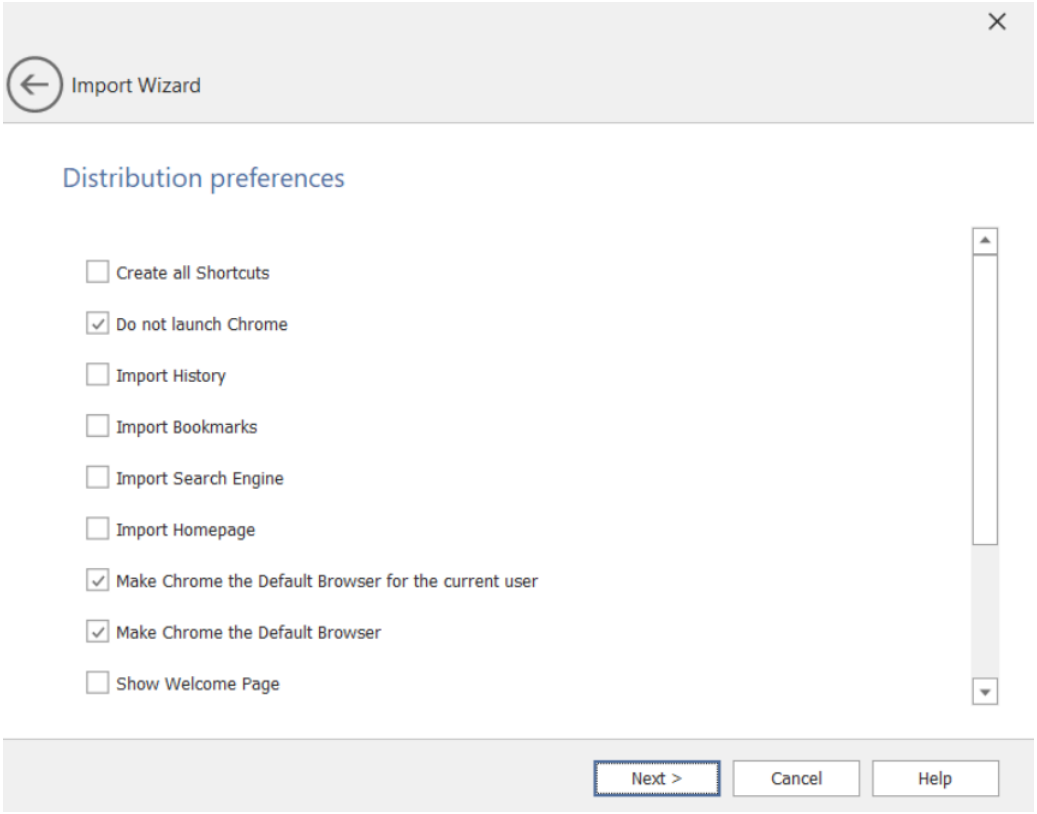


Figure 7-59: Dynamic Panel: Distribution Preferences

Release Notes Panel

The following **Release Notes** panel is a custom dynamic panel for the Google Chrome package.

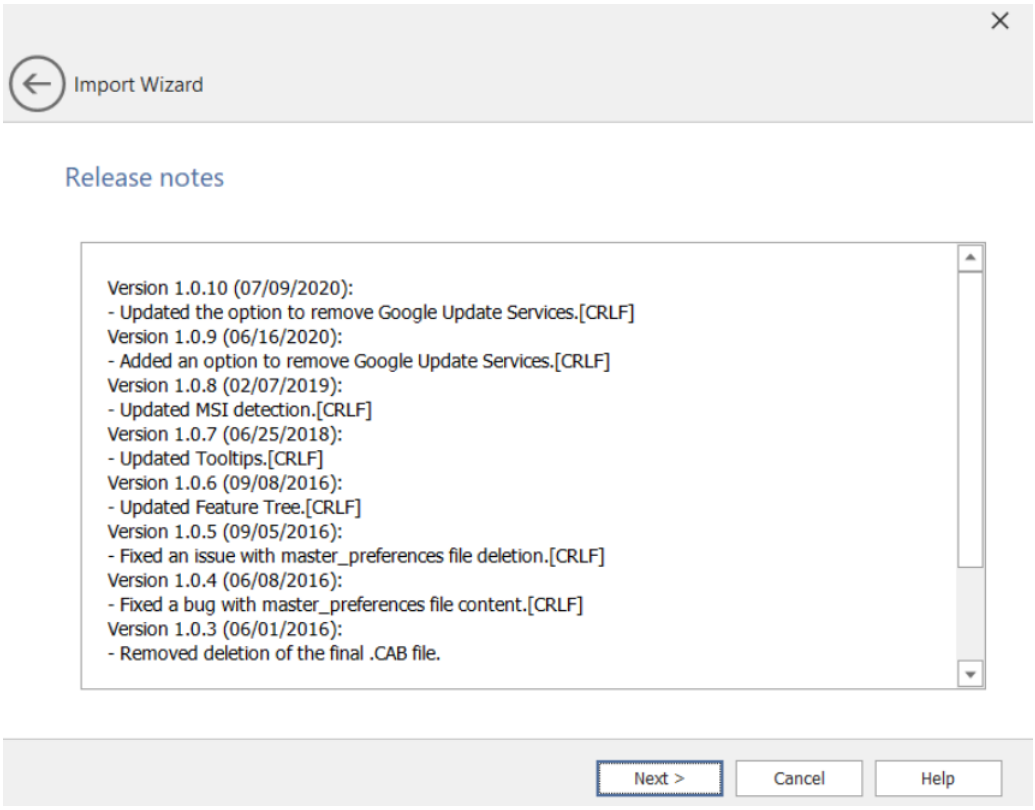


Figure 7-60: Dynamic Panel: Release Notes

Package Type Selection Panel (Single Application)

On the **Package Type Selection** panel, which opens when you select **Single application** on the **Source** panel, you are prompted to select the type of package that you want to import.

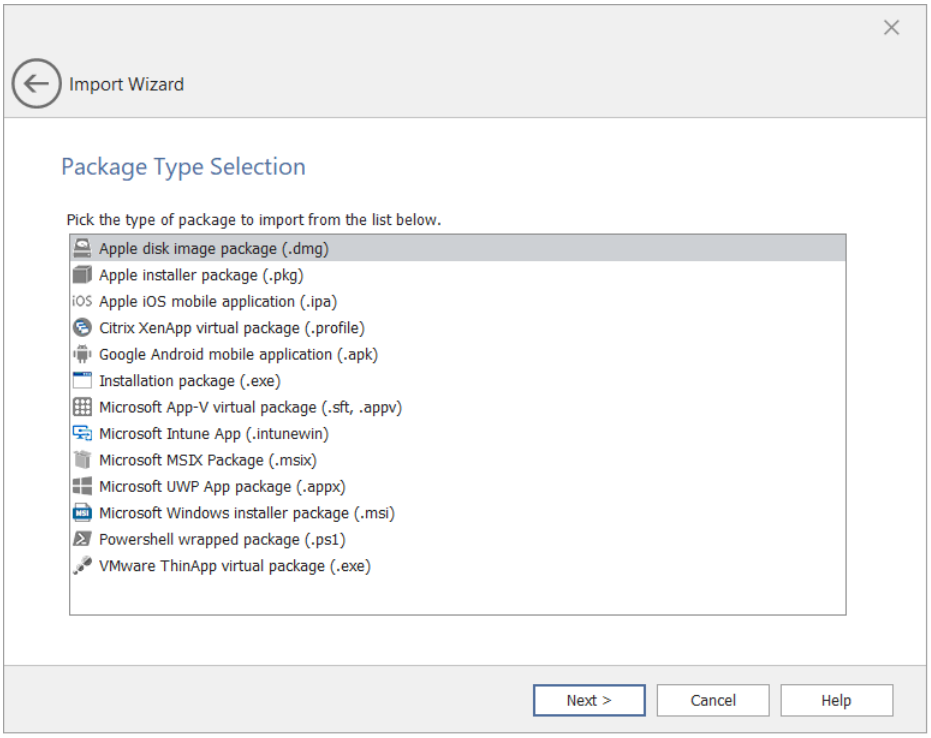


Figure 7-61: Import Wizard / Package Type Selection Panel (Single Application)

Select one of the following package types and then click **Next**:

Table 7-109 • Package Types Listed on Package Type Selection Panel

Package Type	Description
Apple disk image package (.dmg)	macOS desktop application. When you double-click a `.dmg` file, an Apple disk image is "mounted" as a volume within the Finder. To install the application, you usually drag the application icon from the disk image into the Applications folder.
Apple installer package (.pkg)	macOS desktop application. When you double-click a `.pkg` file, the Apple installer application is launched, and you install the package by proceeding through an installation wizard.
Apple iOS mobile app (.ipa)	File used to distribute and install an app on devices (iPhones and iPads) running the Apple iOS operating system.
Citrix XenApp virtual package (.profile)	Virtual package for deployment on Citrix XenApp, an application delivery system for Windows applications.
Google Android mobile app (.apk)	File used to distribute and install an app on devices (phones and tablets) running the Google Android operating system.

Table 7-109 • Package Types Listed on Package Type Selection Panel

Package Type	Description
Installation package (.exe)	Non-MSI legacy setup types (such as InstallShield Professional or ISMP installations). Also, complex installer executable files (`.exe`) that contain bundled Windows Installer packages, including: • InstallShield InstallScript `.exe` files • InstallShield Basic MSI installers that are compressed into a `setup.exe` file • InstallShield Suite Installer `.exe` files • Wise Package Studio `.exe` files • Other executable file types that can be uncompressed by 7-ZIP
Microsoft App-V virtual package (.sft, .appv)	Virtual application designed to run on the Microsoft Application Virtualization platform. An App-V 4.5 or 4.6 package has an `.sft` extension, while an App-V 5.0 package has an `.appv` extension.
Microsoft MSIX Package (.msix)	The Microsoft MSIX package (.msix) format is a newly introduced package by Microsoft.
Microsoft UWP app package (.appx)	The UWP app package (`.appx`) format is the packaging format used to distribute and install apps on Windows 8.x and 10 and is the only format allowed for Universal Windows Platform (UWP) apps. ***Note*** • *You can also upload Windows 8 apps (`.appx`) by selecting the **Microsoft UWP app package (.appx)** package type.* ***Tip*** • *InstallShield 2016 supports creating the UWP app package format (`.appx`) and its desktop and server extensions through an alternate build output, and provides suitability testing to help you identify items unsuitable for the UWP app package format.*
Microsoft Windows Installer package (.msi)	File that contains all of the information that the Windows Installer requires to install or uninstall an application or product and to run the setup user interface. The `.msi` file can also contain one or more transform files (`.mst`) and one or more patches (`.msp`).
PowerShell wrapped package (.ps1)	PowerShell wrapped package (`.ps1`) file, which contains bundled packages.
VMware ThinApp virtual package (.exe)	Self-contained virtual package in VMware ThinApp format that requires no client-side agents or supporting server infrastructure.
Microsoft Intune App (.intunewin)	Select to import a **Microsoft Intune App** (.intunewin) in to the **Application Catalog**.

Package Type Selection Panel (Folder of Multiple Applications)

On the **Package Type Selection** panel, which opens when you select **Folder of multiple applications** on the **Source** panel, you are prompted to select the types of packages that you want to import.

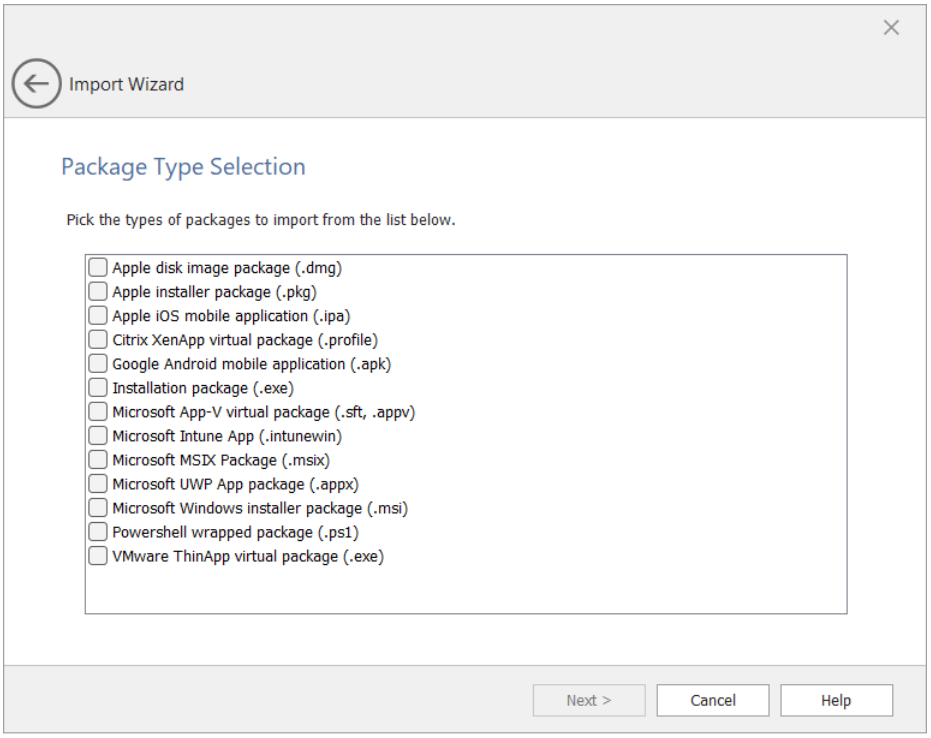


Figure 7-62: Import Wizard / Package Type Selection Panel (Folder of Multiple Applications)

Select one or more of the following package types and then click **Next**:

Table 7-110 • Package Types Listed on Package Type Selection Panel

Package Type	Description
Apple disk image package (.dmg)	When you double-click a `.dmg` file, an Apple disk image is "mounted" as a volume within the Finder. To install the application, you usually drag the application icon from the disk image into the Applications folder.
Apple installer package (.pkg)	Double-clicking a `.pkg` file launches the Apple installer application, where the package is installed by proceeding through an installation wizard.
Apple iOS mobile app (.ipa)	File used to distribute and install an app on devices (iPhones and iPads) running the Apple iOS operating system.
Citrix XenApp virtual package (.profile)	Virtual package for deployment on Citrix XenApp, an application delivery system for Windows applications.
Google Android mobile app (.apk)	File used to distribute and install an app on devices (phones and tablets) running the Google Android operating system.

Table 7-110 • Package Types Listed on Package Type Selection Panel

Package Type	Description
Installation package (.exe)	Non-MSI legacy setup types (such as InstallShield Professional or ISMP installations). Also, complex installer executable files (.exe) that contain bundled Windows Installer packages, including: • InstallShield InstallScript .exe files • InstallShield Basic MSI installers that are compressed into a setup.exe file • InstallShield Suite Installer .exe files • Wise Package Studio .exe files • Other executable file types that can be uncompressed by 7-ZIP
Microsoft App-V virtual package (.sft, .appv)	Virtual application designed to run on the Microsoft Application Virtualization platform. An App-V 4.5 or 4.6 package has an `.sft` extension, while an App-V 5.0 package has an `.appv` extension.
Microsoft Intune package (.intunewin)	A package format that can be distributed using Microsoft Intune, a cloud-based service that focuses on mobile device management and mobile application management.
Microsoft MSIX Package (.msix)	The Microsoft MSIX package (.msix) format is a newly introduced package by Microsoft.
Microsoft UWP app package (.appx)	The UWP app package (`.appx`) format is the packaging format used to distribute and install apps on Windows 8.x and 10 and is the only format allowed for Universal Windows Platform (UWP) apps. ***Note*** • *You can also upload Windows 8 apps (`.appx`) by selecting the **Microsoft UWP app package (.appx)** package type.* ***Tip*** • *InstallShield 2016 supports creating the UWP app package format (.appx) and its desktop and server extensions through an alternate build output, and provides suitability testing to help you identify items unsuitable for the UWP app package format.*
Microsoft Windows Installer package (.msi)	File that contains all of the information that the Windows Installer requires to install or uninstall an application or product and to run the setup user interface. The `.msi` file can also contain one or more transform files (`.mst`) and one or more patches (`.msp`).
PowerShell wrapped package (.ps1)	PowerShell wrapped package (.ps1) file, which contains bundled packages.
VMware ThinApp virtual package (.exe)	Self-contained virtual package in VMware ThinApp format that requires no client-side agents or supporting server infrastructure.

Enterprise Policy File Selection Panel

On the **Enterprise Policy File Selection** panel, which opens when you select the **Enterprise Policy Configurations** group on the **Environment** tab of the Application Manager tree and then click **Import** in the toolbar, you are prompted to select the enterprise policy file that you want to import.

For information on enterprise policy files, see Managing iOS Enterprise Policy Configuration Files.

Browse to an enterprise policy file and then click **Next** to continue.

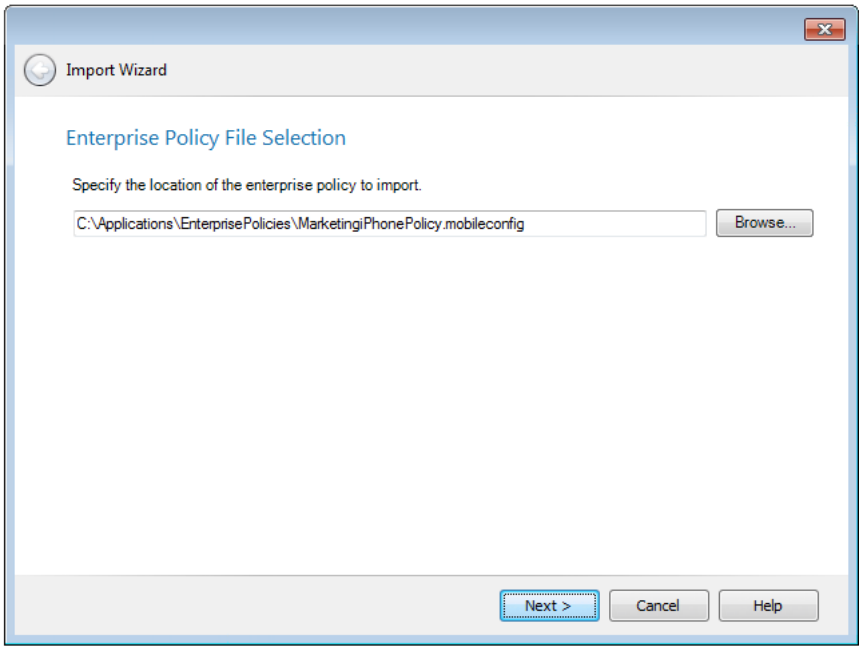


Figure 7-63: Import Wizard / Enterprise Policy File Selection

Security Patch File Selection Panel

On the **Security Patch File Selection** panel, which opens when you select the **Security Patches** group on the **Environment** tab of the Application Manager tree and then click **Import** in the toolbar, you are prompted to select the security patch file that you want to import.

Browse to a security patch file and then click **Next** to continue.

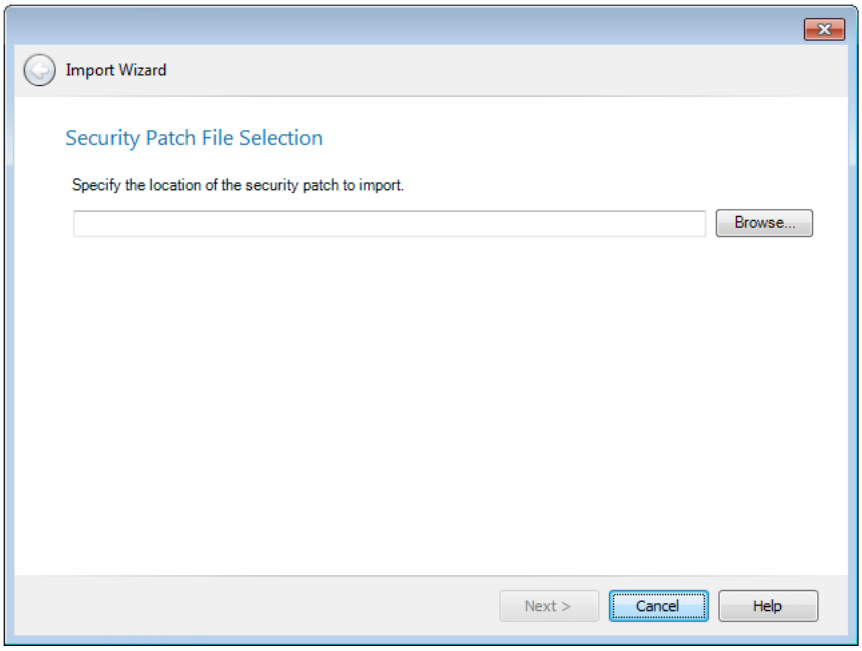


Figure 7-64: Import Wizard / Security Patch File Selection

For information on importing Microsoft Security Patch files and performing Patch Impact Analysis, see Analyzing the Impact of Installing Microsoft Operating System Security Patches.

OS Snapshot Selection Panel

On the **OS Snapshot File Selection** panel, which opens when you select the **Snapshots** group on the **Environment** tab of the Application Manager tree and then click **Import** in the toolbar, you are prompted to select the OS snapshot file that you want to import.

Browse to OS snapshot file and then click **Next** to continue.

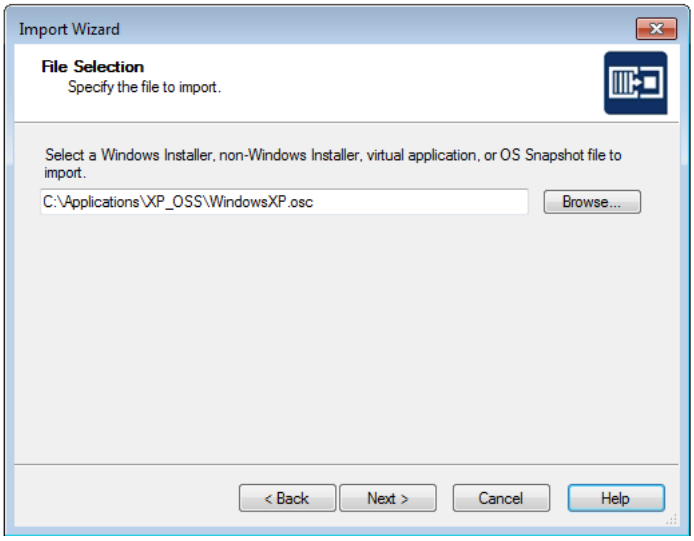


Figure 7-65: Import Wizard / File Selection

Public Store Selection Panel

On the **Public Store Selection** panel, which opens when you select **Link to a public store app** on the **Source** panel, you are prompted to select the public store from which you want to import a deep link to a mobile app,

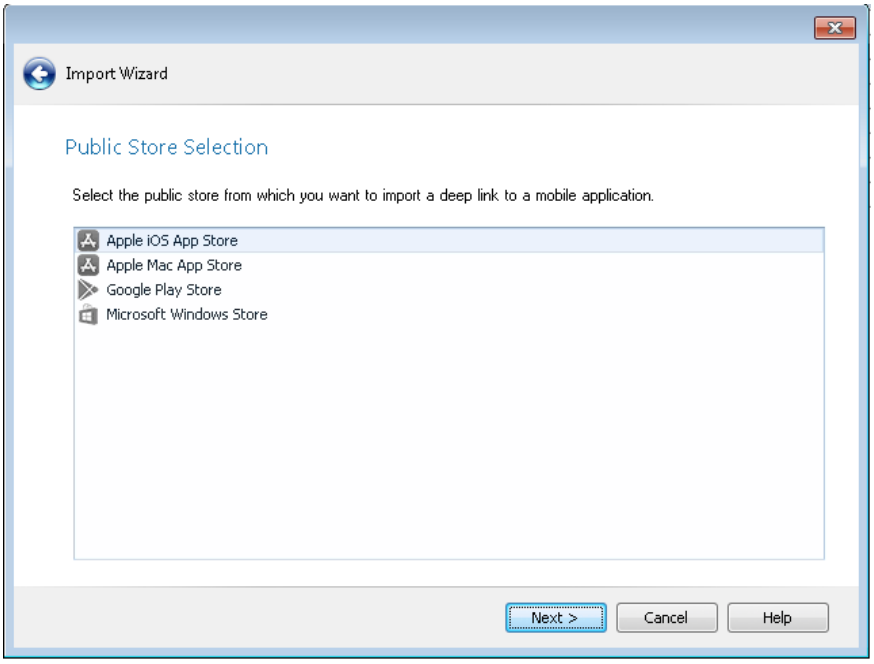


Figure 7-66: Import Wizard / Public Store Selection

You can import links to mobile apps in the Apple iOS App Store, Google Play Store, or Microsoft Windows Store into the Application Catalog. You can also import links to desktop applications in the Apple Mac App Store. This enables you to prepare and manage public store applications in conformance with your standard application readiness processes.

For more information on importing public store mobile apps, see Importing Links to Public Store Applications and Managing Mobile App Metadata.

Store Application Selection Panel

On the **Store Application Selection** panel, which opens when you select **Link to a public store app** on the **Source** panel and select a public store type on the **Public Store Selection** panel, you are prompted to browse to the public store app that you want to insert a link to.

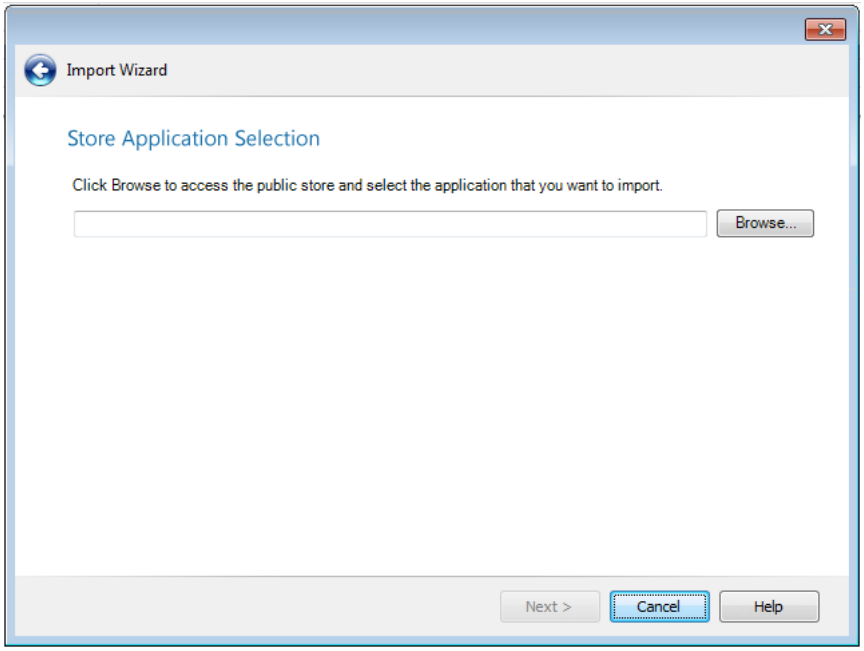


Figure 7-67: Import Wizard / Store Application Selection

Click **Browse** to open the **Browse Application from Store** dialog box, which displays the browser window of the selected public store, such as the Apple App Store or Google Play Store.

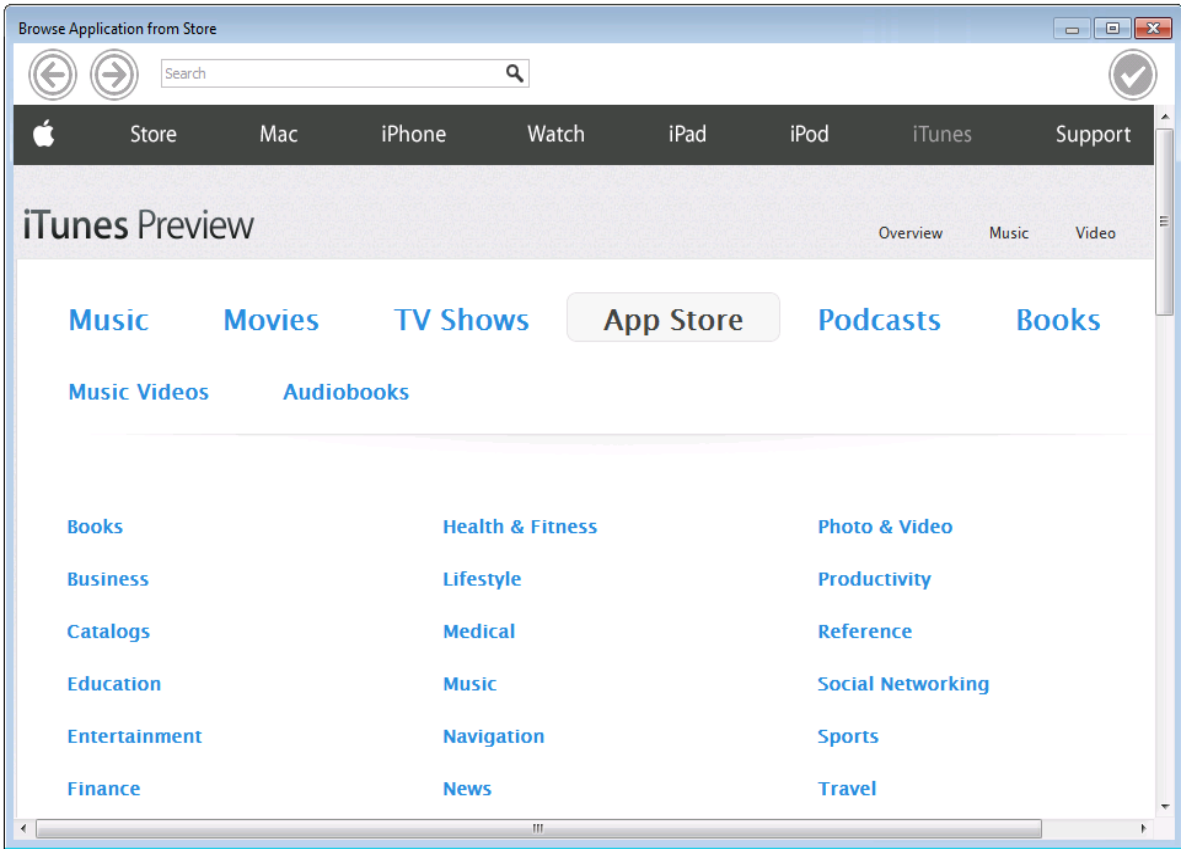


Figure 7-68: Apple App Store

On the **Browse Application from Store** dialog box, use the links in to locate the desired mobile app and open its informational page.

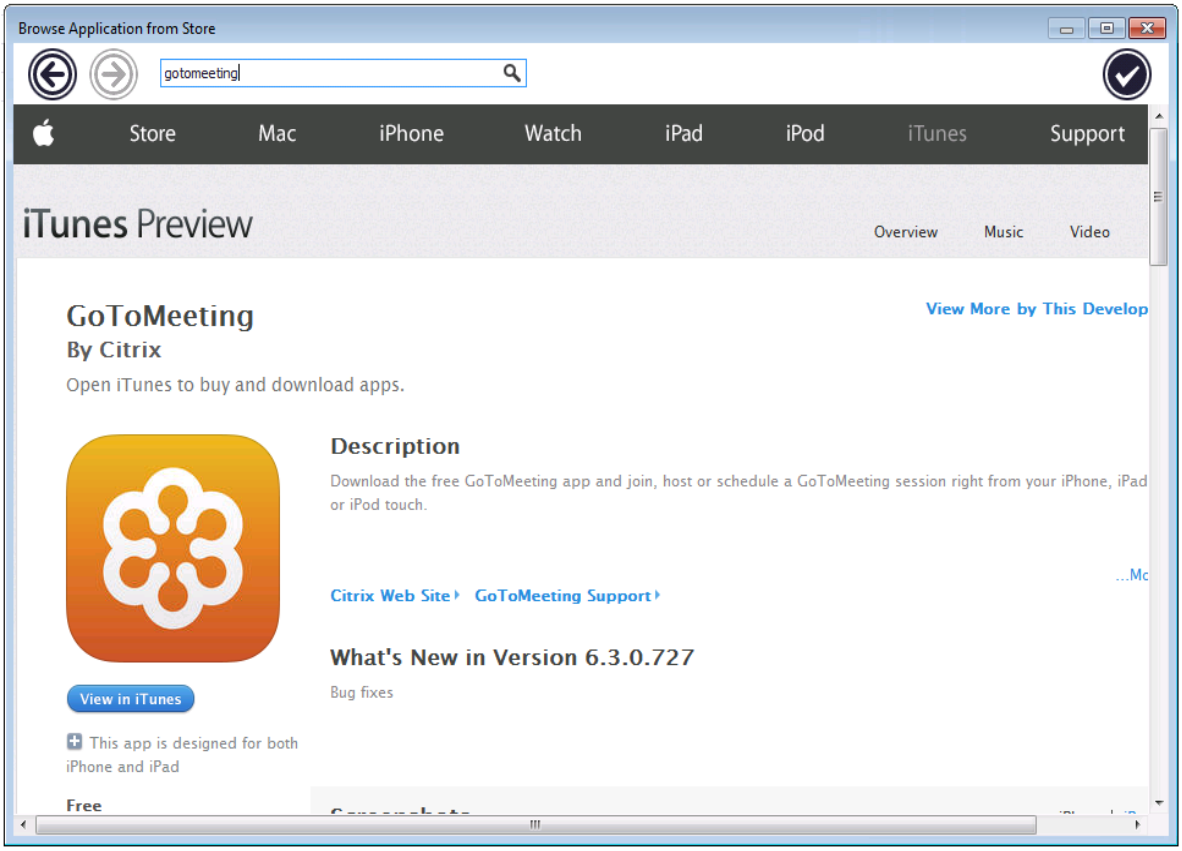


Figure 7-69: Mobile App Informational Page on Apple App Store

Note • *You can also use the arrow keys at the top left of the dialog box to navigate through the public store.*

When you have opened the informational page of the mobile app that you would like to import, click the checkmark button at the top right of the dialog box.

The link to the selected mobile app is now listed on the **Store Application Selection** panel, such as:

`https://itunes.apple.com/us/app/gotomeeting/id424104128?mt=8`

Click **Next** to continue.

Source Server Details Panel

If you select the **Packages from a deployment system** option on the **Source** panel, the **Source Server Details** panel opens and prompts you to select a named connection to a ConfigMgr (Formerly called as System Center Configuration Manager) server.

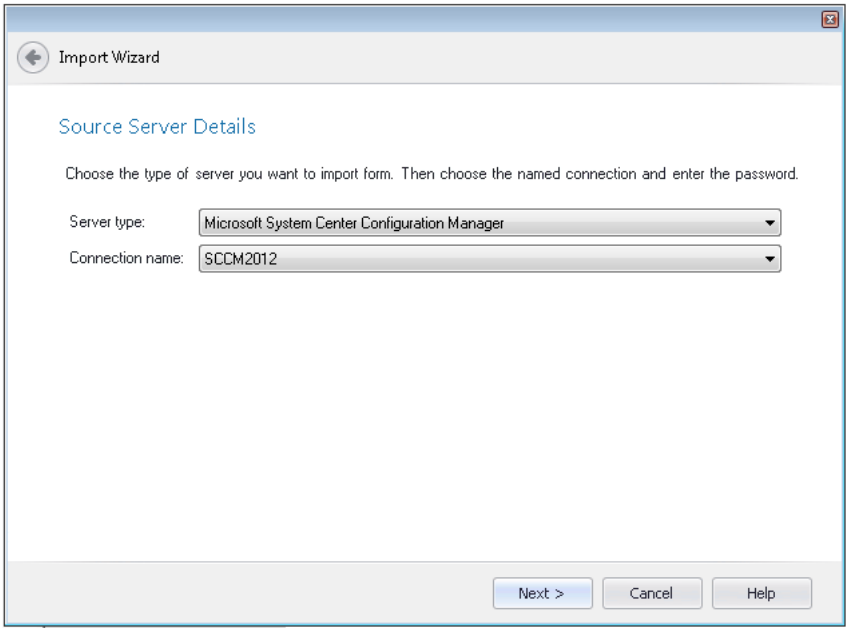


Figure 7-70: Import Wizard / Source Server Details Panel

In order to import packages from ConfigMgr (Formerly called as System Center Configuration Manager), you must first set up a named connection, as described in Creating Multiple Named Connections to Distribution Systems.

After you create a connection, it will be available for selection on the **Source Server Details** panel.

Package File Selection Panel

When you select **Single application** on the **Source** panel of the Import Wizard, the **Package File Selection** panel opens, prompting you to select the package that you want to import.

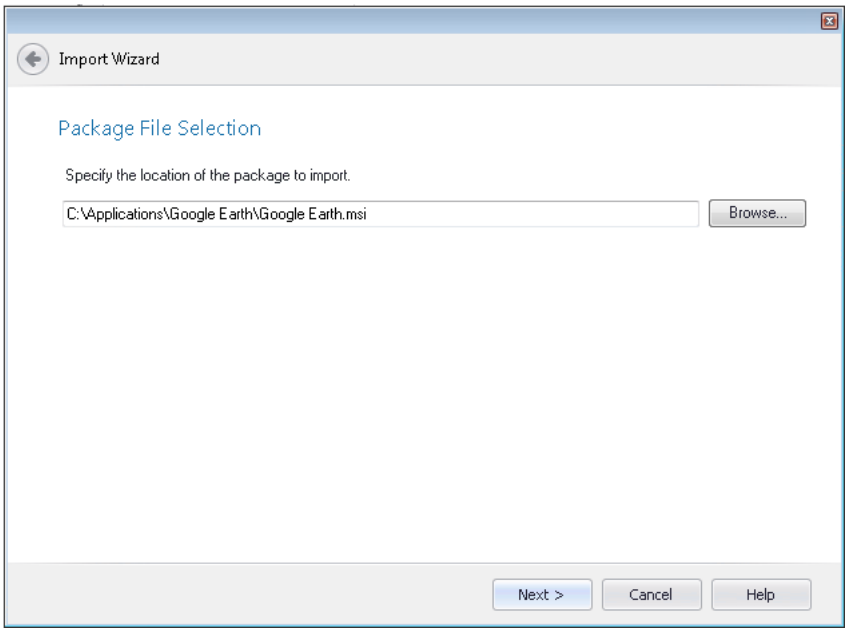


Figure 7-71: Import Wizard / Package File Selection Panel

Browse to the package that you want to import and click **Next**.

Note • *The package that you select must be of the type that you selected on the **Package Type Selection** panel.*

Package Folder Selection Panel

On the **Package Folder Selection** panel of the Import Wizard, which opens if you select **Folder of multiple applications** on the **Source** panel, you are prompted to select the directory that contains the packages that you want to import into the Application Catalog.

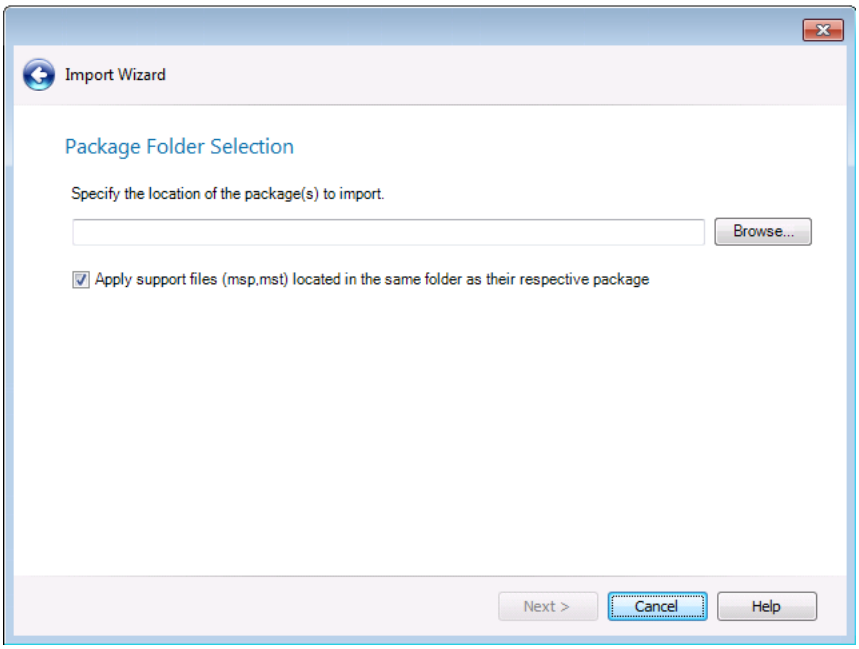


Figure 7-72: Import Wizard / Package Folder Selection Panel

In the **Specify the location of the package(s) to import** field, enter or browse to the directory that contains the packages that you want to import, and then click **Next**.

Optionally, if you also want to import package support files (such as transforms or patch files), select the **Apply support files (.msp, .mst) located the same folder as their respective package** option.

The Import Wizard will search the selected directory and its subdirectories to locate the packages of the types you selected on the Package Type Selection Panel (Folder of Multiple Applications).

Import Wizard's Selection Rules When Importing Packages from a Directory

When importing packages from a directory, the Import Wizard will scan the selected directory and all of its subdirectories for the package types you selected on the Package Type Selection Panel (Folder of Multiple Applications) panel, and will import all packages that are found.

On the **Destination Group** panel of the Import Wizard, you have the option to create subgroups in the Application Catalog based on the subdirectory structure of the selected directory to contain the imported packages, or to import all of the packages into the root of the specified group.

Select Applications (Folder of Multiple Applications) Panel

When you choose **Folder of multiple applications** on the **Source** panel of the Import Wizard, after you specify the directory location of the packages to import, the **Select Applications** panel opens, listing all of the packages of the selected package type in the specified directory.

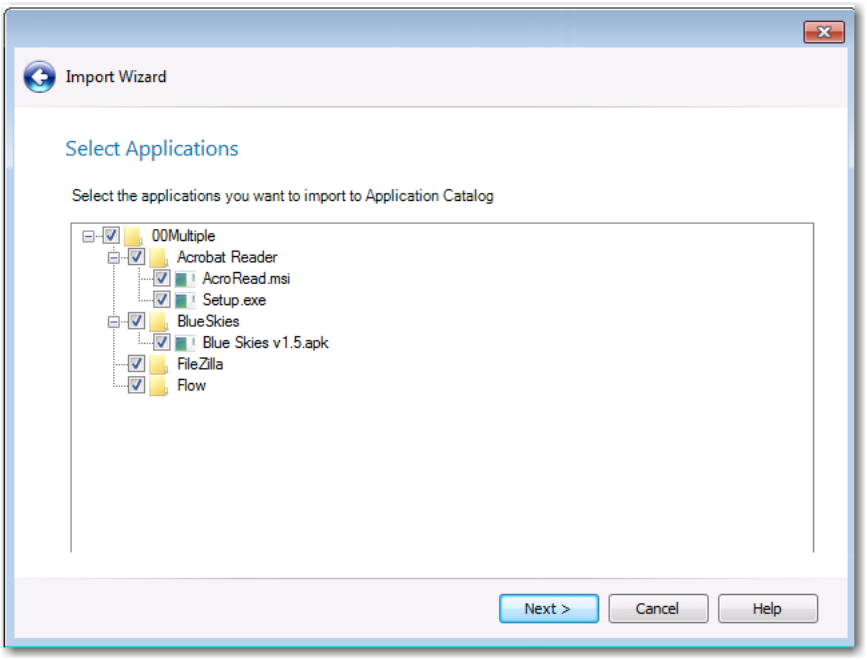


Figure 7-73: Select Applications Panel of Import Wizard (When Importing Folder of Multiple Applications)

By default, all applications containing the selected package type are selected. You can clear the selection of any packages you do not want to import.

Select Applications/Packages Panel

The name and functionality of this panel, which opens after you successfully connect to a ConfigMgr (Formerly called as System Center Configuration Manager) server, depends upon whether you are connected to a System Center 2012 Configuration Manager server or a System Center 2007 Configuration Manager Server.

Select Applications Panel

If you are connected to a System Center 2012 Configuration Manager server, the **Select Applications** panel opens, listing all of the applications in the connected server and prompting you to select the applications you want to import.

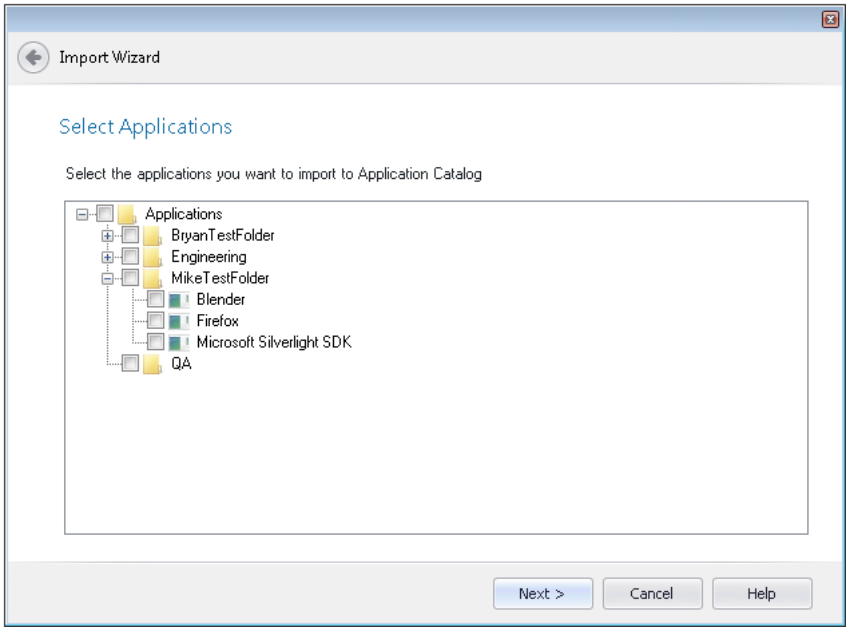


Figure 7-74: Import Wizard / Select Applications Panel

Select the applications that you want to import and click **Next**.

Select Packages Panel

If you are connected to a System Center 2007 Configuration Manager server, the **Select Packages** panel opens, listing all of the packages in the connected server and prompting you to select the packages you want to import.

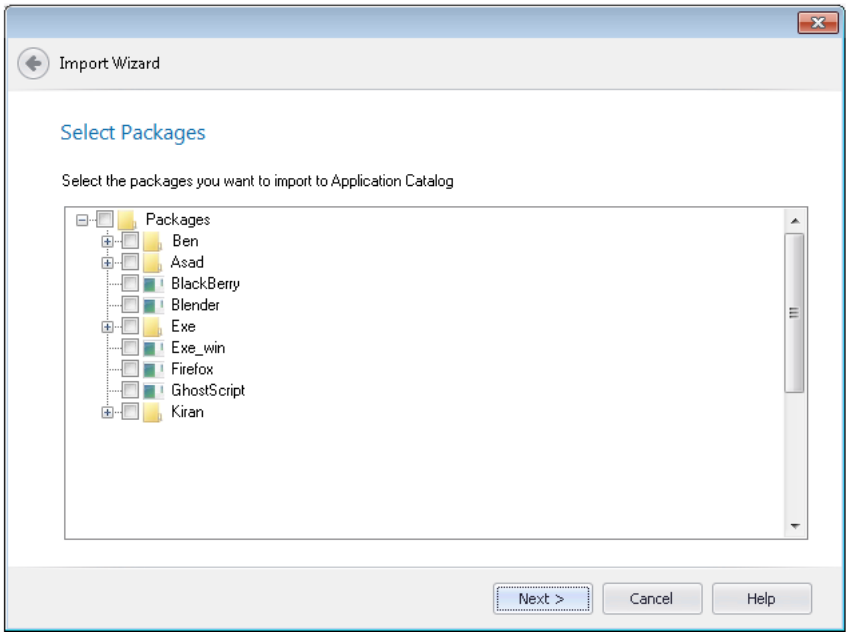


Figure 7-75: Import Wizard / Select Packages Panel

Select the packages that you want to import and click **Next**.

Package Support Files Panel

On the **Package Support Files** panel, you are prompted to optionally import additional support files along with the selected package.

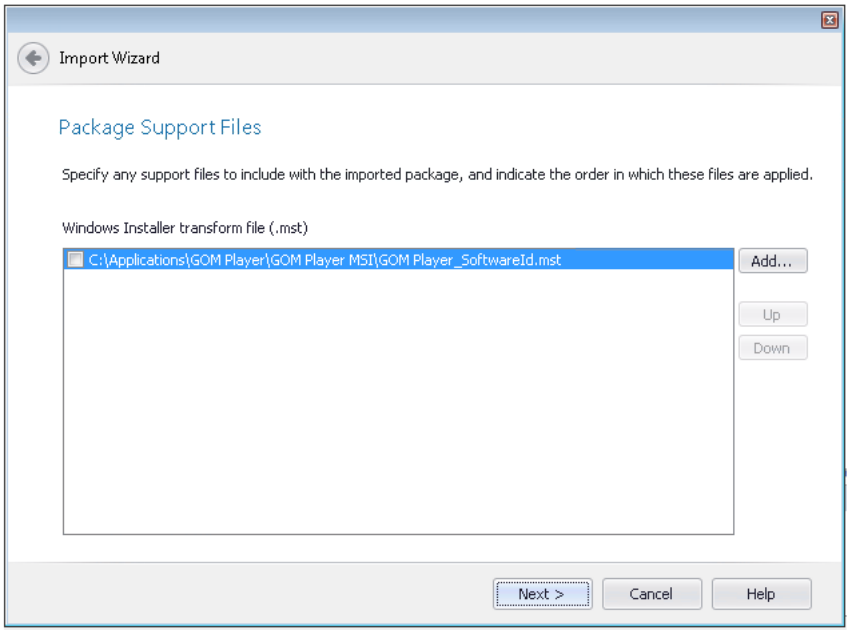


Figure 7-76: Import Wizard / Package Support Files Panel

The following types of support files can be imported:

Table 7-111 • Package Support Files by Package Type

Package Type	Support File	Description
Windows Installer (.msi)	Transform files (.mst)	All of the `.mst` files that are in the same directory as the Windows Installer file you are importing are automatically listed, but only those `.mst` files that AdminStudio determines are probably applicable to this Windows Installer package are selected to be included in the import. If you do not want to import a selected `.mst` file, clear the selection. ***Note*** • *You can add additional transform files and specify the order that they will be applied, as described in Adding Additional Package Support Files and Ordering List.*
	Patch files (.msp)	If a patch file is in the same directory as the Windows Installer file you are importing, that patch file will automatically be listed. If you do not want to import it, clear the selection. ***Note*** • *You can add additional patch files and specify the order that they will be applied, as described in Adding Additional Package Support Files and Ordering List.* ***Note*** • *If you specify an update.exe patch file that was created by Developer/DevStudio/InstallShield Editor, Application Manager will extract the .msp file in the Temp folder and then perform the import.* ***Note*** • *See About the Administrative Installation of Patches.*
Legacy packages (.exe)	Setup configuration files (.ini)	Contains setup and configuration information for a legacy installation package.

Adding Additional Package Support Files and Ordering List

You can import more than one package support file, and specify the order in which the files are applied:

- **To add an additional support file**—Click **Add** and select a file to add to the list.
- **To specify the order of listed support files**—Select a support file in the list and click the **Up** or **Down** button move the file up or down in the list.

About the Administrative Installation of Patches

For patches to be applied to a Windows Installer package, it is necessary to perform an administrative install of the Windows Installer package and then perform an administrative install of each patch package one by one. This way, the content of each patch package is appended to the Windows Installer package at the administrative install location.

In previous releases, when you imported a patch into the Application Catalog, you were prompted to specify a location for an administrative install. However, starting with AdminStudio 2013, you no longer have to specify a location for an administrative install if your Windows Installer package includes patches. Instead, the administrative install operation is automatically performed in a TEMP folder.

Destination Group Panel

On the **Destination Group** panel, select the group into which you want to import the selected package(s).

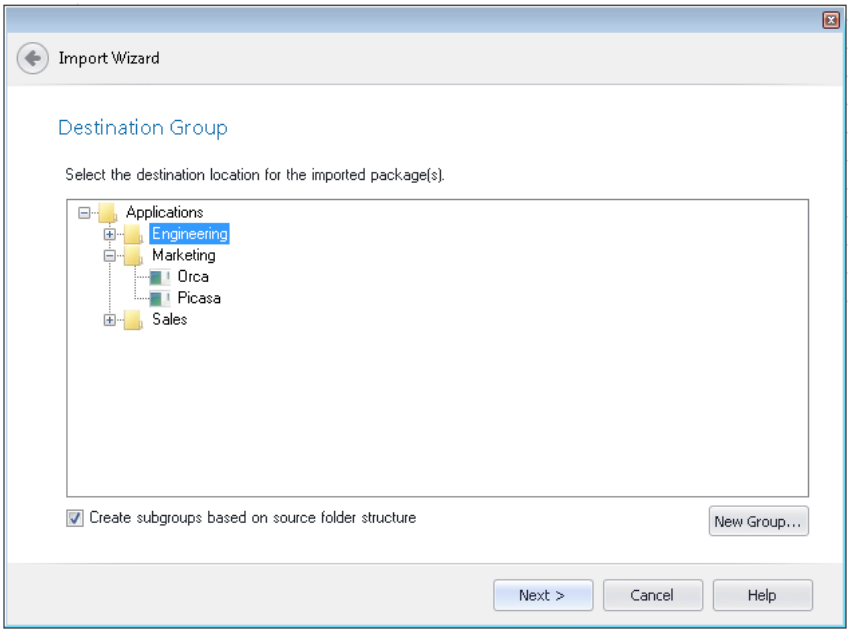


Figure 7-77: Import Wizard / Destination Group Panel

If a group you want to import the package(s) into does not exist, click the **New Group** button to create a new group.

Note • *If you launched the Import Wizard by right-clicking on a group in the tree and then selecting **Import** from the shortcut menu, that group will be selected by default on the **Destination Group** panel.*

Reproducing the Subdirectory Structure of Selected Directory

If you selected the **Folder of multiple applications** option on the **Source** panel, the **Destination Group** panel will have an additional option: **Create subgroups based on source folder structure**. The location of the imported packages in the Application Manager tree depends upon whether this option is selected:

- **Selected**—Subgroups of the selected group will be created in the Application Manager tree that mimic the directory structure of the selected directory, and the packages will be imported into those subgroups.

- **Not selected**—All of the packages in the selected directory (and its subdirectories) will be imported into the root of the selected group.

Selecting an Application Node as a Destination

Application nodes are created in the Application Manager tree using the package's associated Package Code. If multiple packages of different deployment types (such as Windows Installer, App-V, and ThinApp) of the same software product are all imported into the same Group and all have the same Package Code, all of the deployment types will be automatically listed under the same application node.

However, consider the scenario where you are importing a single package file that already has an existing application node in the Application Manager tree (because a package of a different deployment type has already been imported). If you are not sure whether the Package Code of the package you are importing matches that of that application's already imported package, you can choose the desired application node on the **Destination Group** panel to ensure that both packages will be associated with the same application.

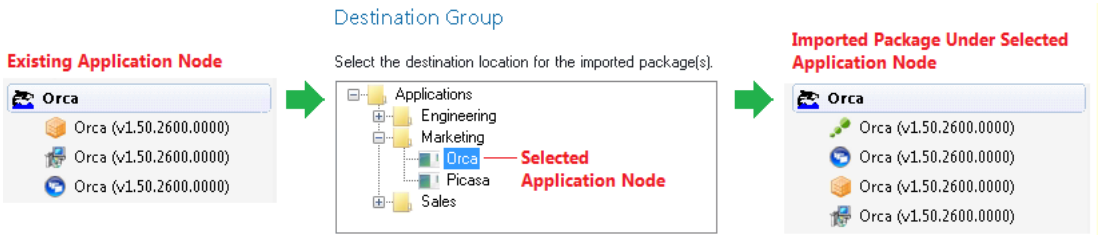


Figure 7-78: Selecting an Application Node as a Destination

Summary Panel

Before executing the import, review the information in the Summary panel about the options selected in the previous panels.

Depending on the import type and how the Import Wizard was invoked, clicking Back returns you to the **Destination Group** panel, **MSM Source Information** panel, **OS Snapshot Information** panel, or **Other Setup Installation Files** panel. Click **Finish** to begin import.

Running the Import Panel

The **Running the Import** panel displays a progress bar and status messages during import. When the import is complete, click **Finish** to close the wizard.

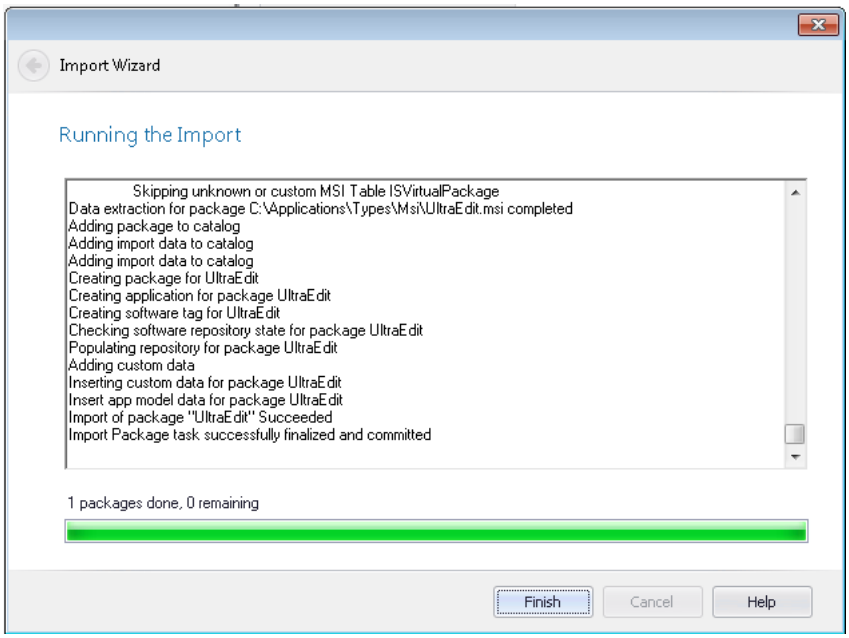


Figure 7-79: Import Wizard / Running the Import Panel

Merge Module Import Wizard

You can use the Merge Module Import Wizard to import multiple merge modules into the Application Catalog at the same time. You can open the Merge Module Import Wizard by first opening the **Merge Modules** tab, and then doing either of the following:

- Click the **Import** button on the ribbon the **Merge Modules** tab.
- Right-click on the **Merge Modules** root group or a merge module in the tree and then select **Import Merge Modules** from the shortcut menu.

The Merge Module Import Wizard consists of the following panels:

- MSM Source Information Panel
- Summary Panel

MSM Source Information Panel

If you are importing a Merge Module, this panel opens, allowing you to import multiple Merge Modules into the Application Catalog at one time.

Click the Browse ( ) button in the **Merge Modules** area and select the merge module file that you want to import. To import multiple patches, you can repeat the procedure as necessary.

The order in which merge modules are applied can be changed by selecting a merge module in the list and clicking the Move Up and Move Down arrows.

If you need to delete a merge module you have added, clear its check box.

Click **Next** to proceed with the import.

Summary Panel

Before executing the import, review the information in the **Summary** panel about the options selected in the previous panels.

Click **Finish** to begin the import of the merge module.

OS Snapshot Wizard

You can use the OS Snapshot Wizard to create a snapshot of your current operating system configuration. You launch the OS Snapshot Wizard from the AdminStudio Tools Gallery or by selecting **OS Snapshot Wizard** from the **AdminStudio Tools** group of the Windows Start menu or from the **Home** tab.

The following topics contain information about each Wizard panel and dialog box available through the OS Snapshot Wizard. The help topics in this reference are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a Wizard or dialog box.

Select one of the following links for OS Snapshot reference information:

- Welcome Panel
- Project Information Panel
- Analyzing Panel
- OS Snapshot Summary Panel
- Analysis Options Dialog Box
- ISSnapshot.ini File

Welcome Panel

You can use the OS Snapshot Wizard to create a snapshot of your current operating system configuration. You launch the OS Snapshot Wizard from the AdminStudio Tools Gallery or by selecting **OS Snapshot Wizard** from the **AdminStudio Tools** group of the Windows Start menu.

The **Welcome** panel appears when you first launch the OS Snapshot Wizard, providing some introductory information about the use of the OS Snapshot Wizard.

The **Next** button advances you to the **Project Information** panel.

Project Information Panel

The Project Information panel gathers information necessary for taking the OS Snapshot.

You must provide the following information before the Start button is enabled, allowing you to take the snapshot.

Table 7-112 • Project Information Panel Options

Option	Description
OS Snapshot project name	Provide a name for the snapshot file (.osc).

Table 7-112 • Project Information Panel Options (cont.)

Option	Description
OS Snapshot project folder	Provide the directory in which snapshot data will be stored. Either enter the path in the field, or click the Browse (...) button to navigate to it. If the directory already exists, a confirmation dialog box opens when you click the Next button.

If you want to review or change current capture settings, click Edit to display the Analysis Options dialog box.

Analyzing Panel

The **Analyzing** panel appears while the OS Snapshot Wizard analyzes your system.

Following the snapshot, the **Summary** panel appears.

OS Snapshot Summary Panel

At the end of the OS Snapshot process, the Summary panel is displayed, containing information about the OS Snapshot that was just performed.

Prior to clicking Finish, review the information to ensure the snapshot contains the data you expected.

Following the OS Snapshot process, you can import the snapshot into the Application Catalog and use it as a baseline to which setups can be compared.

Caution *• OS Snapshots should only be used by Application Manager for comparison purposes. You should never attempt to convert an OS Snapshot into an MSI package.*

Analysis Options Dialog Box

The **Analysis Options** dialog box, accessible by clicking **Edit** from the **Project Information** panel, allows you to specify capture types for the OS snapshots.

You can select the following:

- Files
- INI files
- Shortcuts
- Registry data

Additionally, you can restrict directory analysis to specific directories, which can significantly improve OS Snapshot Wizard performance. Click New to add a directory restriction, edit to modify an existing restriction, or delete to remove a restriction.

Options set in this dialog box apply to the current and subsequent snapshot sessions.

ISSnapshot.ini File

The ISSnapshot.ini file is the default exclusion file for the OS Snapshot Wizard. It contains exclusions to be applied when capturing an OS snapshot, and mainly focuses on specific items that should not be included in applications, such as InstallShield Professional-specific COM settings and OS Snapshot–specific registry entries.

The file is located in the Windows folder, and can be edited using the Exclusions Editor, or using a text editor. See Exclusions Editor Interface.

Note • *It is strongly recommended that you not modify this file, as it increases the likelihood of either inadvertently omitting necessary pieces of the OS snapshot, or including registry entries or files that should not be part of the snapshot.*

Wrap Package Wizard

You can use the Wrap Package Wizard to convert Windows Installer packages (`.msi`) and complex installation packages (`.exe`) to PowerShell wrapped packages (PowerShell script `.ps1` files) using the PowerShell App Deployment Toolkit.

PowerShell is a task automation and configuration management framework from Microsoft. The PowerShell App Deployment Toolkit provides a set of functions to perform common application deployment tasks and to interact with the user during a deployment. By wrapping an existing Windows Installer or complex installation package in a PowerShell script, you are able to automatically perform deployment tasks both before and after installation.

You can convert the following deployment types into a PowerShell wrapped package:

- Windows Installer packages (`.msi`)
- InstallShield InstallScript `.exe` files
- InstallShield Basic MSI installers that are compressed into a `setup.exe` file
- InstallShield Suite Installer `.exe` files
- Wise Package Studio `.exe` files
- Other executable file types that can be uncompressed by 7-ZIP

The Wrap Package Wizard uses the PowerShell template files to perform the conversion. For more information, see PowerShell Script Template Files.

To launch the **Wrap Package Wizard**, right-click on a Windows Installer (`.msi`) or installation package (`.exe`) package in the Application Manager tree and select **Wrap Package** from the menu.

The Wrap Package Wizard consists of the following panels:

- PowerShell Wrapping Options Panel
- Wrapping Package Panel

PowerShell Wrapping Options Panel

On the **PowerShell Wrapping Options** panel of the Wrap Package Wizard, you need to specify the **Output Directory**, the location where the PowerShell-wrapped package will be stored.

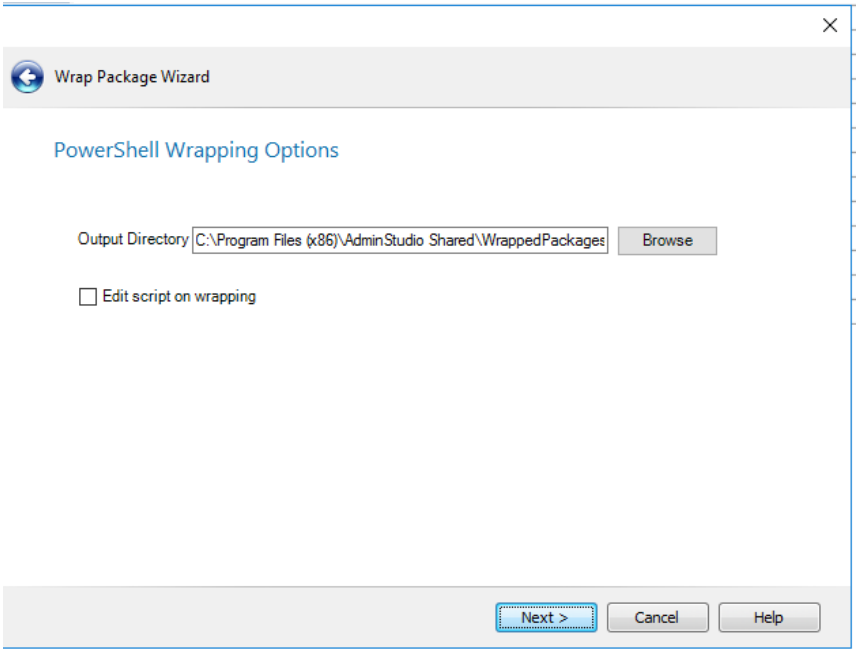


Figure 7-80: PowerShell Wrapping Options Panel of Wrap Package Wizard

If you want to edit the default PowerShell script file in Windows PowerShell ISE after the package has been wrapped, select **Edit script on wrapping** option.

Click **Next** to proceed.

Wrapping Package Panel

After you initiate the wrapping process by clicking **Next** on the **PowerShell Wrapping Options** panel, the **Wrapping Package** panel opens and wrapping is initiated. When the process is complete, confirmation messages are listed.

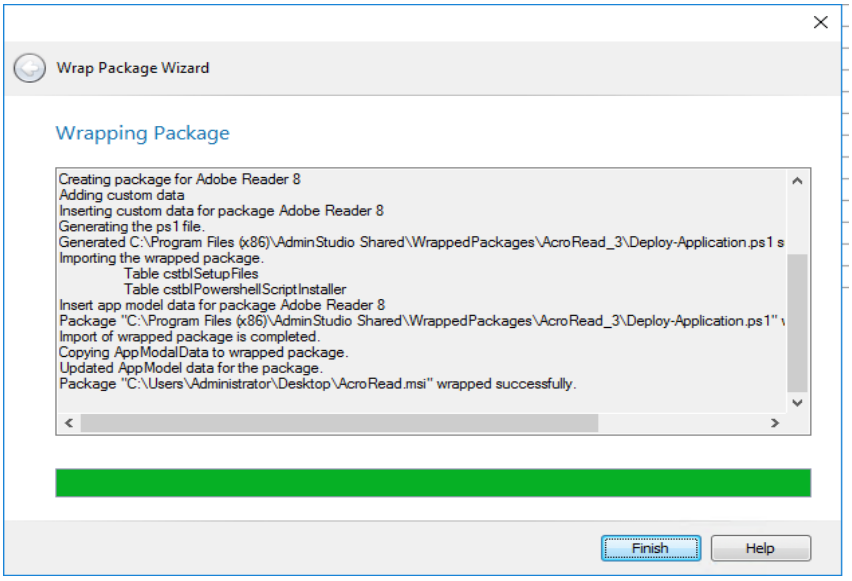


Figure 7-81: Wrapping Package Panel of Wrap Package Wizard

Click **Finish** to close the wizard.

If you selected the **Edit script on wrapping** option on the **PowerShell Wrapping Options** panel, the Windows PowerShell ISE application opens and you are prompted to edit the PowerShell script file that you just created, as described in Editing a PowerShell-Wrapped Package

In the Application Manager tree, the `.msi` or `.exe` deployment type icon for the package you just wrapped has been replaced with a PowerShell deployment type icon.

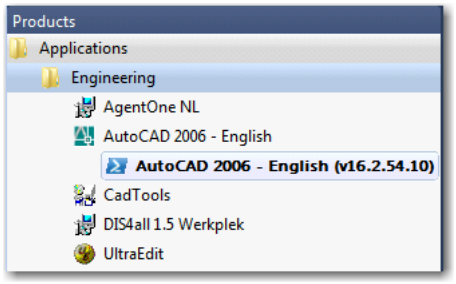


Figure 7-82: PowerShell Deployment Type Icon

Important • *All of the OS compatibility, application virtualization compatibility and best practices ICE tests results that were generated by testing the package before it was wrapped are not carried over and are not viewable on* ***Analyze*** *tab when you select its new PowerShell deployment type.*

Note • *Log files for the Wrap Package Wizard can be found at:* `C:\Windows\Logs\Software`.

Requirement Wizard

The **Requirements** subtab of the **Deployment Data** tab lists user or device requirements that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package. You use the **Requirement Wizard** to add items to this list.

The **Requirement Wizard** is opened by clicking the **Add Requirement** or **Edit Requirement** buttons in the ribbon of the **Requirements** subtab.

The **Requirement Wizard** includes the following panels:

- Welcome Panel
- Create Custom Requirements Panel
- Create User Requirements Panel
- Select the Device Requirements Type Panel
- Configuration Manager Credentials Panel
- Device Requirements from Configuration Manager Panel
- Create Device Requirements Panel
- Summary Panel

Welcome Panel

On the **Welcome** panel, select the requirement type that you would like to use.

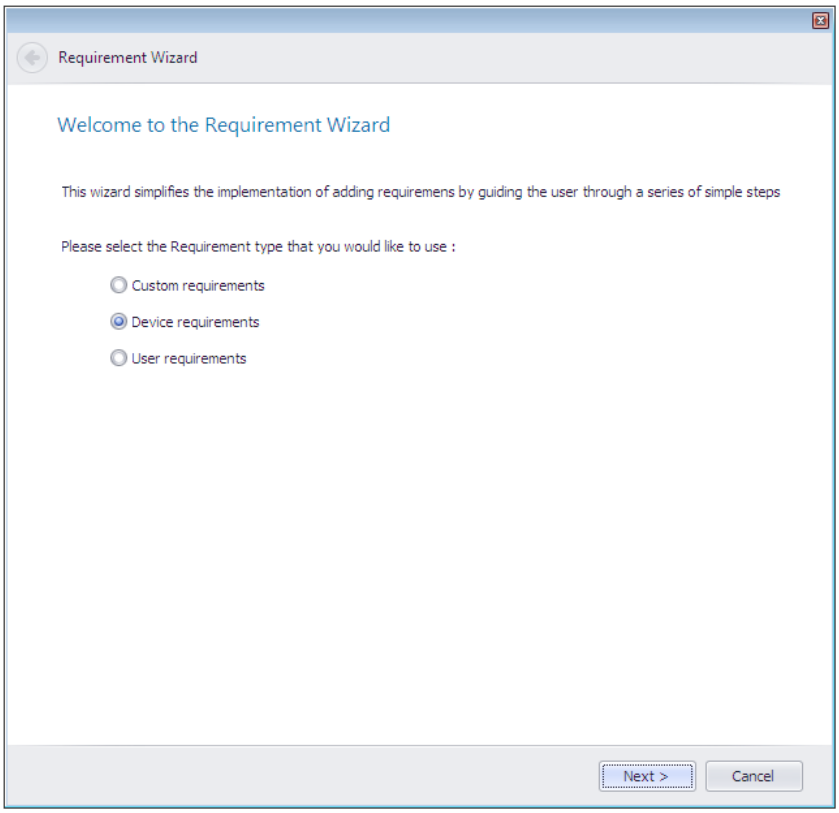


Figure 7-83: Requirement Wizard / Welcome Panel

The **Welcome** panel includes the following properties:

Table 7-113 • Requirement Wizard / Welcome Panel

Property	Description
Custom requirements	Select this option if you want to define a custom requirement that the target system needs to be meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package.
Device requirements	Select this option if you want to define a device requirement (such as CPU speed, free disk space, operating system, etc.) that the target system needs to be meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package.
User requirements	Select this option if you want to select a ConfigMgr (Formerly called as System Center Configuration Manager)-defined user global condition that the target system needs to meet in order for ConfigMgr (Formerly called as System Center Configuration Manager) to be able to successfully deploy this package.

Create Custom Requirements Panel

On the **Create Custom Requirements** panel, which appears if you select **Custom requirements** on the **Welcome** panel, you specify the custom conditions for the requirement.

When you initially view the **Create Custom Requirements** panel, you are prompted to either select an existing condition and click **Create** to open the Create Global Condition Dialog Box and create a new condition. Once you create a condition, it is available for selection when creating other **Requirements**.

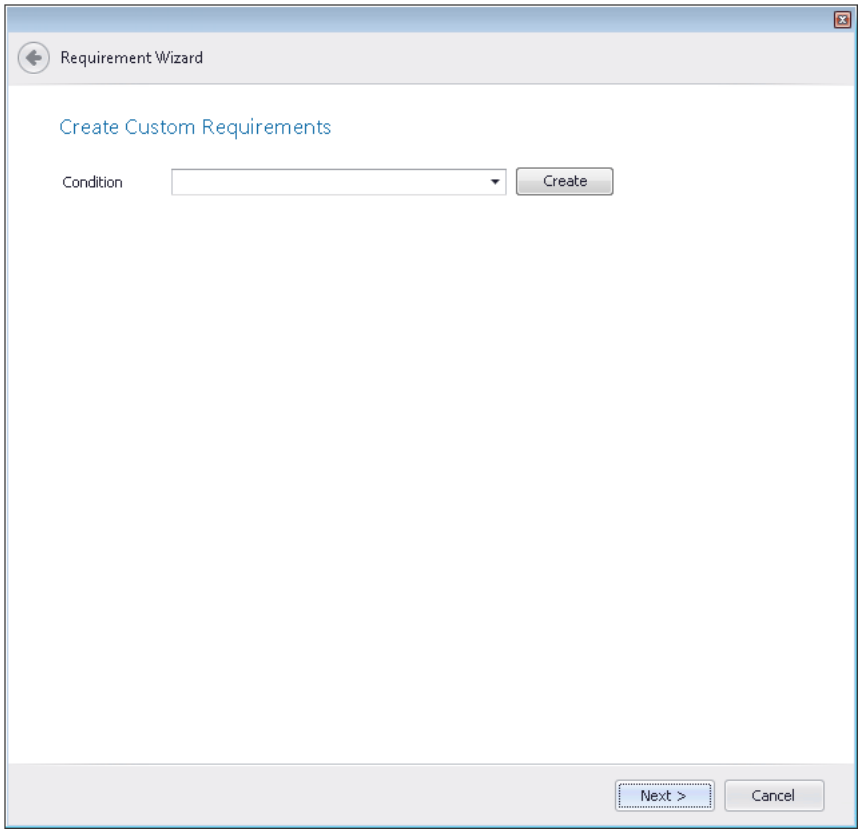


Figure 7-84: Create Custom Requirements Panel / Initial View

The fields displayed on the **Create Custom Requirements** panel depend upon the **Setting Type** of the selected global condition.

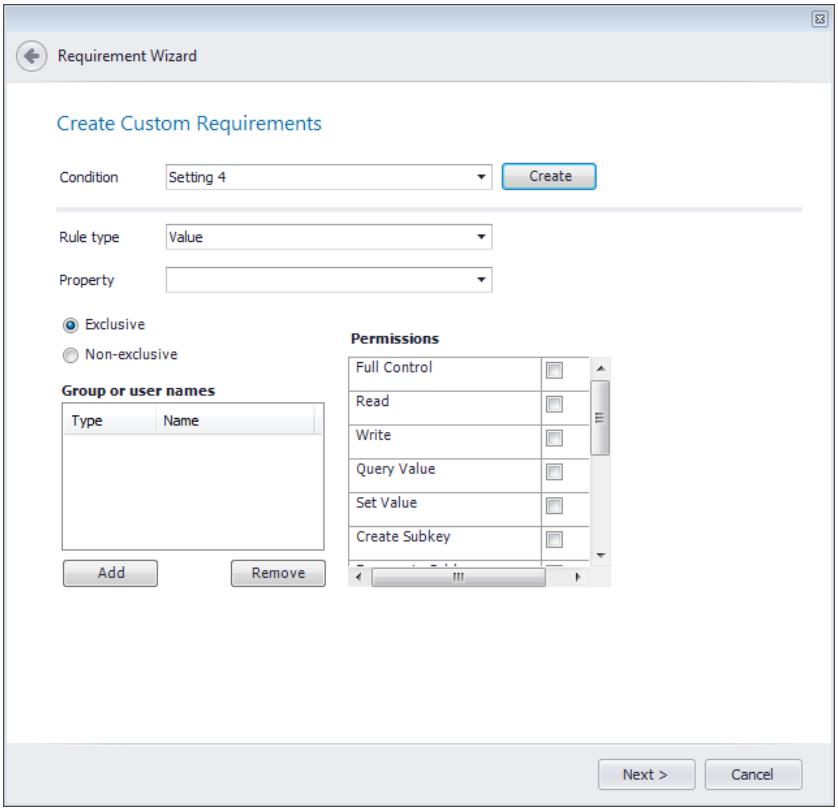


Figure 7-85: Requirement Wizard / Create Custom Requirements Panel - Setting Type of Registry Key

The **Create Custom Requirements** panel includes the following properties:

Table 7-114 • Requirement Wizard / Create Custom Requirements Panel

Property	Description
Condition	Select a condition from the list or click the **Create** button to open the Create Global Condition Dialog Box and create a condition. ***Note*** *• See Create Global Condition Dialog Box for information on using this dialog box to create a global condition.*
Rule Type	Select one of the following options: • **Value**—Select to create a condition that searches for a defined condition meeting a specific value. If you select this option, additional fields are displayed on the **Create Custom Requirements** panel that you can use to identify the specific value. • **Existential**—Select to create a condition that searches for the existence of a defined condition. If you select this option, the **The selected global condition must exist on client devices** option appears.

Table 7-114 • Requirement Wizard / Create Custom Requirements Panel

Property	Description
Property	Select the **Property** of the **File system** or **Registry key** condition that you want to use to create the requirement. ***Note*** *• Only displayed when the selected global condition's* ***Setting Type*** *is set to either* ***File system*** *or* ***Registry key****.*
Operator	Select an operator to use in this custom requirement. Available options are: **Between**, **Greater than or equal to**, **Greater than**, **Equals**, **Less equals**, **Less than**, **None of**, **Not equal to**, or **One of**. ***Note*** *• Only displayed when selected global condition's* ***Setting Type*** *is set to* ***File system****,* ***IIS metabase****,* ***Registry value****,* ***Script****, or* ***SQL query****,* ***Wql query****, or* ***Xpath query****.*
Value	Select the value of the selected **Property** that you want to use in this requirement. ***Note*** *• Displayed for global conditions of all* ***Setting Types*** *except for* ***Registry*** *key, but only when* ***Rule type*** *is set to* ***Value*** *(not* ***Existential****).*
Exclusive / Non-exclusive	Specify exclusivity option. ***Note*** *• Only displayed for global conditions with a* ***Setting Type*** *of* ***Registry key*** *that also have a* ***Rule type*** *of* ***Value****.*
Group or user names	Click **Add** to add users or groups to this list. On the **Enter User** or **Group Name** dialog box, you enter the name of the user or group using the format `Domain\User` or `Domain\Group`, and specify whether you want to **Allow** or **Deny** access to this user or group. ***Note*** *• Only displayed for global conditions with a* ***Setting Type*** *of* ***Registry key*** *that also have a* ***Rule type*** *of* ***Value****.*
Permissions	Select the permissions in this list to **Allow** or **Deny** that permission to the selected user or group. ***Note*** *• Only displayed for global conditions with a* ***Setting Type*** *of* ***Registry key*** *that also have a* ***Rule type*** *of* ***Value****.*

Create User Requirements Panel

On the **Create User Requirements** panel, you use the **Condition**, **Rule Type**, **Operator**, and **Value** fields to build a user requirement.

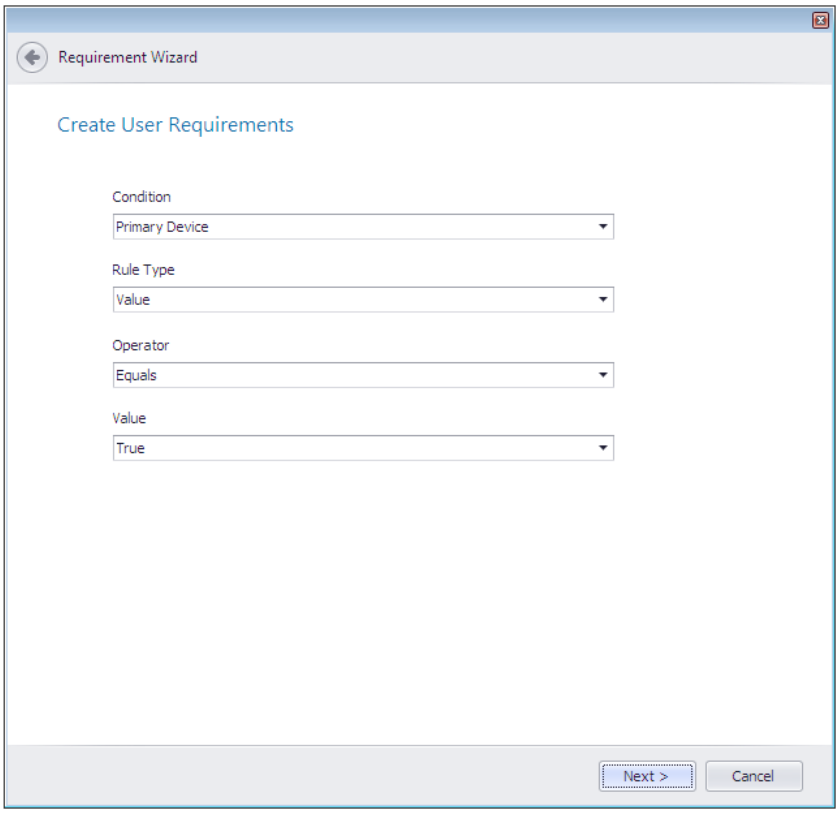


Figure 7-86: Requirement Wizard / Create User Requirements Panel

The **Create User Requirements** panel includes the following properties:

Table 7-115 • Requirement Wizard / Create User Requirements Panel

Property	Description
Condition	Select a condition type from the list. For user requirements, **Primary Device** is the only condition type listed.
Rule Type	Select a rule type from the list. For custom device requirements, **Value** is the only type listed.
Operator	Select a rule type from the list. For user requirements, **Equals** is the only operator listed.
Value	Select either **True** or **False** to define this user requirement.

Select the Device Requirements Type Panel

On the **Select the Device Requirements Type** panel, specify whether you want to create a custom device requirement or select a device requirement from Configuration Manager.

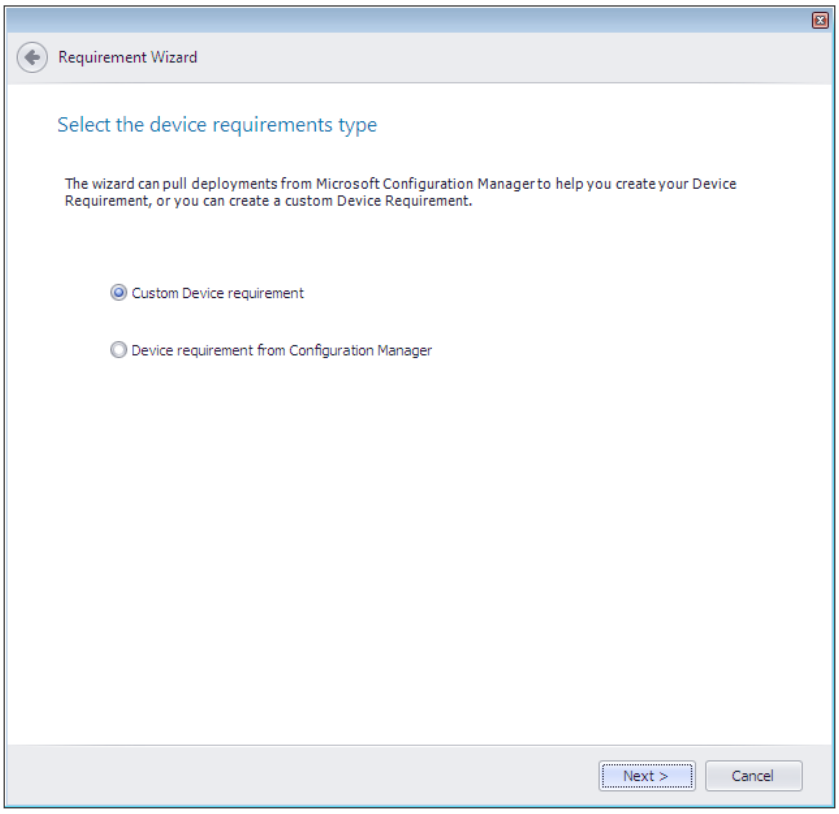


Figure 7-87: Requirement Wizard / Select the Device Requirements Type Panel

The **Select the Device Requirements Type** panel includes the following properties:

Table 7-116 • Requirement Wizard / Select the Device Requirements Type Panel

Property	Description
Custom Device requirement	Select to create your own device requirement.
Device requirement from ConfigMgr	Select to use a device requirement that was defined in ConfigMgr (Formerly called as System Center Configuration Manager).

Configuration Manager Credentials Panel

On the **Configuration Manager Credentials** panel, you enter connection information for ConfigMgr (Formerly called as System Center Configuration Manager).

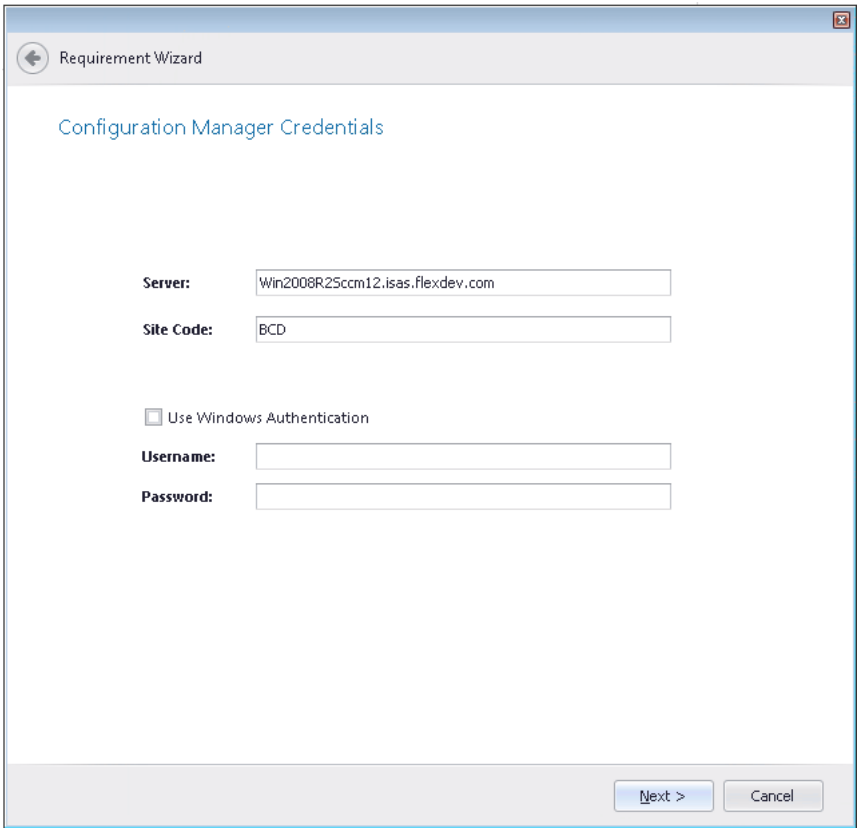


Figure 7-88: Requirement Wizard / Configuration Manager Credentials Panel

The **Configuration Manager Credentials** panel includes the following properties:

Table 7-117 • Requirement Wizard / Configuration Manager Credentials Panel

Property	Description
Server	Enter the name of the Configuration Manager 2012 Server you want to connect to. This field is pre-populated with the name of the Configuration Manager System Center server that you have entered on the Distribution System tab of the Options dialog box.
Site Code	Enter the code that identifies the Configuration Manager site you want to connect to.
Use Windows Authentication	Select this option if you want to use Windows network authentication (your network login ID) to log into this ConfigMgr (Formerly called as System Center Configuration Manager) Server.
Username and Password	If using server authentication, enter the **Username** and **Password** of that server.

Device Requirements from Configuration Manager Panel

The **Device Requirements from Configuration Manager** panel lists those applications in the System Center 2012 Configuration Manager server that have defined device requirements. Select the application in the list that matches the one that you are editing, and click **Next** to continue.

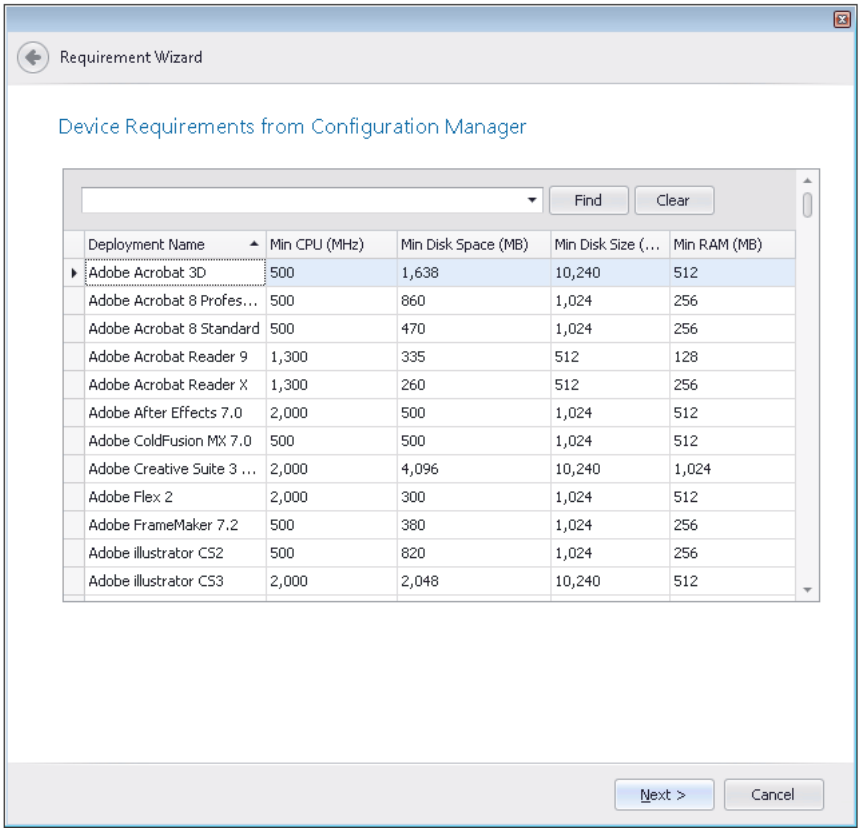


Figure 7-89: Requirement Wizard / Device Requirements from Configuration Manager Panel

The **Device Requirements from Configuration Manager** panel includes the following properties:

Table 7-118 • Requirement Wizard / Device Requirements from Configuration Manager Panel

Property	Description
Search box	Use to filter the list of device requirements.
Deployment Name	Name application associated with the device requirement.
Min CPU (MHz)	Defined minimum CPU speed requirement, in MHz.
Min Disk Space (MB)	Defined minimum disk space requirement, in MB.
Min Disk Size (MB)	Defined minimum disk size requirement, in MB.
Min RAM (MB)	Defined minimum RAM requirement, in MB.

Create Device Requirements Panel

On the **Create Device Requirements** panel, you specify a device requirement.

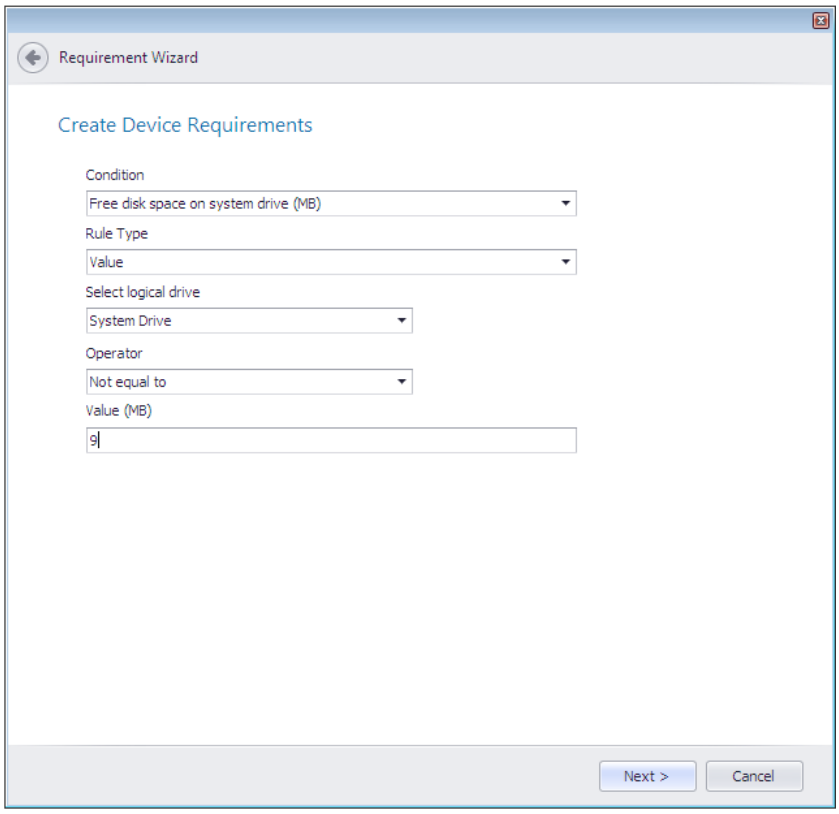


Figure 7-90: Requirement Wizard / Create Device Requirements Panel

The **Create Device Requirements** panel includes the following properties:

Table 7-119 • Requirement Wizard / Create Device Requirements Panel

Property	Description
Condition	Select one of the following conditions: • **Active Directory Site** • **Co-Managed device** • **Configuration Manager Site** • **CPU Speed (MHz)** • **Disk space** • **Intune O365 Pro Plus management** • **Number of processors** • **Operating system** • **Operating system language** • **Organizational unit (OU)** • **Total physical memory (MB)** • **Windows Store inactive**
Rule Type	Select a rule type from the list. For custom device requirements, **Value** is the only type listed.
Operator	Select an operator from the list. Possible sets of operators are: • **One of** or **None of** • **Equals**, **Not equal to**, **Greater than**, **Less than**, **Between**, **Greater than or Equal to**, or **Less than or equal to**

Table 7-119 • Requirement Wizard / Create Device Requirements Panel

Property	Description
[Additional Fields]	Additional fields are displayed depending upon the **Condition** selected. Use these fields to define the requirement for the selected **Condition**. • **Active Directory Site**—Click the **Add** button and add a site to the **Active Directory Sites** list. • **Co-Managed Device** - Click the **Add** button and specify a valid operator and value. • **Configuration Manager Site**—Click the **Add** button and add a site to the **Configuration Manager Sites** list. • **CPU Speed (MHz)**—Enter a value, in MHz, in the **Value (MHz)** text field. • **Disk space**—Select a drive from the **Select logical drive** list and enter a value, in MBs, in the **Value (MB)** text box. • **InTune O365 Pro Plus Management** - Click the **Add** button and specify a valid operator and value. • **Number of processors**—Enter a number in the **Value** text box. • **Operating system**—Select operating systems from the **Select Operating System** list. You can choose just a major category (such as **Windows 8** or **Windows Server 2012**) or you can identify a specific operating system / service pack / processor type combination, such as **All Windows 8 (32-bit)**. • **Operating system language**—Select languages from the **Select Operating System Language(s)** list. • **Organizational unit (OU)**—Click the **Add** button and add a OU to the list. • **Total physical memory (MB)**—Enter a value, in MBs in the **Value (MB)** text box. • **Windows Store inactive**—Enter a value in the **Value** text box.

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the requirement to the list.

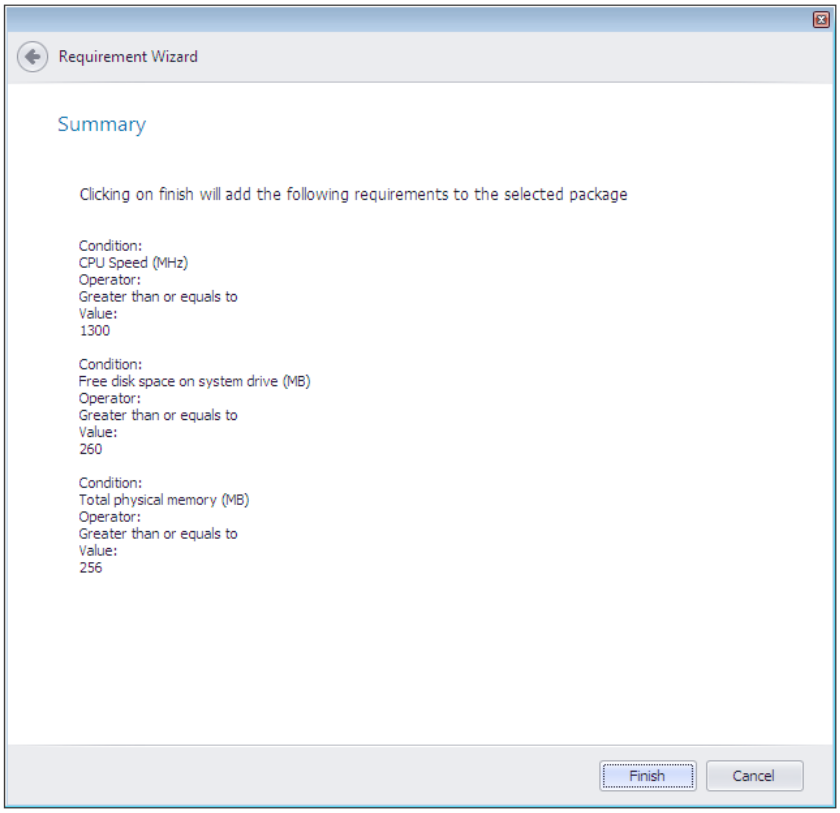


Figure 7-91: Requirement Wizard / Summary Panel

Requirement Wizard of Intune Deployment Data Tab

The **Requirements** subtab of the **Intune Deployment Data** tab lists device or additional requirements that able to successfully deploy this package. You use the **Requirement Wizard** to add items to this list.

The **Requirement Wizard** is opened by clicking the **Add Requirement** or **Edit Requirement** buttons in the ribbon of the **Requirements** subtab.

The **Requirement Wizard** includes the following panels:

- Welcome Panel
- Device Requirements Panel
- Additional Requirements Panel
- Summary Panel

Welcome Panel

On the Welcome panel, select the requirement type that you would like to use.

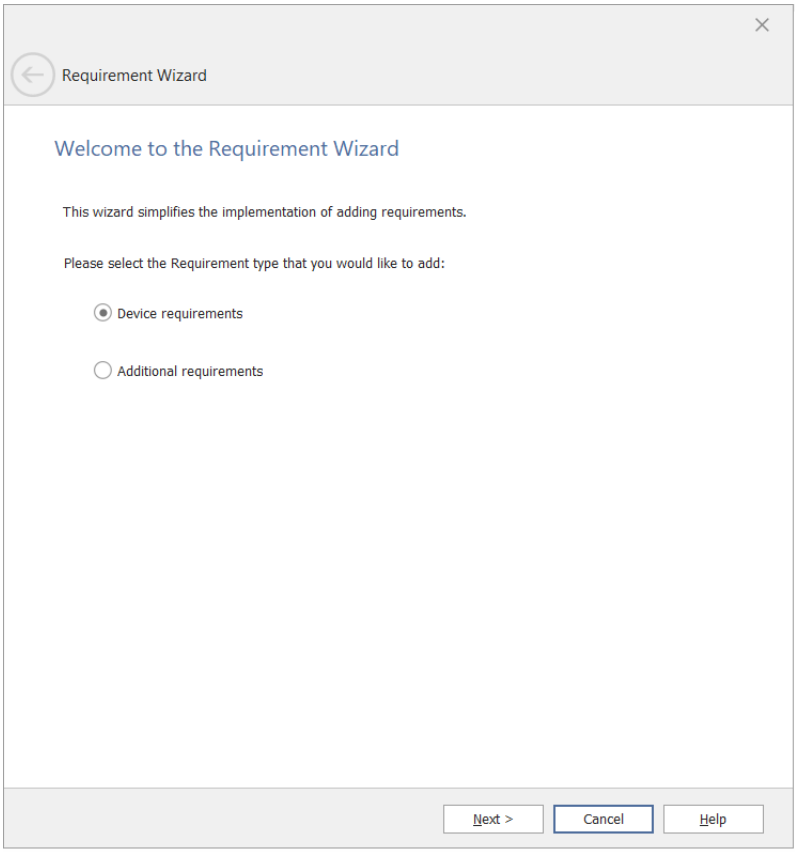


Figure 7-92: Requirement Wizard / Welcome Panel

The Welcome panel includes the following properties:

Table 7-120 • Requirement Wizard / Welcome Panel

Property	Description
Device requirements	Select this option if you want to define a device requirement (such as CPU speed, free disk space, operating system, etc.) that the system needs to successfully deploy this package.
Additional requirements	Select this option if you want to define a additional requirement that the system needs to successfully deploy this package.

Device Requirements Panel

On the **Device Requirements** panel, you specify a device requirement.

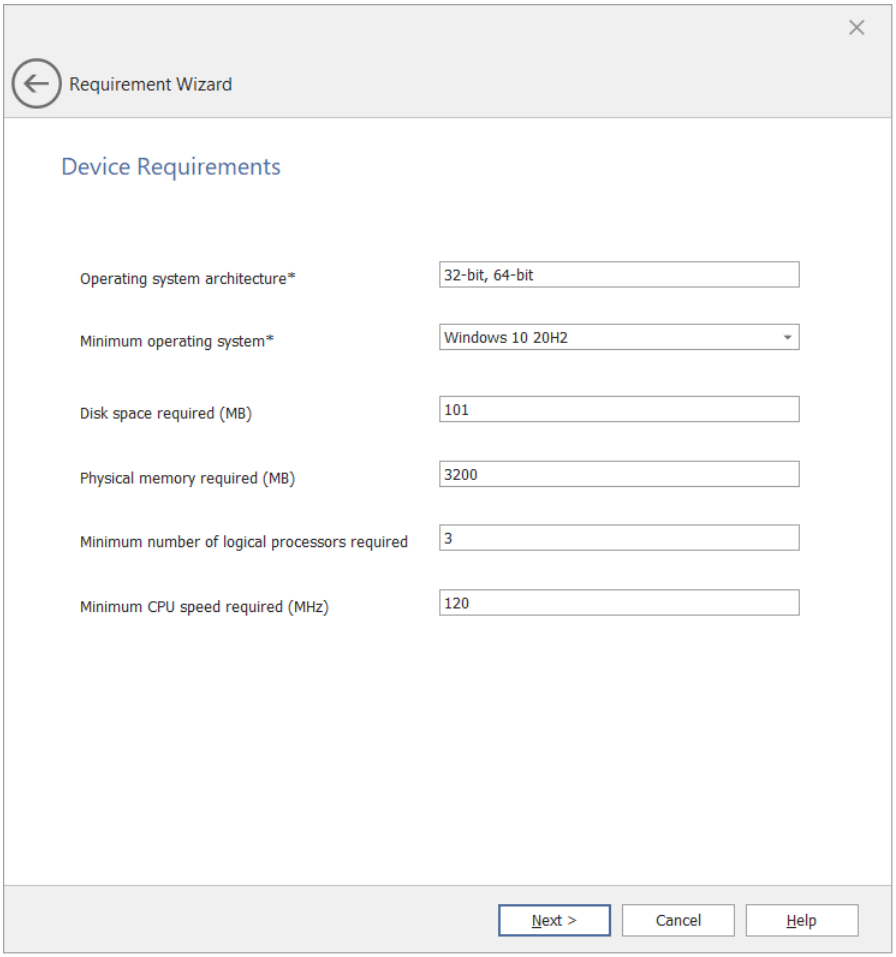


Figure 7-93: Requirement Wizard / Device Requirements Panel

The Device Requirements panel includes the following properties:

Table 7-121 • Requirement Wizard / Device Requirement Panel

Property	Description
Operating system architecture	Choose either 32 bit or 64 bit or both from the **Operating system architecture** list.
Minimum operating system	Select operating system from the **Minimum operating system** list.
Disk space required (MB)	Enter free disk space value in MBs, in the text field.
Physical memory required (MB)	Enter physical memory (RAM) space in MBs, in the text field.
Minimum number of logical processors required	Enter a number in the text field.
Minimum CPU speed required (MHz)	Enter a value in MHz, in the text field.

Additional Requirements Panel

On the Additional Requirements panel, select and specify the requirement type.

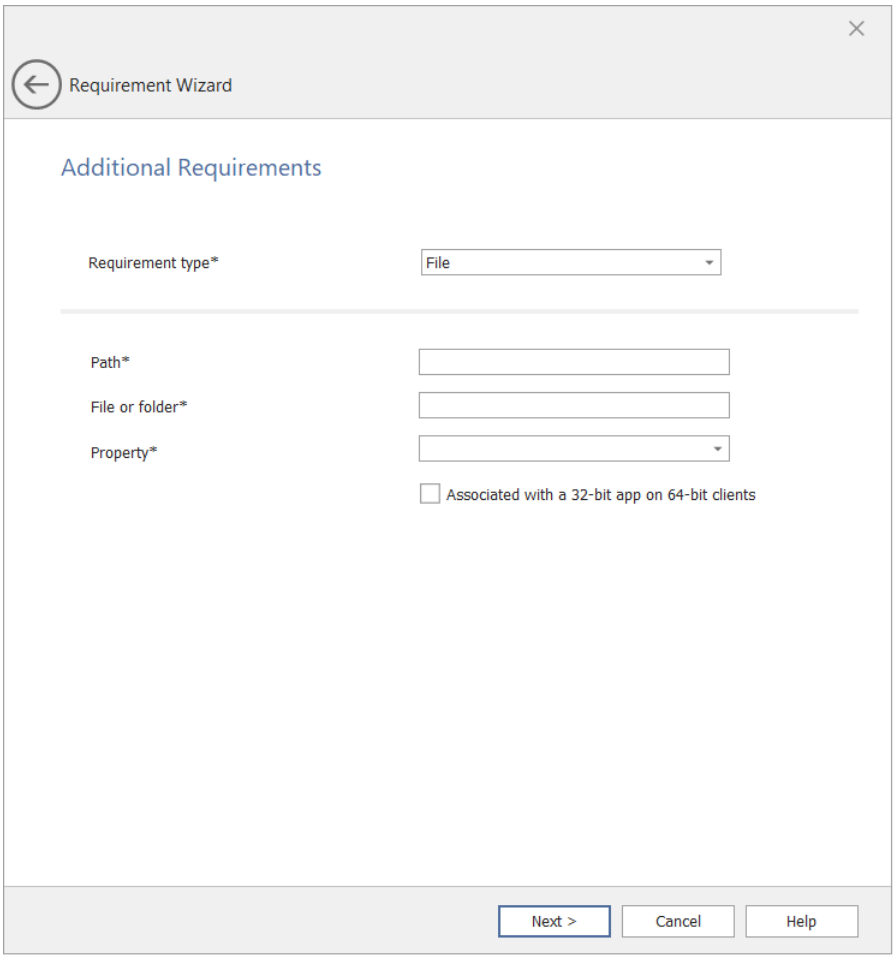


Figure 7-94: Requirement Wizard / Additional Requirements Panel

The Additional Requirements panel includes the following properties:

Table 7-122 • Requirement Wizard / Additional Requirement Panel

Property	Description
Requirement type	Select one of the following Requirement type: • File • Registry • Script
Path	Enter the path of the selected File.
File or folder	Enter the name of the file or folder that you are using in this method.

Table 7-122 • Requirement Wizard / Additional Requirement Panel

Property	Description
Property	Select one of the following Property: • File or folder exists • Date modified • Date created • String (version) • Size in MB
Key Path	Enter the path of the selected **Registry** type.
Value Name	Enter the register value in the text box.
Registry Key Requirement	Select one of the following Registry Key Requirement: • Key exists • Key does not exits • String comparison • Version comparison • Integer comparison
Script name	Enter name of the script.
Script file	Enter or pasting the script code directly into the **Script file** text box. You can also click the **Browse** button and select the script that you want to add for this additional requirement. ***Note*** *• If the application is detected, the requirement process will provide a **0** value exit code and will write a string value to STDOUT.*
Select output data type	Select one of the following output data type Key: • String • Date and Time • Integer • Floating Point • Version • Boolean

Table 7-122 • Requirement Wizard / Additional Requirement Panel

Property	Description
Operator	Select one of the following operators: • Equals • Not equal to • Greater than or equal to • Greater than • Less than or equal to • Less than
Value	Enter the value to define the condition.
Associated with a 32-bit app on 64-bit clients	Select this option to expand any path environment variables in the 32-bit context on 64-bit clients. Unselect this option to expand any path variables in the 64-bit context on 64-bit clients. 32-bit clients will always use the 32-bit context. By default this option will be unselected. ***Note*** *• For the Registry Requirement Type:* • *Select this option to search the 32-bit registry on 64-bit clients.* • *Unselect this option to search the 64-bit registry on 64-bit clients. 32-bit clients will always search the 32-bit registry. By default this option will be unselected.*
Run script as 32-bit process on 64-bit clients	Select this option to run the script in a 32-bit process on 64-bit clients. Unselect this option to run the script in a 64-bit process on 64-bit clients. 32-bit clients run the script in a 32-bit process. By default this option will be unselected.
Run this script using the logged on credentials	Run this script using the signed in device credentials.
Enforce script signature check	Select this option to verify that the script is signed by a trusted publisher, which will allow the script to run without warnings or prompts. The script will run unblocked. Unselect this option to run the script with end-user confirmation, but without signature verification.

Summary Panel

On the Summary panel, a summary of your selections is listed. Click **Finish** to add the requirement to the list.

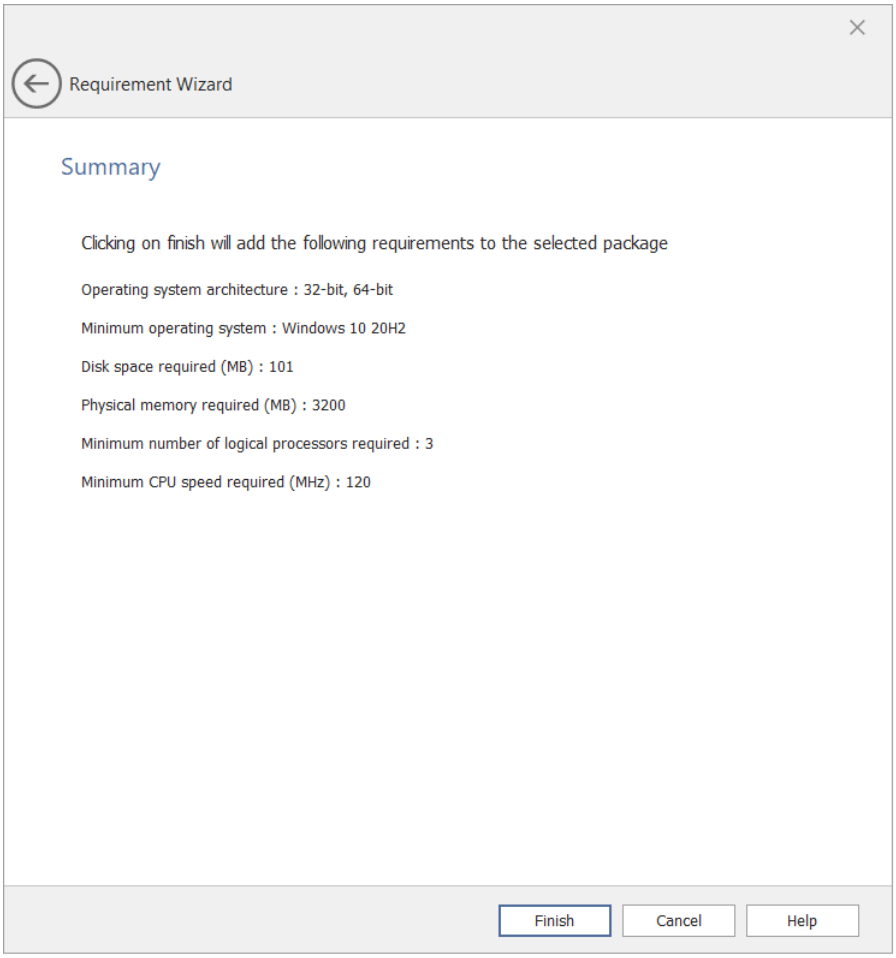


Figure 7-95: Requirement Wizard / Summary Panel

Return Code Wizard of Intune Deployment Data Tab

The **Return Codes** subtab of the **Intune Deployment Data** tab lists code type that able to successfully deploy this package. You use the **Requirement Wizard** to add items to this list.

The **Return code Wizard** is opened by clicking the **Add Requirement** or **Edit Requirement** buttons in the ribbon of the **Return Codes** subtab.

The **Return code Wizard** includes the following panels:

- Welcome Panel
- Summary Panel

Welcome Panel

On the Welcome panel, specify the return code and select the code type.

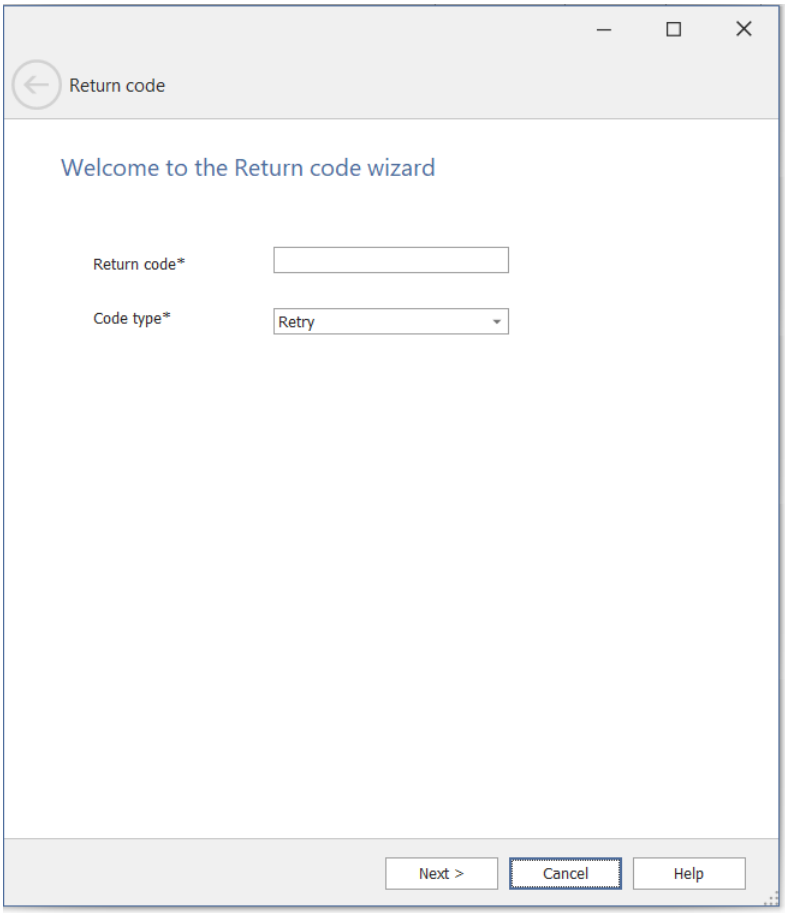


Figure 7-96: Return Code Wizard / Welcome Panel

The Welcome panel includes the following properties:

Table 7-123 • Return Code Wizard / Welcome Panel

Property	Description
Return code	Enter the valid return code in the text box.
Code type	Select one of the following code type: • Success • Soft Reboot • Hard Reboot • Retry • Failed

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the to the list.

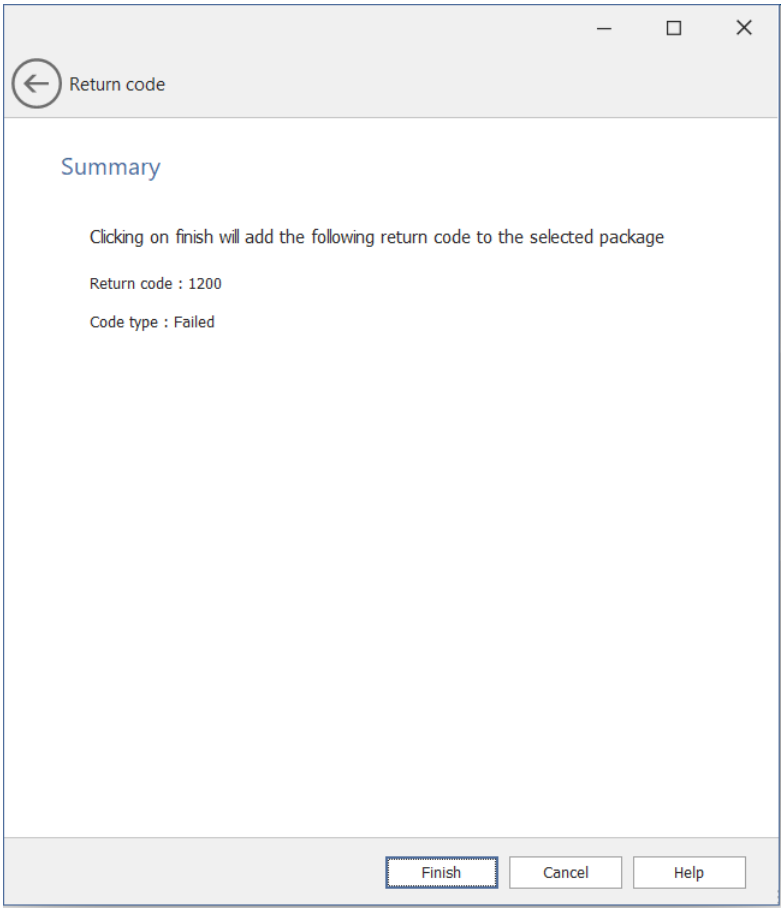


Figure 7-97: Return Code Wizard / Summary Panel

Supersedence Wizard

The **Supersedence** subtab of the **Deployment Data** tab lists other packages that this package would supersede if installed on the same target machine (meaning that the package on the target system would need to be uninstalled prior to installing this package). You use the **Supersedence Wizard** to add items to this list.

The **Supersedence Wizard** is opened by clicking the **Add Supersedence** or **Edit Supersedence** buttons in the ribbon of the **Supersedence** subtab.

The **Supersedence Wizard** includes the following panels:

- Welcome Panel
- Deployment Types in Application Catalog Panel
- Configuration Manager Credentials Panel
- Deployment Types in Configuration Manager 2012 Panel
- Summary Panel

Welcome Panel

On the **Welcome** panel, select the supersedence method you want to use.

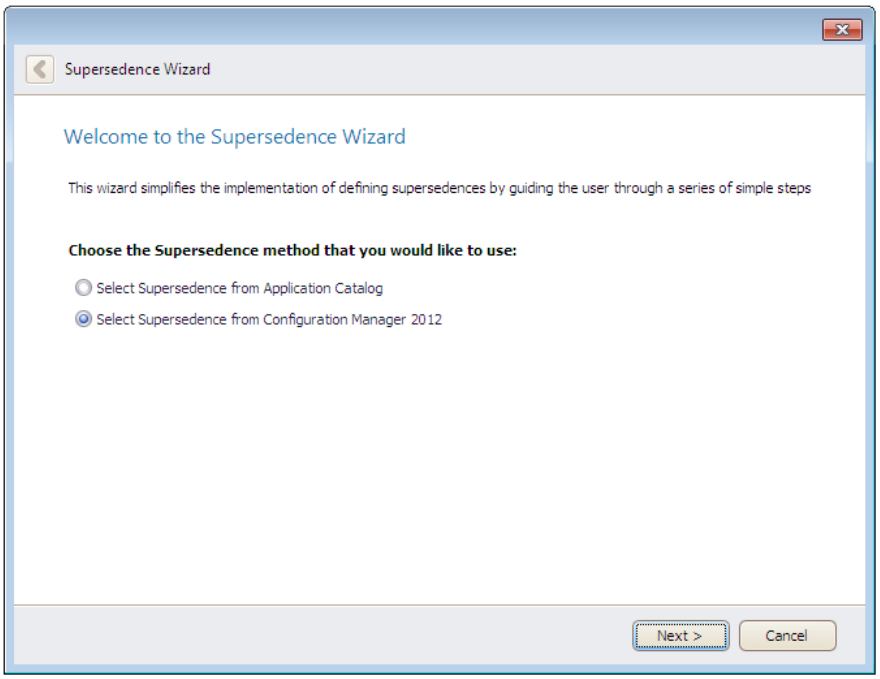


Figure 7-98: Supersedence Wizard / Welcome Panel

The **Welcome** panel includes the following properties:

Table 7-124 • Supersedence Wizard / Welcome Panel

Property	Description
Select Supersedence from Application Catalog	Select to select a supersedent application from the Application Catalog.
Select Supersedence from Configuration Manager 2012	Select to select a supersedent application from Configuration Manager 2012.

Deployment Types in Application Catalog Panel

On the **Deployment Types in Application Catalog** panel, select the dependent supersedence(s) from the list.

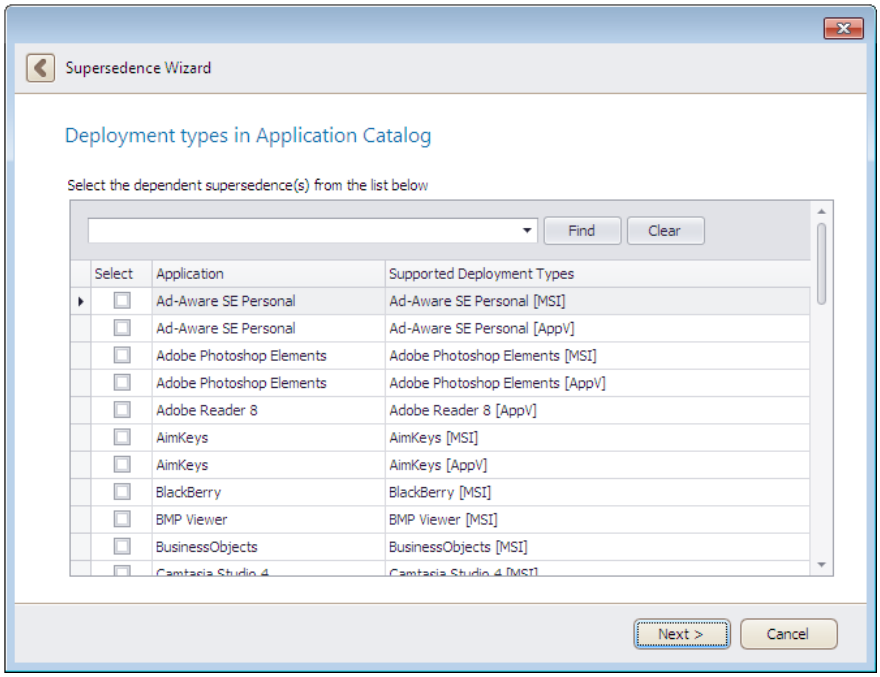


Figure 7-99: Supersedence Wizard / Deployment Types in Application Catalog Panel

The **Deployment Types in Application Catalog** panel includes the following properties:

Table 7-125 • Supersedence Wizard / Deployment Types in Application Catalog Panel

Property	Description
Find / Clear	Use to filter application list.
Application	Application name.
Supported Deployment Types	Lists all of the application's supported deployment types.

Configuration Manager Credentials Panel

On the **Configuration Manager Credentials** panel, specify Configuration Manager connection credentials.

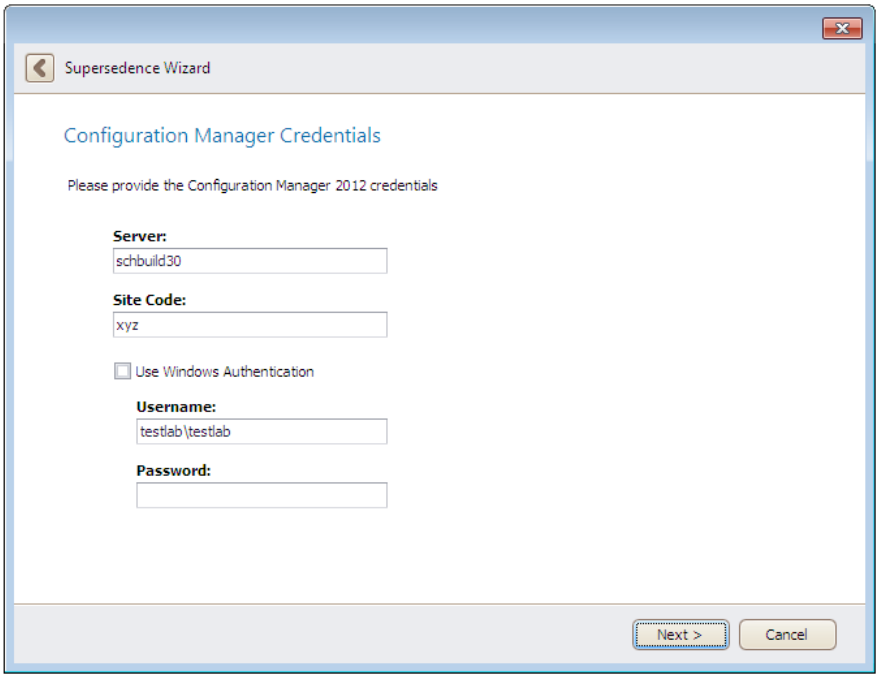


Figure 7-100: Supersedence Wizard / Configuration Manager Credentials

The **Configuration Manager Credentials** panel includes the following properties:

Table 7-126 • Supersedence Wizard / Configuration Manager Credentials Panel

Property	Description
Server	Enter the name of the Configuration Manager Server you want to connect to.
Site Code	Enter the code that identifies the Configuration Manager site you want to connect to.
Use Windows Authentication	Select this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server.
Username and Password	If using server authentication, enter the **Username** and **Password** of that server.

Deployment Types in Configuration Manager 2012 Panel

On the **Deployment Types in Configuration Manager 2012** panel, select dependent supersedences from the list of applications from Configuration Manager 2012.

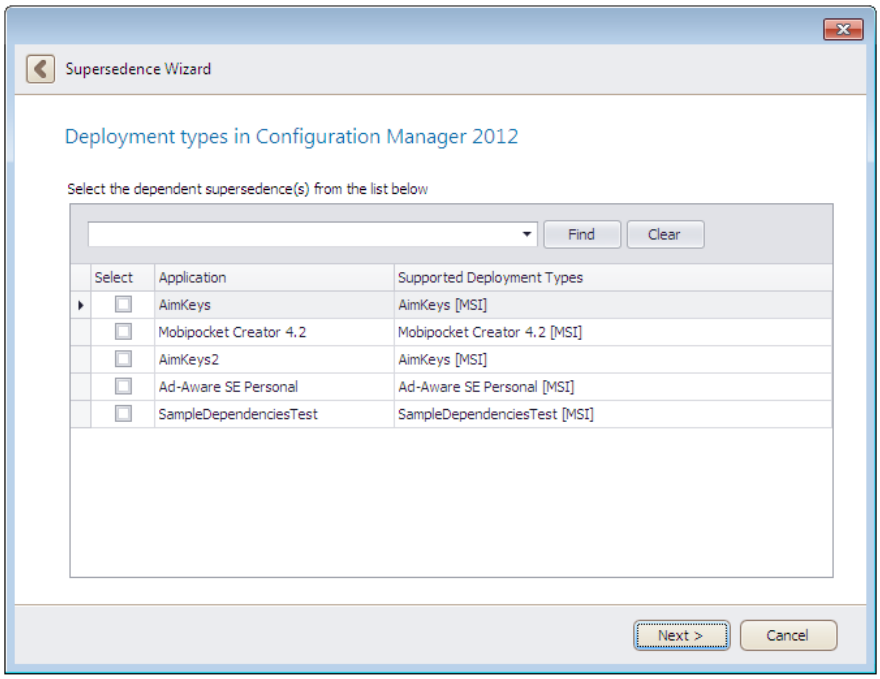


Figure 7-101: Supersedence Wizard / Deployment Types in Configuration Manager 2012 Panel

The **Deployment Types in Configuration Manager 2012** panel includes the following properties:

Table 7-127 • Supersedence Wizard / Deployment Types in Configuration Manager 2012 Panel

Property	Description
Find / Clear	Use to filter application list.
Application	Application name.
Supported Deployment Types	Lists all of the application's supported deployment types.

Summary Panel

On the **Summary** panel, a summary of your selections is listed. Click **Finish** to add the supersedences to the list.

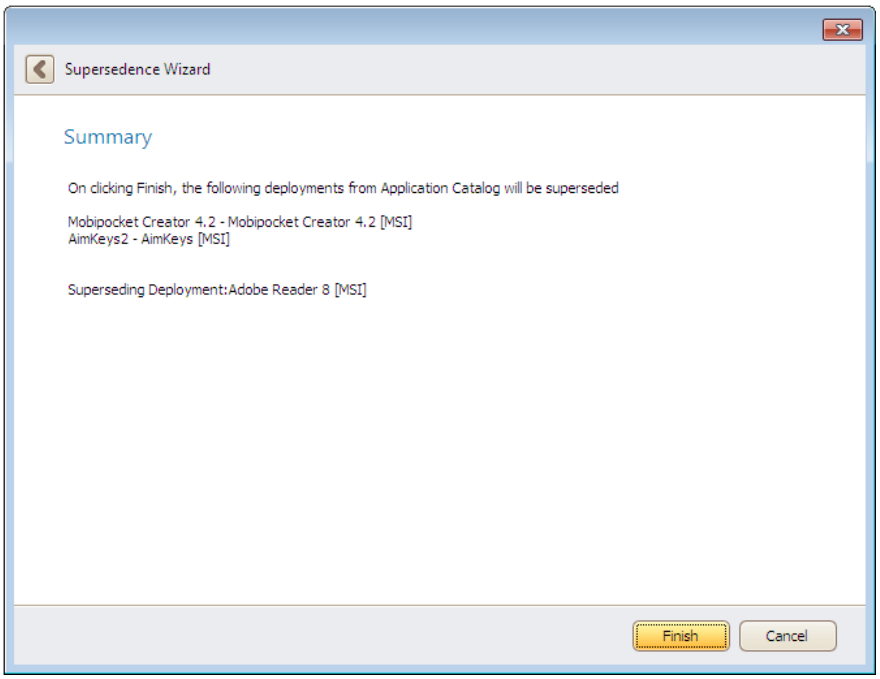


Figure 7-102: Supersedence Wizard / Summary Panel

Test on Virtual Machine Wizard

You can use the **Test on Virtual Machine Wizard** to quickly launch a specified virtual machine and install a selected Windows Installer (.msi), App-V package (.appv), PowerShell-wrapped package (.ps1), or installation executable (.exe) package (both legacy installers and complex installation executables) for testing. This wizard uses the capability of the Automated Application Converter tool to spin up the selected virtual machine and install the selected package.

Note • *Both legacy installers and complex installer executables (which contain bundled Windows Installer packages) can be tested using the Test on Virtual Machine Wizard.*

The Test on Virtual Machine Wizard is launched by right-clicking on the deployment type node of a Windows Installer package (`.msi`), App-V package (`.appv`), PowerShell-wrapped package (`.ps1`) or installer executable (`.exe`) package and selecting **Test on Virtual Machine** from the shortcut menu.

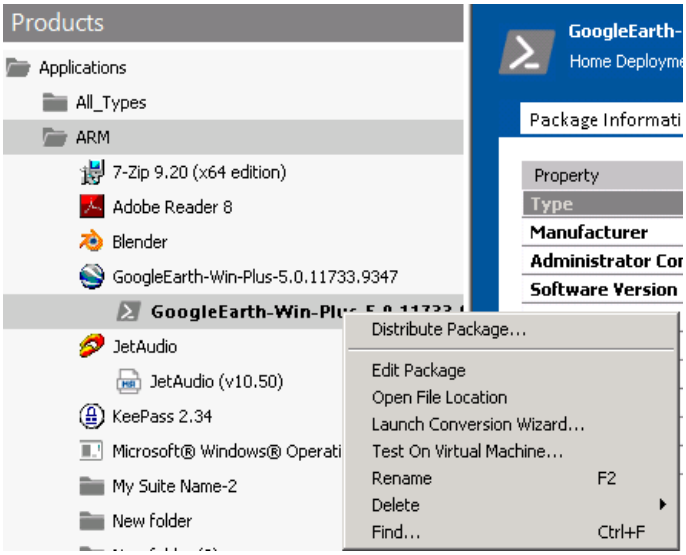


Figure 7-103: Test on Virtual Machine Option

Note • *The **Test on Virtual Machine** selection on the shortcut menu is available on both the **Home** and the **Analyze** tabs of Application Manager.*

The **Test on Virtual Machine Wizard** consists of the following panels:

- Automated Application Converter Test Settings Panel
- Summary Panel
- Performing the Test Process Panel

Automated Application Converter Test Settings Panel

The **Virtual Machine List** field on the **Automated Application Converter Test Settings** panel of the Test on Virtual Machine Wizard lists the virtual machines defined in the Automated Application Converter settings file that is selected on the **Plugin Options > Automated Application Converter Plugin** tab of the **Options** dialog box.

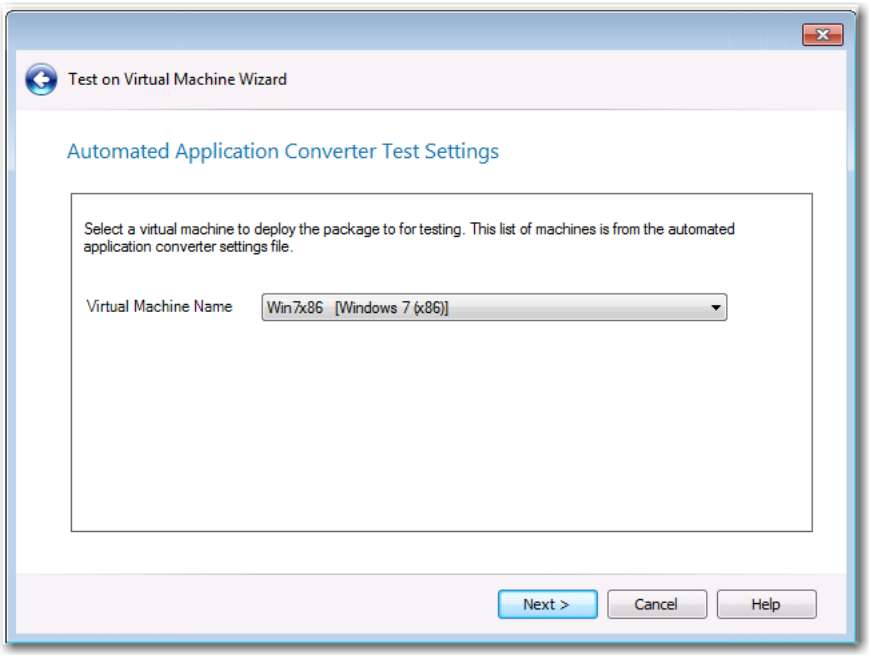


Figure 7-104: Automated Application Converter Test Settings Panel / Test on Virtual Machine Wizard

Select the name of the virtual machine that you want to use for testing and click **Next**.

Summary Panel

The **Summary** panel lists the selections you have made on the previous panels of the Test on Virtual Machine Wizard. Click **Next** to launch the package on the virtual machine for testing.

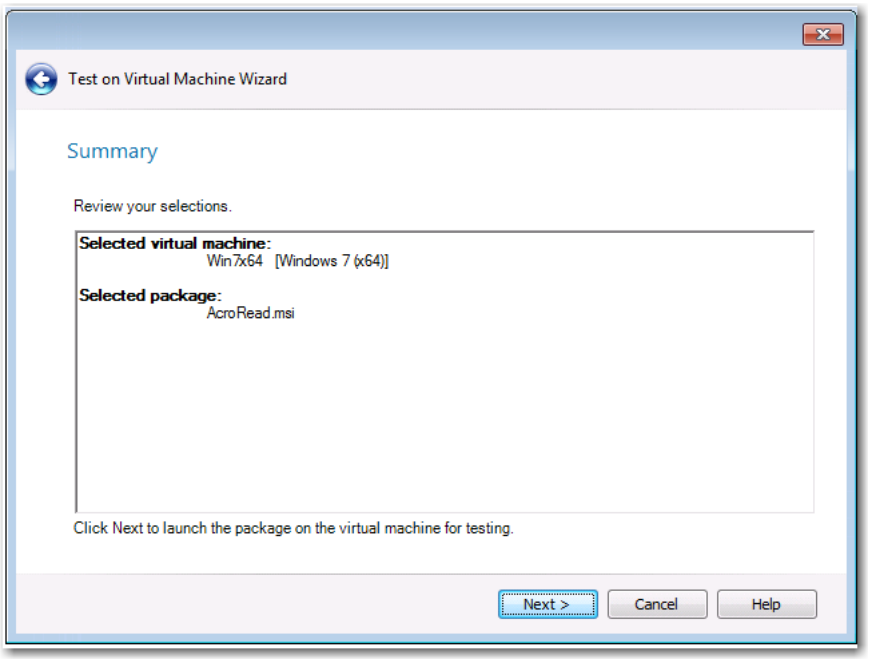


Figure 7-105: Summary Panel / Test on Virtual Machine Wizard

Performing the Test Process Panel

After you click **Next** on the Summary panel of the Test on Virtual Machine Wizard, the **Performing the Test Process** panel opens. The selected package is launched on the specified virtual machine for testing, and progress messages appear on this panel. When the package has been installed and launched on the virtual machine, the **Remote Desktop** button will become enabled.

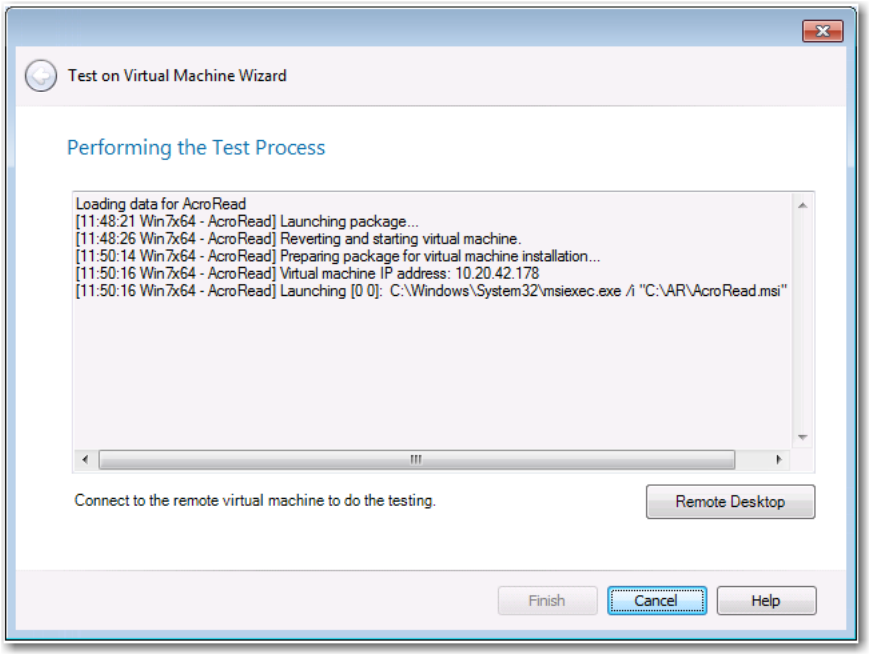


Figure 7-106: Performing the Test Process Panel / Test on Virtual Machine Wizard

Click the **Remote Desktop** button to connect to the specified virtual machine and perform testing. You may be prompted for login credentials to the virtual machine image. A Remote Desktop session opens displaying the virtual image where this package has been installed.

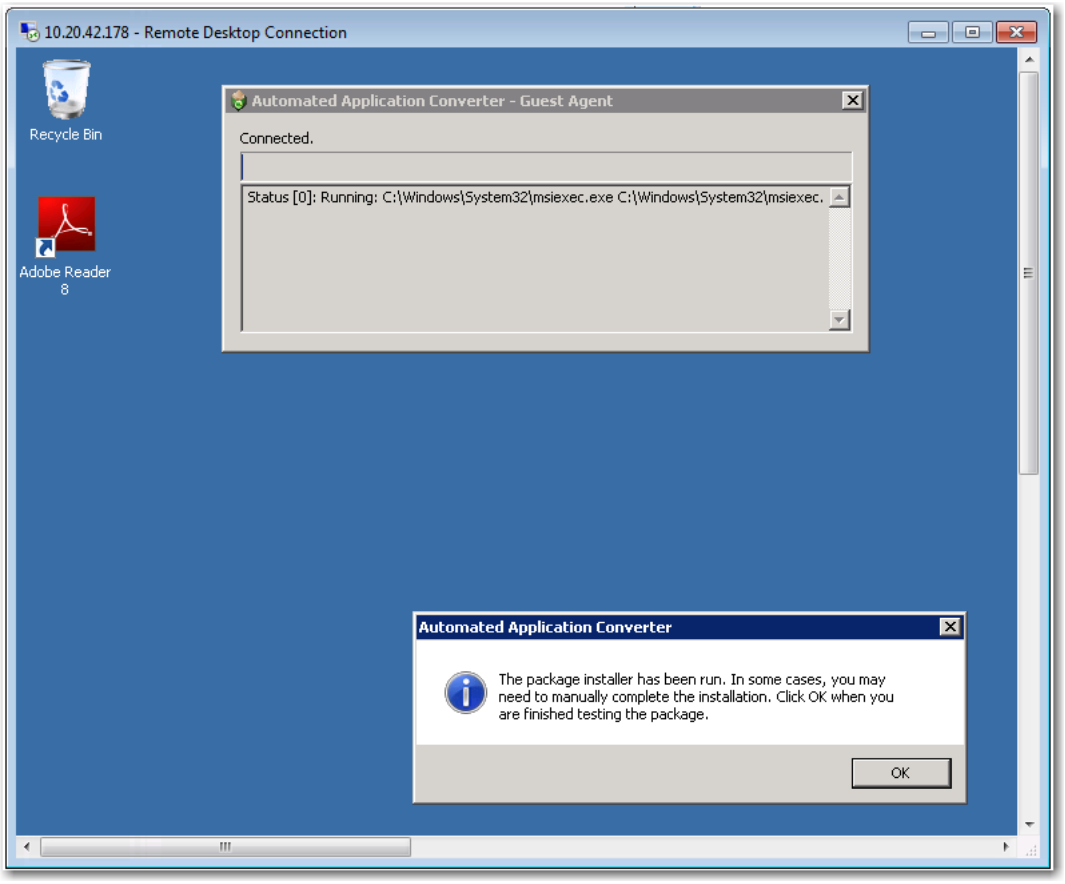


Figure 7-107: Test Session in Remote Desktop Connection Window

Use the installed shortcuts to launch the package and perform the desired testing. When you have finished testing the package, click **OK** to close the Remote Desktop session and shut down the virtual machine.

When you return to the **Test on Virtual Machine Wizard**, click **Finish** to close the wizard.

Upgrade Wizard

When you attempt to open an AdminStudio 5.0, 5.5, 6.0, 7.0, or 7.5 Application Catalog in AdminStudio 2022 R2, you are prompted to upgrade it to use the AdminStudio 2022 R2 schema.

Log files for the upgrade are created in the following directory:

```
AdminStudio Shared Directory\ConflictSolver\Logs
```

Note • *Note the following regarding upgrading an existing Application Catalog:*

- *The upgrade of AdminStudio 3.0, 3.01, and 3.5 databases is not supported by AdminStudio 7.0 or later.*
- *Starting with AdminStudio 8.0, Microsoft Access databases are not supported.*
- *Starting with AdminStudio 9.01, Oracle databases are not supported.*
- *When an SQL Server Application Catalog database is upgraded, the old tables are not dropped from the Application Catalog.*

The Upgrade Wizard consists of the following panels:

- **Welcome Panel**—Initial panel displayed when the Upgrade Wizard is launched. Click **Next** to proceed with the upgrade.
- **Progress Panel**—Displays the results of the upgrade. Click **Finish** to exit the Upgrade Wizard.

User Permissions in Application Manager

Security and permissions can be assigned to Application Manager users to restrict the tasks that they can perform in Application Manager. Detailed information on these permissions and how to assign them can be found in AdminStudio Client Tools Permissions in the Managing Roles and Permissions section of the help library.

Note • *Typically only AdminStudio Administrators and a few select users will have access to create new Application Catalogs, upgrade Application Catalogs, or otherwise affect multiple users.*

Database Server Permissions

In order to operate some AdminStudio tools, AdminStudio users require specific database permissions. Depending upon the type of user, you may wish to be more selective in the permissions you assign to these users.

If you have AdminStudio Enterprise Edition, you can assign permissions to individual users using the Role functionality in AdminStudio Enterprise Server, as described in AdminStudio and Workflow Manager Roles and Permissions. Otherwise, you can provide more selective restrictions at the database server level using the information in the following table, AdminStudio Database Server Permissions.

Every AdminStudio user will need at a minimum read privilege to every table in the Application Catalog. The minimum permissions are described below, based upon the type of operation you want the user to perform.

Table 7-128 • AdminStudio Database Server Permissions

Type	Description
General User Administrative Process	General administrative processes cover a range of activities such as adding groups, moving packages around, adding comments, updating extending attributes, etc. For example, these tables include `cstblPackage`, `cstblGroups`, and `cstblGroupPackages`. Any Application Catalog table which is not referenced explicitly in the discussion for the other AdminStudio processes should be considered a general user administrative table. Most AdminStudio users should have write access to these tables.

Table 7-128 • AdminStudio Database Server Permissions (cont.)

Type	Description
Import Process	The user importing MSI packages, merge modules, or just about anything will require write access to a significant set of Application Catalog tables depending on the type of import. For example: • **MSI package file**—For MSI package file import operations, those Application Catalog tables with a `csmsi` prefix are populated. • **Merge modules**—For merge module import operations, the `csmsm` prefixed tables are used. • **Patches**—For patch import operations, the `cspch` prefix tables are used. • **OS snapshots**—For OS snapshot import operations, the `osc` prefix tables are used.
Validation Process	For this process, the user will need to be able to write entries into the `cstblValidationResults` and `cstblValidationConfiguration` tables.
Dependency Scanning Process	For this process, the user will need to be able to write entries into the `cstblPackageExeDependencies` table.
Conflict Detection and Resolution Process	For this process, the user will need to be able to write entries into the `cstblConflict` prefixed table names.
Patch Impact Analysis Process	For this process, the user will need to be able to write entries into the `cstblPatchConflict` prefixed table names. This process will create and delete some temporary tables and, as such, the user performing this process should have the necessary server privileges to perform these operations.
Package Auto Import Process	The Package Auto Import process will ultimately generate a series of Import operations, and so the user performing these operations should have the Import process rights described above. If the user wants to edit these operations in the Wizard, then they will need write accession to the `cstblSubscribed` prefixed tables.
Workflow Operations Process	For this process, the user will need to be able to write entries into the `wftbl` prefixed table names.
Tools Properties Operations	For this process, the user will need to be able to write entries into the `wftblTools` table name.
Pre-Deployment Testing	For this process, the user will need to be able to write entries into the `pdt` prefixed table names.

Note *• A number of processes within AdminStudio generate data which can subsequently be deleted by the AdminStudio user. Any discussion of the minimum privileges required for a specific AdminStudio process will also imply the privileges to delete this same data.*

8

Repackaging Legacy Installations Using the Repackaging Wizard

Installations created for the Windows Installer service dramatically differ from traditional installations, making reusing legacy installations impossible without using a repackaging tool. You can use Repackager's Repackaging Wizard to capture the data placed on your system during installation and convert it into a Windows Installer (`.msi`) package, which you can then customize and distribute according to your organization's needs.

Documentation regarding using the Repackaging Wizard is presented in the following sections:

Table 8-1 • Using the Repackaging Wizard

Section	Description
About Repackaging	Introduces you to repackaging, explains various repackaging methods, lists Repackaging Best Practices, explains how to include the InstallScript Engine with a Windows installer package, and reviews Repackager options.
Repackaging Methods	Describes the methods of repackaging that the Repackaging Wizard supports.
Configuring Repackager to Ensure Optimal Installation Capture	Describes how to configure Repackager in order to get optimal results when capturing an installation.
Repackaging Legacy Installations Using the Repackaging Wizard	Explains how to use the Repackaging Wizard to convert the following installations: • InstallShield Professional 1.x to 5.1.x • InstallShield Professional 5.5 to 7.x • InstallShield InstallScript MSI • InstallShield DevStudio 9.x InstallScript • InstallShield Editor InstallScript
Documenting Repackaging Steps Using the Microsoft Step Recorder Tool	Explains how to use the Microsoft Step Recorder documentation tool to document the steps taken during repackaging.

Table 8-1 • Using the Repackaging Wizard (cont.)

Section	Description
Repackaging Wizard Reference	Describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackaging Wizard.

Note • *For information on other Repackager features, see Converting Legacy Installations Using the Repackager Interface.*

About Repackaging

This section introduces you to repackaging, lists Repackaging Best Practices, and explains how to set Repackager options.

- Purpose of Repackaging Applications
- Supported Legacy Installation Types
- Repackaging 64-Bit Applications
- Repackaging Options Comparison
- Repackaging Wizard Best Practices
- About Repackaging on Clean Systems
- Including the InstallScript Engine With a Windows Installer Package

Purpose of Repackaging Applications

Installations created for the Windows Installer service dramatically differ from traditional installations, making reusing legacy installations impossible without using a repackaging tool. Repackager assists you by capturing the data placed on your system during installation and converting it into a Windows Installer (`.msi`) package, which you can then customize and distribute according to your organization's needs.

Repackaging an installation into a Windows Installer package provides the following benefits:

- **Can customize it using InstallShield Editor or Tuner**—You can further configure or customize the Windows Installer package to meet your specific needs by editing the `.msi` file in InstallShield Editor or by creating transforms in InstallShield Editor or Tuner.
- **Can perform conflict analysis and resolution**—You can use Application Catalog to check the Windows Installer package for conflicts that may exist between it and other known Windows Installer packages in an Application Catalog database, ensuring the proper installation and functioning of your installations.
- **Can implement application repair and feature advertising**—Finally, once converted to a Windows Installer package, the installation can take advantage of Windows Installer functionality such as application repair and feature advertisement.

Supported Legacy Installation Types

You can use both the Repackaging Wizard and the Repackager interface to create Repackager projects. The tool that you use depends upon the type of installation you are converting:

Table 8-2 • Methods of Creating Repackager Projects

Tool	Installation Source
Repackaging Wizard	You can use the Repackaging Wizard to convert the following installations: • InstallShield Professional 1.x to 5.1.x • InstallShield Professional 5.5 to 7.x • InstallShield InstallScript MSI • InstallShield DevStudio 9.x InstallScript • InstallShield Editor InstallScript See Repackaging Legacy Installations Using the Repackaging Wizard.
Repackager Interface	You can use the Repackager interface to convert the following installations: • Repackager 3.x output (.inc) • Microsoft SMS projects (.ipf) • Microfocus ZENworks project files (.axt/.aot) • WinINSTALL projects (.txt) (6.0, 6.5, 7.x) • Wise installation projects (.wse) • InstallShield Professional log files (.isl) See Converting Legacy Installations Using the Repackager Interface. ***Edition*** • *The Repackager interface is included with AdminStudio Professional and Enterprise Editions.*

Repackaging 64-Bit Applications

Edition • *The Repackager interface is included with AdminStudio Professional and Enterprise Editions.*

Repackager has the capability to repackage both 32-bit and 64-bit applications, as well as hybrid applications (both 32-bit and 64-bit). The Repackaging Wizard remains a 32-bit application, but can be run on both 32-bit (x86) and 64-bit (x64) Windows operating systems. The following table lists the operating systems to use to repackage both 32-bit and 64-bit applications, and the operating systems those repackaged applications will run on.

Table 8-3 • Repackaging 32-Bit and 64-Bit Applications

Application Type	Repackage on ...	Will run on ...
64-bit application	Windows 64-bit OS ***Note*** • *You can use either the* ***Installation Monitoring*** *or* ***Snapshot*** *method to repackage a 64-bit application on a 64-bit operating system.*	Windows 64-bit OS
32-bit application	Windows 32-bit OS ***Important*** • *While it is possible to repackage a 32-bit application on a 64-bit OS, it is recommended that you use a 32-bit OS, to avoid inadvertently capturing any 64-bit data. If Repackager captures any 64-bit data, it will flag the package as a 64-bit application, meaning that it will only run on a 64-bit OS. See Excluding 64-Bit Data.*	Windows 64-bit OS or Windows 32-bit OS

Excluding 64-Bit Data

It is strongly recommended that you repackage 32-bit applications on a 32-bit OS. However, if you choose to repackage a 32-bit application on a 64-bit OS, you need to make sure that you exclude any unnecessary 64-bit data, such as data from a 64-bit Windows Service that could be running or 64-bit files (stored in the `System64Folder`, `ProgramFiles64Folder`, or `CommonFiles64Folder` directories) or 64-bit registry entries (any entries stored in a node other than `WOW6432Node`).

Repackaging Options Comparison

Edition • *The Repackager interface is included with AdminStudio Professional and Enterprise Editions.*

The following table details the different options available to you when using Repackager, based upon source type, product and version:

Table 8-4 • Repackaging Options Comparison Chart

Source	Product / Version	Repackaging Method	Result
Media	IS Professional 1.x to 5.1.x	**Repackaging Wizard** Installation Monitoring **or** Snapshot	Repackager project with no feature delineation
	IS Professional 5.5 to 7.x	**Repackaging Wizard** Installation Monitoring **or** Snapshot	Repackager project with feature delineation, including registry entries and shortcuts
	IS InstallScript MSI IS Editor InstallScript IS DevStudio 9.*x* InstallScript	**Repackaging Wizard** Installation Monitoring **or** Single Step Snapshot	Repackager project with feature delineation, including registry entries and shortcuts
Project	Repackager 3.x output (`.inc`) Microsoft SMS projects (`.ipf`) Microfocus ZENworks projects (`.axt/.aot`) WinINSTALL projects (`.txt`) (6.0, 6.5, 7.x) Wise installation projects (`.wse`)	**Repackager Interface** Select **Open** on the **File** menu to have Repackager automatically convert file to a Repackager project	Repackager project with no feature delineation
	InstallShield Editor Pro log files (`.isl`)	**Repackager Interface** Select **Open** on the **File** menu to have Repackager automatically convert file to a Repackager project	Repackager project with feature delineation

Once you have created a Repackager project, you can visually examine the files, `.ini` files, shortcuts, and registry data from the installation, and exclude any non-essential items. Then, you can build the Repackager project into an InstallShield Editor project (`.ism`) for further editing, or create a Windows Installer package (`.msi`).

Repackaging Wizard Best Practices

***Edition** • The Repackager interface is included with AdminStudio Professional and Enterprise Editions.*

To ensure optimal performance of the Repackaging Wizard during repackaging and when working with Repackager projects, the following best practices are recommended:

- Repackage on a Clean System
- Launch Repackager Remotely or Install Repackager on the Clean Machine
- Use the Repackager Interface to Exclude Unwanted Items

- Exit All Other Applications
- Only Repackage Non-Windows Installer Setups

Repackage on a Clean System

It is essential that you repackage applications on a "clean" system to ensure you capture all changes made by the installation. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. Repackaging on a clean system provides the following benefits:

- **Prevents you from capturing Repackager files**—By repackaging on a clean system, you are ensuring that you do not inadvertently capture Repackager files during repackaging.
- **Ensures that you capture all of the necessary setup files**—If you do not repackage on a clean system, you may not capture all of the necessary files for the setup because the files may already be installed on the system.

Note • *For more information, see About Repackaging on Clean Systems.*

Launch Repackager Remotely or Install Repackager on the Clean Machine

Because it is best to keep the number of packages installed on the clean machine to a minimum, you should launch Repackager remotely from the clean machine or install Repackager on the clean machine:

- **Launch Repackager Remotely**—You could install Repackager on a shared network drive and then launch Repackager remotely from the clean machine. See Launching Repackager Remotely.
- **Install Repackager on clean machine**—You could install a copy of Repackager onto the clean machine. While it is preferable to launch Repackager remotely from the clean machine, if you do not have network access to an installation of the AdminStudio client tools, this is your best option. See Installing Repackager on a Clean Machine.

Both of these options are explained in Configuring Repackager to Ensure Optimal Installation Capture.

Use the Repackager Interface to Exclude Unwanted Items

You should repackage using the provided exclusions and then use the Repackager interface to visually remove unwanted items from the capture.

Because this occurs post-capture, you do not need to recapture the legacy setup if you inadvertently exclude items from the Windows Installer package you are building.

Note • *Since Windows Installer does not support packaging device drivers, you would need to create Custom Actions to install device drivers. See Using Custom Actions in the Windows Installer help section for more information.*

Exit All Other Applications

Other applications may lock files or directories, and may hinder the performance of the setup and repackaging. Therefore, exit all applications prior to repackaging.

Only Repackage Non-Windows Installer Setups

Windows Installer setups should not be repackaged. They should either be edited in InstallShield Editor, or, as Microsoft recommends, by creating a transform. This can be done using InstallShield Editor or Tuner.

You should not repackage Windows Installer (`.msi`) packages for the following reasons:

- Repackaging a Windows Installer package is against Microsoft Best Practices.
- If you make changes to a Windows Installer package, vendors will no longer provide support for that product.
- If you repackage a Windows Installer package, the component codes within the package are not retained and hence future patching or upgrades will not work.
- Traditionally, repackaging tools will ignore the Windows Installer-specific data in the Registry. This will result in an incomplete package.

Also, Repackager is not intended for repackaging operating system installations or service packs, or deeply integrated operating system components such as Internet Explorer. Moreover, components such as MDAC or DCOM should be included in the clean image, or installed by a setup using the vendor's redistributable.

Exception to This Rule

In general, due to the reasons listed above, it is not recommended to repackage a vendor-created Windows Installer package to create a new Windows Installer package. However, some IT organizations may elect to repackage Windows Installer packages in order to simplify them, which should make them more reliable and less likely to violate the organization's and Microsoft's recommended best practices.

If you choose to repackage a Windows Installer package, you need to keep in mind that you may no longer be able to:

- Directly deploy vendor-provided patches for this package, OR
- Use any vendor-provided automatic updating service for this package.

Therefore, you should only consider repackaging a Windows Installer package if your IT staff is also willing to invest resources into periodically repackaging that application's vendor patches into an updated Windows Installer package.

Note • *Tightly-controlled organizations probably would not want to have automatically-updating software, so the inability to use an automatic updating service may not be of concern to them.*

About Repackaging on Clean Systems

For optimal results when using Repackager or OS Snapshot, you should perform these processes on a clean system. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. It is the baseline system that the computer requires to run.

Although it may be tempting to consider basic software, such as Microsoft Office, as part of the clean system, this can result in poor snapshots and repackaged setups. Each application you install on the baseline system adds to the DLLs, changes versions of files, makes new registry entries, etc. This may cause Repackager or the OS Snapshot Wizard to miss these during capture, which ultimately may lead to missing files or registry entries in repackaged setups, or unexpected conflicts between the operating system and Windows Installer packages.

Note • *For more information on setting up a clean system to repackage on, see Configuring Repackager to Ensure Optimal Installation Capture.*

Alternate-Language Repackaging on Clean Machines

The standalone Repackager setup for clean machines does not install any language resources other than US English. Therefore, if you are Repackaging a setup on a clean system in a language other than in US English, you need to ensure you point to the correct template in the Repackaged Output View.

This can be on a mapped network drive, or you can copy the language-populated template (for example, `ISProjBlankTpl.ism`) to your clean system from the `[AdminStudioInstallDirectory]\Editor\Support` directory.

Language-specific templates are available when you purchase InstallShield Editor Language Packs.

Including the InstallScript Engine With a Windows Installer Package

Should you need to include the InstallScript engine with your setup, all the major releases of the InstallScript engine are available in the `InstallScript_Engines` folder on the AdminStudio installation CD. For more information, see the *Update to the Latest InstallShield Installation Engines* Knowledge Base article at:

https://community.flexera.com/t5/InstallShield-Knowledge-Base/Update-to-the-Latest-InstallShield-Installation-Engines/ta-p/4591

Repackaging Methods

Repackager supports three methods of repackaging:

Table 8-5 • Repackaging Methods

Repackaging Method	Description
Installation Monitoring Method	Repackager monitors system changes as an application is installed, and that data can be converted into a Windows Installer package. Installation Monitoring is the default method. ***Edition*** • *The Repackaging Wizard Installation Monitoring method is included with AdminStudio Professional and Enterprise Editions.*
Snapshot Method	Repackager compares a system snapshot before and after an installation, determines the changes that were made, and that data can be converted to a Windows Installer package.

Installation Monitoring Method

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

When using the **Installation Monitoring** method, the Repackaging Wizard monitors a system for any processes that are created during an installation. The Installation Monitoring method determines the dynamic interdependencies between files.

By monitoring these processes in the background, the Repackaging Wizard can identify files, shell links, and registry entries that are added, modified, or removed by the installation. The resulting files and Repackager output file can be converted into a Windows Installer package.

Installation Monitoring Method Considerations

Consider the following about Installation Monitoring when selecting a repackaging method:

- **Faster than Snapshot**—Installation Monitoring is significantly faster than the Snapshot repackaging method.
- **Clean system not required**—Although it is still a good practice to repackage on a clean system, it is not as important when using Installation Monitoring technology as it is when you use the Snapshot method.
- **Can exclude processes from the project**—When using the Installation Monitoring method, you can specify the processes that you want to exclude from the Installation Monitoring.
- **Enhanced system reboot handling**—On Windows Vista and newer, system reboots are almost instantaneous and do not allow running applications to properly shut down, which may result in a loss of data. When using the Installation Monitoring method, Repackager successfully handles a system reboot and delays it until you click the Reboot button on the Repackaging Wizard.
- **Windows side-by-side support**—The Repackager Installation Monitoring method scans and detects changes made to the Windows SxS (Side-by-Side) store and automatically includes the proper merge modules.

Snapshot Method

When using the **Snapshot** method, the Repackaging Wizard takes a reference snapshot of a system as a baseline configuration, performs the installation, and then takes a second snapshot.

The difference between the two snapshots is stored in a directory you specify, along with the Repackager output file (`.inc`). This file can then be converted into a Windows Installer package (`.msi`) using Repackager.

Snapshot Method Considerations

Consider the following about Snapshot technology when selecting a repackaging method:

- **Slower than Installation Monitoring**—The Snapshot method is significantly slower than the Installation Monitoring repackaging method.
- **Clean system is required**—When repackaging using Snapshot technology, you should use a clean system, with a baseline configuration for your target environment. If you do not repackage on a clean system, you may not capture all of the necessary files for the setup because the files may already be installed on the system.
- **Exclude anti-virus software directories**—Any machine that you use to repackage most likely has anti-virus software installed on it, even a “clean” machine. While you are repackaging an application, the real-time virus detection feature of anti-virus software could automatically update various cached files in its directories. In order to avoid repackaging errors when using the **Snapshot** method, you should exclude these directories. See Excluding Directories and Subdirectories for more information.

Note • *Anti-virus software does not affect repackaging using the **Installation Monitoring** method.*

Configuring Repackager to Ensure Optimal Installation Capture

Both repackaging methods, **Installation Monitoring** and **Snapshot**, involve installing an application and recording the system changes made by that installation. To ensure that you capture *all* changes made by the installation, you should, ideally, install the application onto a "clean machine" (a computer with only the operating system installed), as described in About Repackaging on Clean Systems.

Depending upon your network connectivity, you should configure Repackager on a clean machine in one of the following ways:

Table 8-6 • Methods to Configure Repackager

Repackager Configuration	Description
Launching Repackager Remotely	If you have connectivity from a clean machine to a computer or network location that contains an installation of Repackager, you should launch Repackager remotely.
Installing Repackager on a Clean Machine	If you do not have any network connectivity on the clean machine, you should install Repackager on the clean machine.

Launching Repackager Remotely

Because you want to avoid installing applications on the clean machine, you should launch Repackager remotely from the clean machine.

To launch Repackager remotely, perform the following tasks:

- Prerequisites to Launch Repackager Remotely
- Sharing Directories on a Machine with an Installation of AdminStudio
- Creating a Shortcut to the Repackaging Wizard on the Clean Machine
- Remotely Launching Repackaging Wizard on the Clean Machine

Prerequisites to Launch Repackager Remotely

To launch the Repackager remotely in a clean machine, the following redistributables must be installed:

Table 8-7 • Redistributables & version

Redistributables	Versions
Microsoft Visual C++ 2008 (x86)	9.0.30729
Microsoft Visual C++ 2010 (x86)	10.0.40219
Microsoft Visual C++ 2012 (x86)	11.0.50727.1

Table 8-7 • Redistributables & version

Redistributables	Versions
Microsoft Visual C++ 2012 (x86)	11.0.61030.0
Microsoft Visual C++ 2015 (x86)	14.0.24212.0

Sharing Directories on a Machine with an Installation of AdminStudio

To share directories on a machine where AdminStudio is installed, perform the following steps:

Task ***To share the Repackager and AdminStudio Shared folders:***

1. Locate a production machine with network access that has AdminStudio installed on it.

   Tip • *Check to make sure that this installation of AdminStudio has already been activated before proceeding.*

2. Open Windows Explorer and locate the following directory:

   `[AdminStudioInstallDirectory]\Repackager`

3. Right-click the Repackager directory and then click **Properties**. The **Repackager Properties** dialog box opens.
4. Open the **Sharing** tab of the **Repackager Properties** dialog box.
5. Click the **Advanced Sharing** button to open the **Advanced Sharing** dialog box.
6. Select **Share this folder** and configure sharing rights as necessary.
7. Click **OK** to close the **Advanced Sharing** dialog box and click **Close** to close the **Repackager Properties** dialog box.
8. Repeat the steps above to also share the `AdminStudio Shared` directory used by that installation of AdminStudio.

Creating a Shortcut to the Repackaging Wizard on the Clean Machine

To create a shortcut to Repackaging Wizard on the clean machine, perform the following steps:

Task ***To create a shortcut to the Repackaging Wizard on the clean machine:***

1. On this clean machine with network access, right-click on the **Computer** icon in Windows Explorer and select **Map network drive...** from the shortcut menu. The **Map Network Drive** dialog box opens.
2. Specify the drive letter you want use to represent the shared location.
3. Click **Browse**. The **Browse for Folder** dialog box opens.
4. Select the shared Repackager directory on the production machine (that you configured in Sharing Directories on a Machine with an Installation of AdminStudio) and click OK.
5. Click **Finish** to exit the **Map Network Drive** dialog box.

6. From Windows Explorer, navigate to the drive mapped to the shared Repackager directory on the production machine.
7. Right-click on the Repack.exe file (the Repackaging Wizard executable file), point to **Send To**, and click **Desktop (create shortcut)**. A shortcut to Repackager in the shared directory is now on the Desktop.

Remotely Launching Repackaging Wizard on the Clean Machine

To remotely launch the Repackaging Wizard on the clean machine, perform the following steps:

Task ***To remotely launch the Repackaging Wizard on the clean machine:***

1. On the clean machine, double-click the **Repackaging Wizard** shortcut on the desktop (that you created in Creating a Shortcut to the Repackaging Wizard on the Clean Machine). The **Welcome** panel of the **Repackaging Wizard** opens.

   Important • *Because you are running the Repackaging Wizard remotely, the online help topics cannot be viewed. However, you can view a version of AdminStudio Help Library online at:*

   https://docs.flexera.com

2. Continue using the Repackaging Wizard to capture a legacy setup, following the instructions in Repackaging Legacy Installations Using the Repackaging Wizard.

   Caution • *On the **Set Target Project Information and Capture Settings** panel of the Repackaging Wizard, do not set the **Project path to store files** field to a location on the clean machine; instead choose a network location.*

Installing Repackager on a Clean Machine

It is essential that you repackage applications on a “clean" system to ensure you capture all changes made by the installation. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. It is the baseline system that the computer requires to run.

While you want to avoid installing applications on the clean machine, if the clean machine does not have network connectivity to an installation of Repackager (which is required in order to run Repackager remotely), you have to install Repackager locally on a clean machine by running the Repackager installation.

To install a standalone version of Repackager on a clean machine, perform the following steps.

Note • *You cannot install Repackager on a machine that already has a copy of Repackager installed.*

Task ***To install Repackager on a clean machine:***

1. Build a “clean machine”—a computer with only the operating system and necessary service packs installed on it.
2. Download StandaloneRepackager.exe from the Flexera Product and License Center using the same credentials you used when you downloaded the full installer.

3. Copy `StandaloneRepackager.exe` to the clean machine.
4. Launch the setup. The **Welcome Panel** opens.
5. Click **Next**. The **License Agreement** panel opens.
6. Select the **I accept the terms of the license agreement** option and click **Next**. The **Customer Information** panel opens.
7. Enter a **User Name** and **Organization** name to identify this installation of Repackager.
8. Enter the **Activation Code** you received for the edition of AdminStudio that you purchased.
9. Click **Next**. The **Destination Folder** panel opens.
10. If you want to install Repackager in the specified directory, click **Next**. If you want to select a different directory, click **Change**, select a new directory, and then click **Next**. The **AdminStudio Shared Location** panel opens.

    The `AdminStudio Shared` directory contains shared information for repackaging and conflict identification, and other AdminStudio functions. With regard to Repackager, the `AdminStudio Shared` directory contains the following:

    - Repackager `isrepackager.ini` exclusion list
11. Specify the location of your organization's `AdminStudio Shared` directory, and click **Next**. The **Ready to Install** panel opens.
12. Click **Install** to begin the installation process. The **Installing Repackager** panel opens. When installation is complete, the **InstallShield Wizard Completed** panel opens.
13. Click **Finish** to exit the Wizard. A Repackager shortcut will be added to the Windows **Start** menu under **AdminStudio**, **AdminStudio Tools**.

Repackaging Legacy Installations Using the Repackaging Wizard

One frequently used method of creating a Repackager project is to repackage a legacy setup. Fundamentally, this involves monitoring the execution of a non-Windows Installer setup and converting changes made by the setup into a Windows Installer file.

Repackager provides the Repackaging Wizard for accomplishing this task. Using this Wizard, you can select the repackaging method (either Snapshot or Installation Monitoring), specify the setup(s) you want to repackage, and run the setup(s). When the Repackaging Wizard has finished its analysis, Repackager automatically creates a Repackager project (`.irp`) file, which can be modified in Repackager. You can then convert this file to an InstallShield Editor project (`.ism`) for further editing, or convert it directly to a Windows Installer package (`.msi`).

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

When using the Repackaging Wizard to repackage a legacy setup, you can use any of the following methods:

Table 8-8 • Repackaging Methods

Repackaging Method	Description
Installation Monitoring Method	Repackager monitors system changes as an application is installed, and that data can be converted into a Windows Installer package. Installation Monitoring is the default method. See Repackaging Using the Installation Monitoring Method. ***Edition*** • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*
Snapshot Method	Repackager compares a system snapshot before and after an installation, determines the changes that were made, and that data can be converted to a Windows Installer package. This is the default method. See Repackaging Using the Snapshot Method.
Using InstallScript Scan	You can use the Repackaging Wizard and InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package. See Repackaging an InstallScript MSI Setup to a Basic MSI Setup

Repackaging Using the Installation Monitoring Method

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

When you choose the **Installation Monitoring** method of repackaging, Repackager monitors system changes as an application is installed, and then you can convert that data into a Windows Installer package. Installation Monitoring is the default method.

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

To repackage an installation using the Installation Monitoring method, perform the following steps:

- Step 1: Selecting the Repackaging Method.
- Step 2: Excluding Processes (Optional)
- Step 3: Collecting Product Information
- Step 4: Adding Additional Setup Programs (Optional)

- Step 5: Set Target Project Information
- Step 6: Set Capture Settings (Optional)
- Step 7: Beginning the Repackaging Process

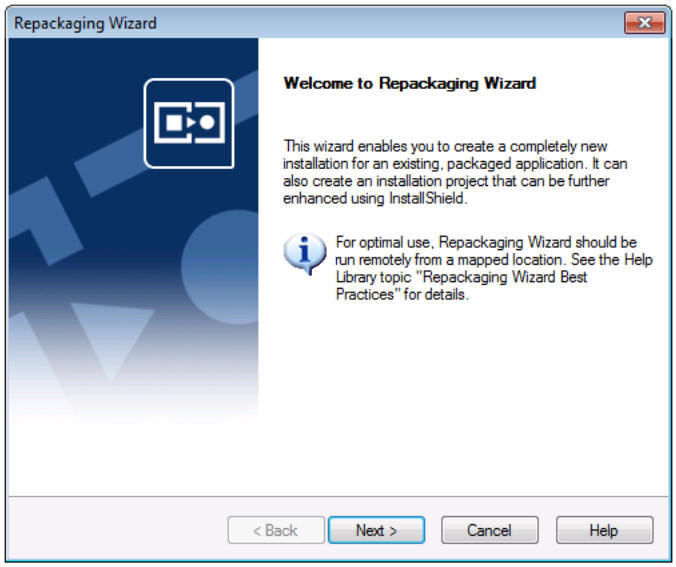
Step 1: Selecting the Repackaging Method

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

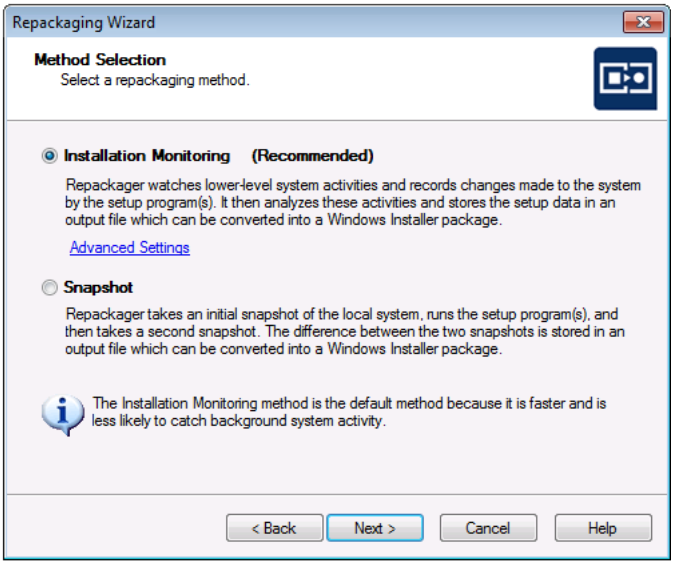
In this step, you launch the Repackaging Wizard and select the Installation Monitoring repackaging method.

Task ***To select a repackaging method:***

1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.



2. Click **Next**. The Method Selection Panel opens.



3. Select **Installation Monitoring**.
4. Continue with Step 2: Excluding Processes (Optional).

Step 2: Excluding Processes (Optional)

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

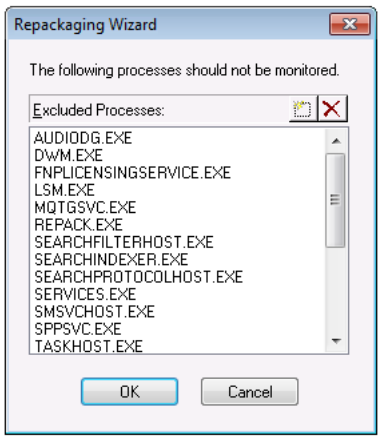
During Installation Monitoring, Repackager captures all of the activity of each service or process running on the machine, and then processes this collected data. However, many services running on a machine may have nothing to do with the installation being repackaged.

- **If you want to modify the default excluded processes list**, perform the following steps.
- **If you do not want to modify the default excluded processes list**, continue with Step 3: Collecting Product Information.

Tip • *If you know that the installation that you are capturing is from a self-extracting* **.exe** *file and if you want to use the* ***Installation Monitoring*** *method, you should add the name of that* **.exe** *file to the excluded processes list.*

Task ***To exclude processes from Installation Monitoring:***

1. On the **Method Selection** panel, click the **Advanced Settings** link. The **Excluded Processes** dialog box opens, listing a default set of processes.



2. To add a process to this list, click the New ( ) button to add a new blank line to this list, and enter the name of the process that you want to exclude.

3. To delete a process from this list, select the process and click the Delete ( ) button.

Note • *The changes you make to the excluded processes list are persisted for future Repackaging sessions. Therefore, once you have entered an appropriate set of processes to exclude for your machine, you can skip this optional step.*

4. Click **OK** to return to the Method Selection Panel.

5. Continue with Step 3: Collecting Product Information.

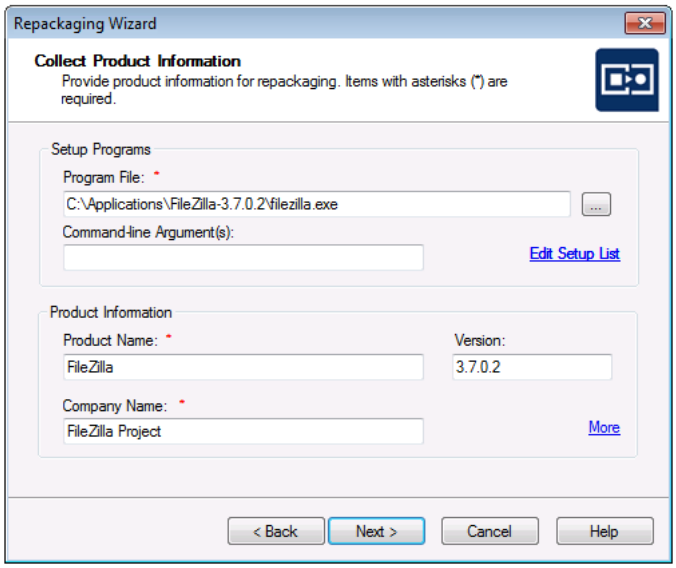
Step 3: Collecting Product Information

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

In this step, you will specify the installation you want to repackage and enter any command-line arguments to be used when the installation is run.

Task ***To enter product information:***

1. On the Method Selection Panel, click **Next**. The **Collect Product Information** panel opens.



2. Click the Browse (...) button next to the **Program File** field and select the installation program that you are repackaging.
3. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when the installation is run.
4. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
5. If you want to associate websites with this installation, click the **More** link to open the Additional Product Information dialog box, enter the **Product URL** and **Support URL** for the application you are repackaging, and click OK.
6. Continue with Step 4: Adding Additional Setup Programs (Optional).

Step 4: Adding Additional Setup Programs (Optional)

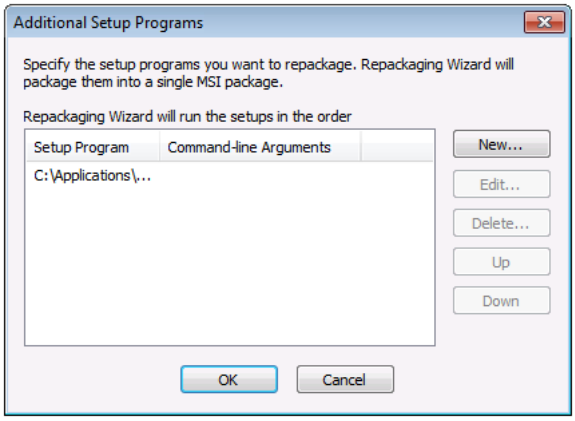
Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

You can specify additional setup programs to repackage together with this installation. Additional setup programs share the same product name, version number, and company name in the repackaged installation. However, as you locate each additional setup program to repackage, you can specify command-line parameters pertaining only to that setup. You can also specify the order in which the setups are run, should it be necessary.

- **If you want to add additional setup programs**, perform the following steps.
- **If you do not want to add additional setup programs**, continue with Step 5: Set Target Project Information.

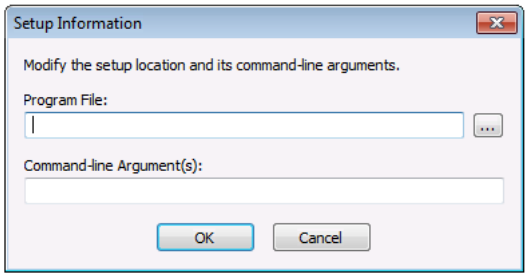
Task ***To add additional setup programs, perform the following steps:***

1. On the Collect Product Information Panel, click the **Edit Setup List link**. The **Additional Setup Programs** dialog box opens.



2. If you want to **add** a setup program, perform the following steps:

   a. Click **New**. The **Setup Information** dialog box opens.



   b. Click the Browse (...) button next to the **Program File** field and select the setup program that you want to add.

   c. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when this setup is run.

   d. Click **OK** to return to the **Additional Setup Programs** dialog box.

   e. If necessary, click the **Up** and **Down** buttons to change the order in which the setups are run.

3. If you want to **edit** an existing setup program, perform the following steps:

   a. On the Additional Setup Programs dialog box, select the program that you want to edit and click **Edit**. The Setup Information dialog box opens.

   b. Modify the **Program File** and **Command-line Argument(s)** fields.

   c. Click **OK** to return to the Additional Setup Programs dialog box.

4. If you want to **delete** a listed setup program, perform the following steps:

   a. Select the program that you want to delete and click **Delete**. A dialog box opens prompting you to confirm the deletion.

b. Click **OK** to confirm the deletion and return to the Additional Setup Programs dialog box, where the deleted program is no longer listed.

5. Click **OK** to return to the **Collect Product Information** panel.
6. Continue with Step 5: Set Target Project Information.

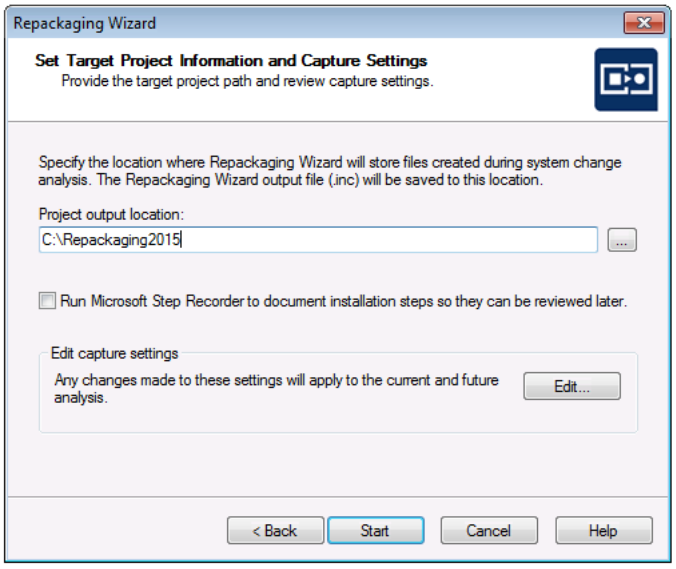
Step 5: Set Target Project Information

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

In this step, you identify the location where you want files created by Repackager to be stored. For the Installation Monitoring repackaging method, it is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

Task ***To set target project information and capture settings:***

1. On the Collect Product Information Panel, click **Next**. The Set Target Project Information and Capture Settings Panel opens.



2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 6: Set Capture Settings (Optional).

Step 6: Set Capture Settings (Optional)

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

From the **Set Target Project Information** and **Capture Settings Panel**, you can specify the following capture types for the repackaging session:

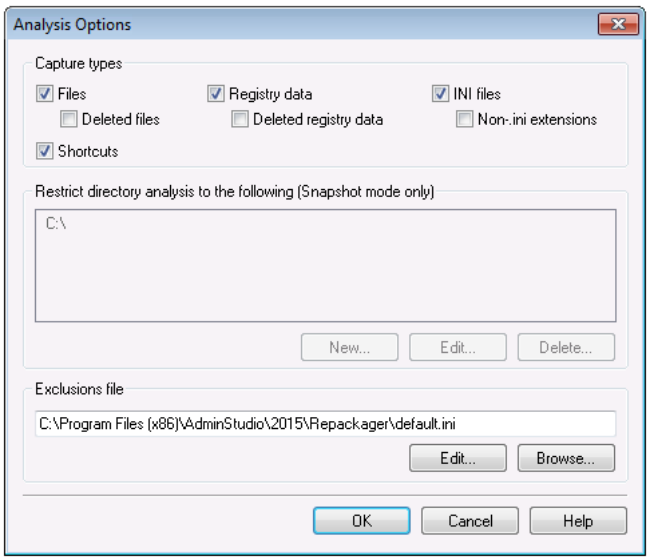
- Files and deleted files
- .ini files and .ini files with non-.ini extensions
- Shortcuts
- Registry data and deleted registry data

Options set in this dialog box apply to the current and subsequent repackaging sessions.

- **If you want to set capture settings**, perform the following steps.
- **If you do not want to set capture settings**, continue with Step 7: Beginning the Repackaging Process.

Task ***To set capture settings:***

1. On the **Set Target Project Information and Capture Settings** panel, click **Edit**. The **Analysis Options** dialog box opens.



Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

2. Select the capture types that you want to use for this repackaging session:

Type	Description
Files	Capture file names during repackaging.

Type	Description
Deleted files	Capture deleted file names during repackaging. ***Note*** • *If you select this option, deleted files will be displayed on the Deleted Files View of the Repackager interface.*
Registry data	Capture registry data during repackaging.
Deleted registry data	Capture deleted registry data during repackaging. ***Note*** • *If you select this option, deleted registry entries will be displayed on the Deleted Registry Entries View of the Repackager interface.*
INI files	Capture .ini files during repackaging.
Non-ini extensions	Capture .ini files with non-.ini extensions during repackaging.
Shortcuts	Capture shortcuts during repackaging.

3. Click **OK** to return to the **Set Target Project Information and Capture Settings** panel.
4. Continue with Step 7: Beginning the Repackaging Process.

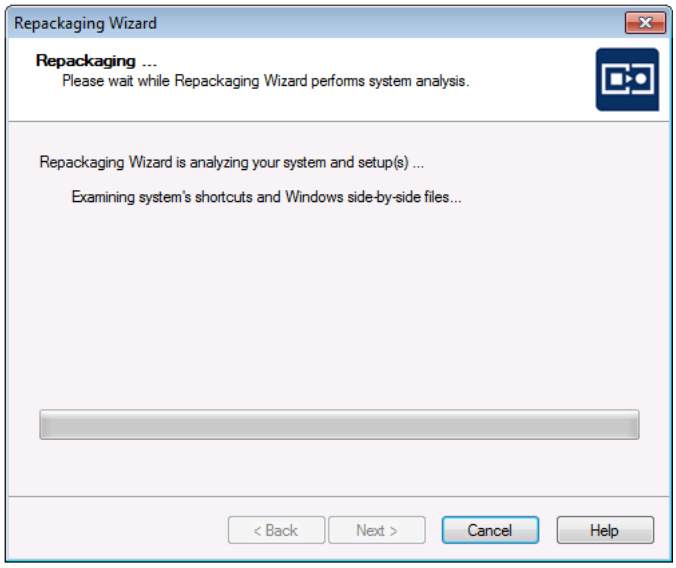
Step 7: Beginning the Repackaging Process

Edition • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

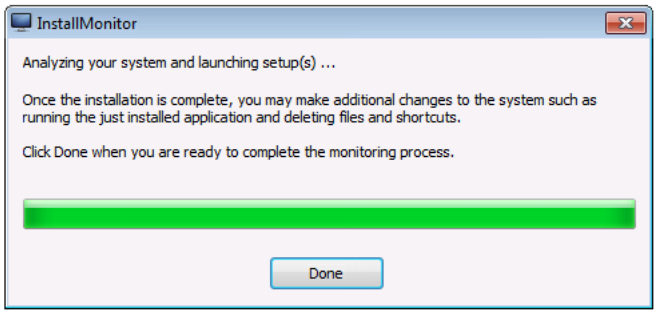
In this step you will begin the repackaging process.

Task ***To begin the repackaging process:***

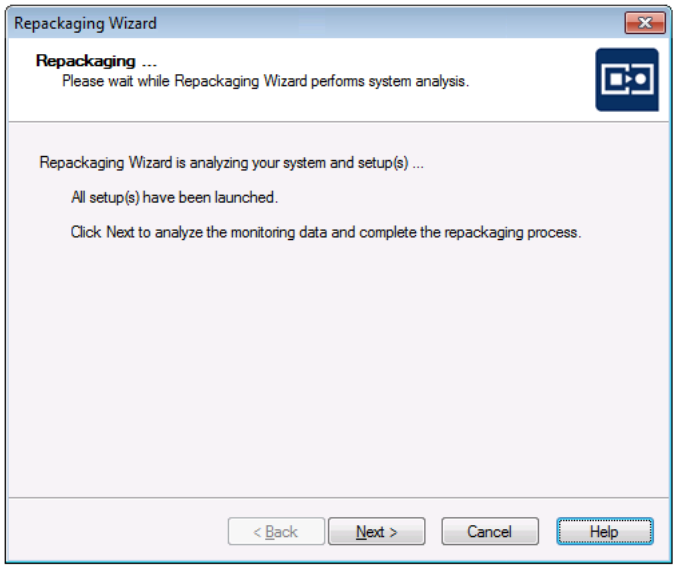
1. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status. Then, the selected setup program will be launched.



2. Follow the prompts until the installation has completed. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.

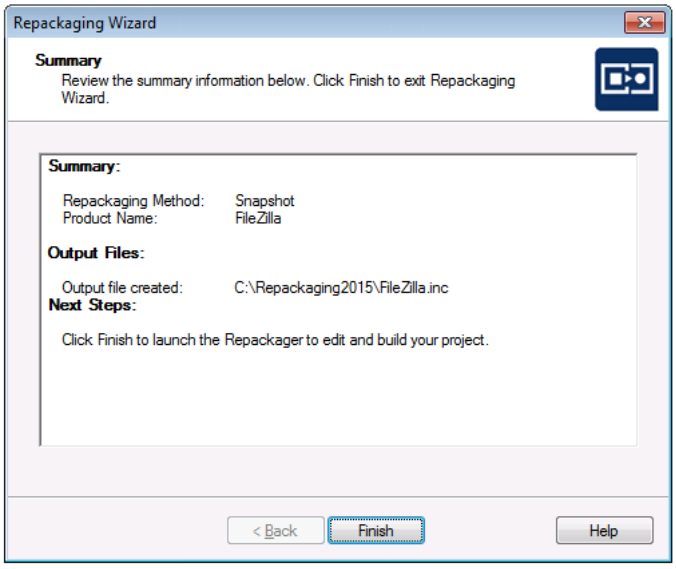


3. When you are ready to complete the monitoring process, click **Done**. You are then prompted to click **Next** to analyze the monitoring data and complete the repackaging process.



4. When you are ready to complete the repackaging process, click **Next**. The Repackaging Wizard then analyzes the system and setup data that it collected.

   Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.



5. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.
6. Continue with the instructions in Working With Repackager Projects.

Repackaging Using the Snapshot Method

When using the **Snapshot** method of repackaging, the Repackaging Wizard takes a reference snapshot of a system as a baseline configuration, performs the installation, and then takes a second snapshot.

The difference between the two snapshots is stored in a directory you specify, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and the source files. The Repackager project file can then be converted into a Windows Installer package (`.msi`).

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

Types of Snapshot Repackaging

There are two types of Snapshot repackaging:

Single Step

When Repackaging in a single step:

- You specify at least one setup program to repackage.
- Repackager first takes an initial system snapshot.
- Repackager then runs the setup program(s) you selected.
- Then Repackager takes a second snapshot to create the script file that can be converted into a Windows Installer package.

You also have the option of requiring the Repackager to prompt you before running the setup program(s), allowing you the opportunity to make changes to your system that you want included in the final package.

See Performing Single Step Snapshot Repackaging.

Multiple Step

When repackaging in multiple steps:

- You run the Repackager to obtain an initial system snapshot, after which the Repackager exits.
- You can then perform any modifications to the system, such as changing configurations, running installations, and so forth.
- After making the necessary modifications, you would then run the Repackager again to analyze system status changes.
- Repackager compares the final snapshot to the initial snapshot to determine the system changes that were made, and then records that information in a script file.

See Performing Multiple Step Snapshot Repackaging.

Performing Multiple Step Snapshot Repackaging

To repackage an installation using the **Multiple Step Snapshot** method, perform the following steps:

- Step 1: Selecting the Repackaging Method.
- Step 2: Initial Analysis
- Step 3: Install Setup and Make Manual System Changes

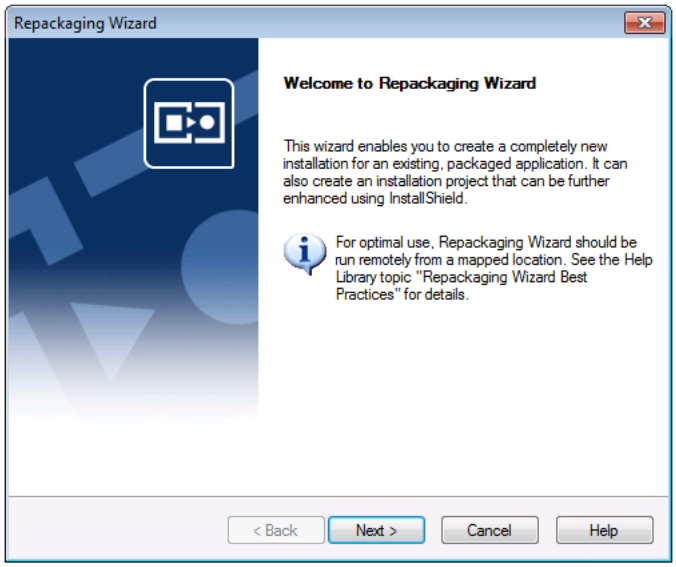
- Step 4: Entering Product Information
- Step 5: Set Target Project Information
- Step 6: Set Capture Settings (Optional)
- Step 7: Beginning the Repackaging Process

Step 1: Selecting the Repackaging Method

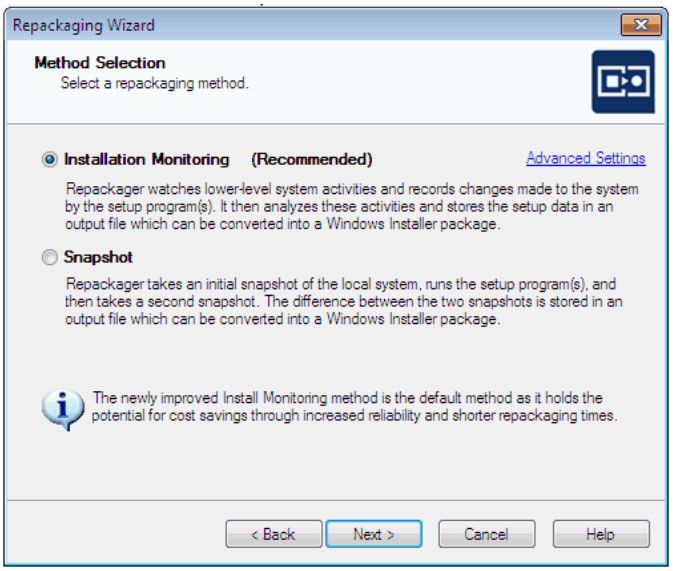
In this step, you launch the Repackaging Wizard and select the **Snapshot** repackaging method.

Task ***To select a repackaging method:***

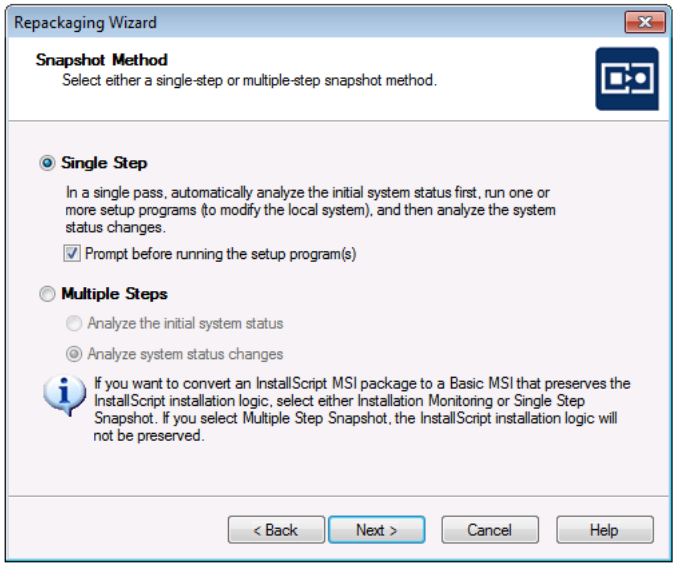
1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.



2. Click **Next**. The Method Selection Panel opens.



3. Select **Snapshot** and click Next. The Snapshot Method panel opens.



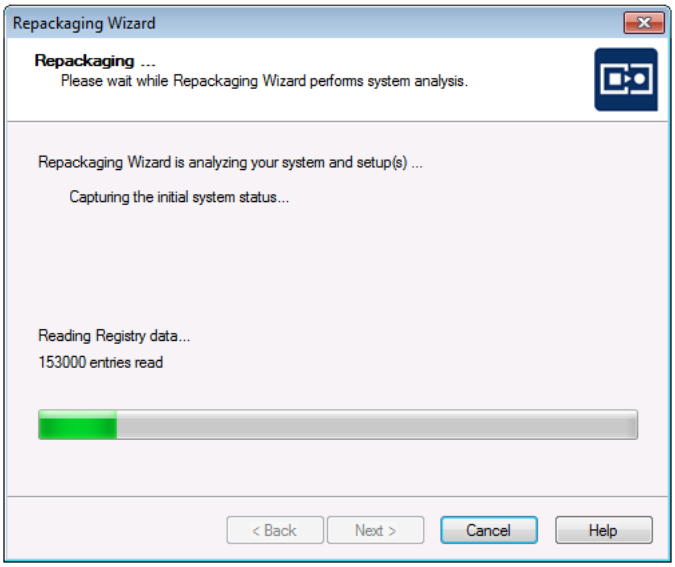
4. On the Snapshot Method panel, select **Multiple Steps**. The **Analyze the initial system status** option is enabled.
5. Select the **Analyze the initial system status** option.
6. Continue with Step 2: Initial Analysis.

Step 2: Initial Analysis

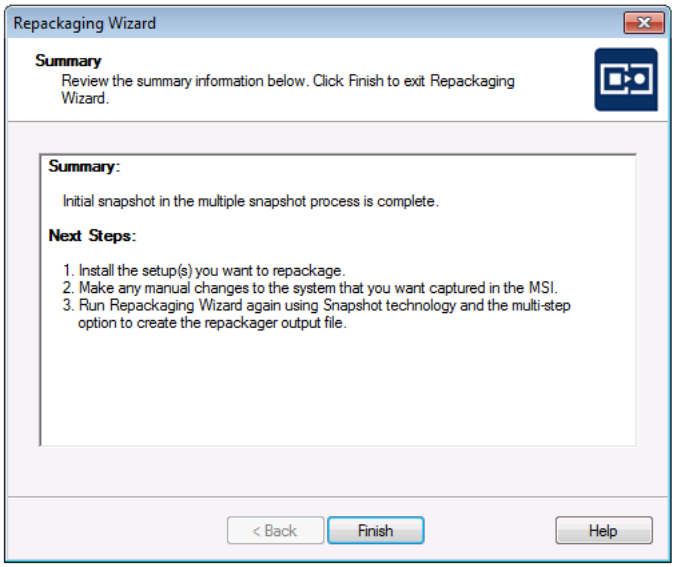
In this step, the Repackaging Wizard takes an initial snapshot of your system.

Task ***To perform initial analysis:***

1. On the Snapshot Method panel, click **Next**. The Repackaging Panel of the Repackaging Wizard opens, displaying the progress of the initial system status capture.



When Repackager finishes capturing the initial system status, the **Summary** panel opens, prompting you to install the application you are repackaging.



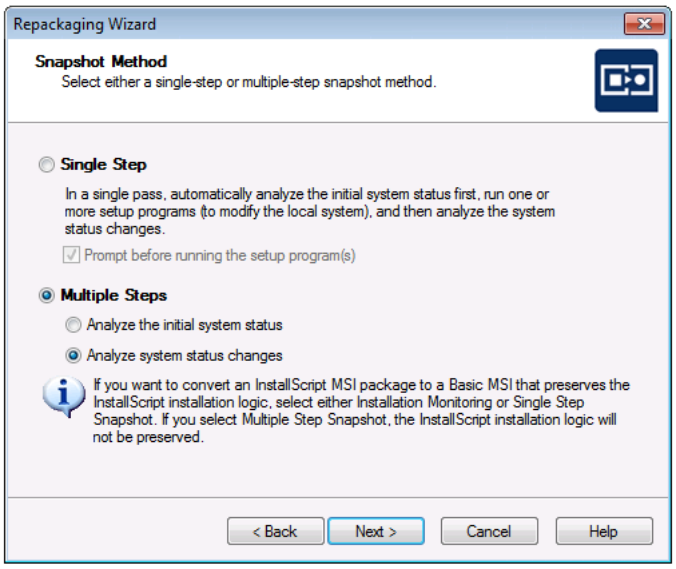
2. Click **Finish** to close the Repackaging Wizard.
3. Continue with Step 3: Install Setup and Make Manual System Changes.

Step 3: Install Setup and Make Manual System Changes

In this step, you will manually launch the installation of the application you are repackaging, and then you will make any manual changes to the system that you want captured in the Windows Installer package.

Task ***To install setup and make manual system changes:***

1. Launch the installation program of the application you are repackaging.
2. Follow the prompts until the installation has completed.
3. When the installation is complete, make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.
4. Launch the Repackaging Wizard again. The **Welcome Panel** opens.
5. Click **Next**. The **Method Selection Panel** opens.
6. Select **Snapshot** and click **Next**. The **Snapshot Method Panel** opens with **Multiple Steps** already selected, and the **Analyze system status changes** option now enabled and selected.



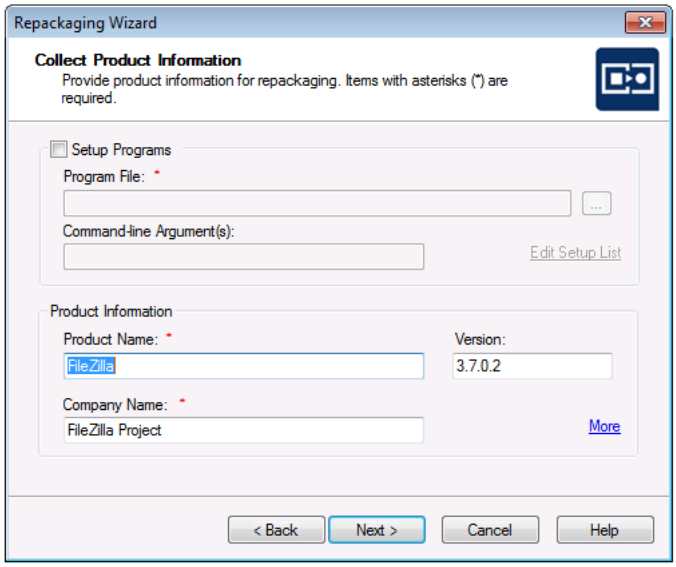
7. Continue with Step 4: Entering Product Information.

Step 4: Entering Product Information

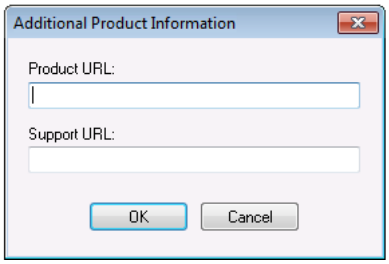
In this step, you will enter product information for the application that you just installed.

Task ***To enter product information:***

1. On the **Snapshot Method Panel**, click **Next**. The **Collect Product Information Panel** opens. Because you are now performing the second step of a multiple-step Snapshot, the **Setup Programs** area is disabled (because you have already installed the application you are repackaging).



2. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
3. If you want to associate websites with this installation package, perform the following steps:
   a. Click the **More** link. The **Additional Product Information** dialog box opens.



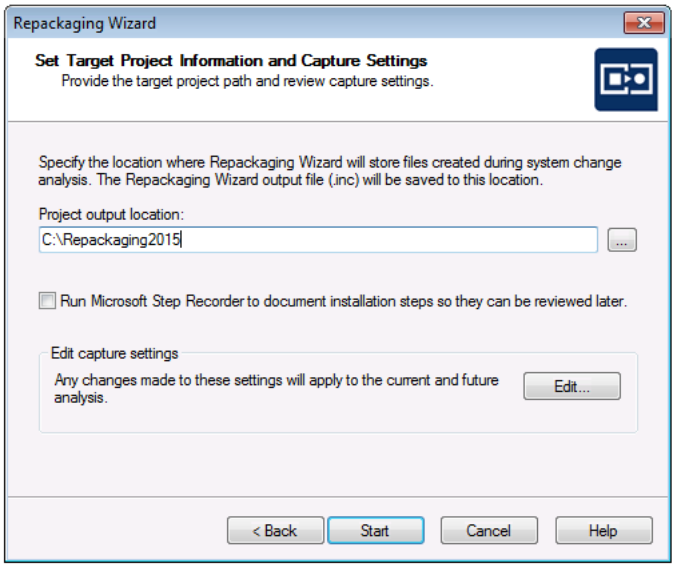
   b. Enter the **Product URL** and **Support URL** for the application you are repackaging.
   c. Click **OK**.
4. Continue with Step 5: Set Target Project Information.

Step 5: Set Target Project Information

In this step, you identify the location where you want files created by Repackager to be stored.

Task ***To set target project information and capture settings:***

1. On the **Collect Product Information Panel**, click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 6: Set Capture Settings (Optional).

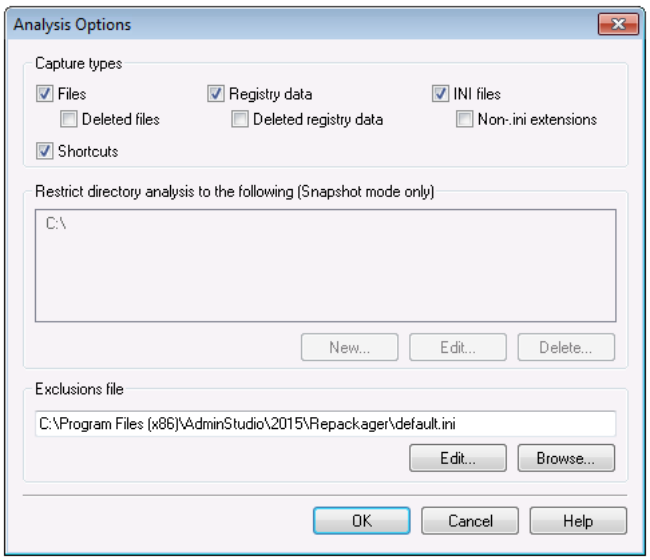
Step 6: Set Capture Settings (Optional)

From the **Set Target Project Information and Capture Settings Panel**, you can specify capture types for the repackaging session such as files, .ini files, shortcuts, and Registry data. You can also restrict directory analysis to specific directories, which can significantly improve repackaging performance.

- **If you want to modify the default capture settings**, perform the following steps.
- **If you do not want to modify the default capture settings**, click **Next** and continue with Step 7: Beginning the Repackaging Process.

Task ***To modify capture settings:***

1. On the **Set Target Project Information and Capture Settings Panel**, click **Edit**. The **Analysis Options** dialog box opens.



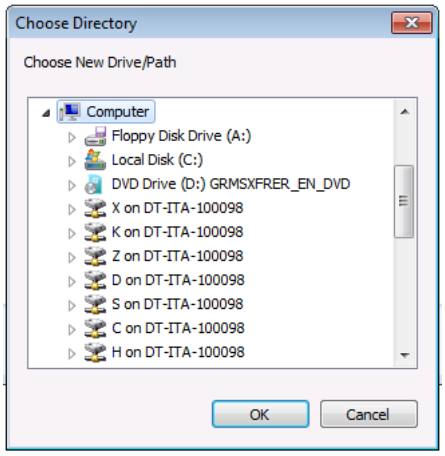
Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

2. Select the capture types that you want to use for this repackaging session:
   - **Files**
   - **Deleted files**

     Note • *If you select this option, deleted files will be displayed on the Deleted Files View of the Repackager interface.*
   - **INI files**
   - (INI files with) **Non-.ini extensions**
   - **Shortcuts**
   - **Registry data**
   - **Deleted registry data**

     Note • *If you select this option, deleted registry entries will be displayed on the Deleted Registry Entries View of the Repackager interface.*
3. If you want to restrict directory analysis to specific directories, first select the `C:\` in the **Restrict directory analysis to the following** list and click **Delete**. You will be prompted to confirm the deletion.

4. Next, to indicate the specific directories, click **New**. The **Choose Directory** dialog box opens.



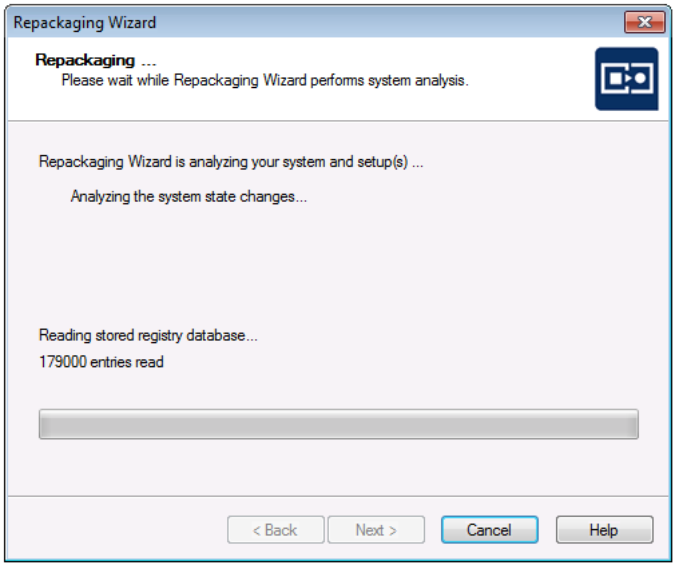
5. Select a directory to include and click **OK**. The selected directory is now listed on the **Analysis Options** dialog box. Repeat this process to add additional directories.
6. If you want to modify an existing restriction, or delete a restriction, select the listed directory and click **Edit** or **Delete**.
7. Click **OK** to return to the **Set Target Project Information and Capture Settings Panel**.
8. Continue with Step 7: Beginning the Repackaging Process.

Step 7: Beginning the Repackaging Process

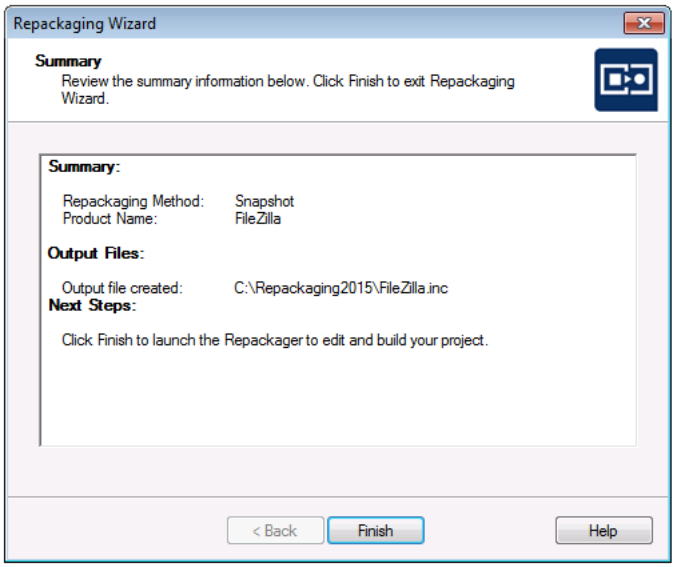
In this step you will begin the repackaging process.

Task ***To begin the repackaging process:***

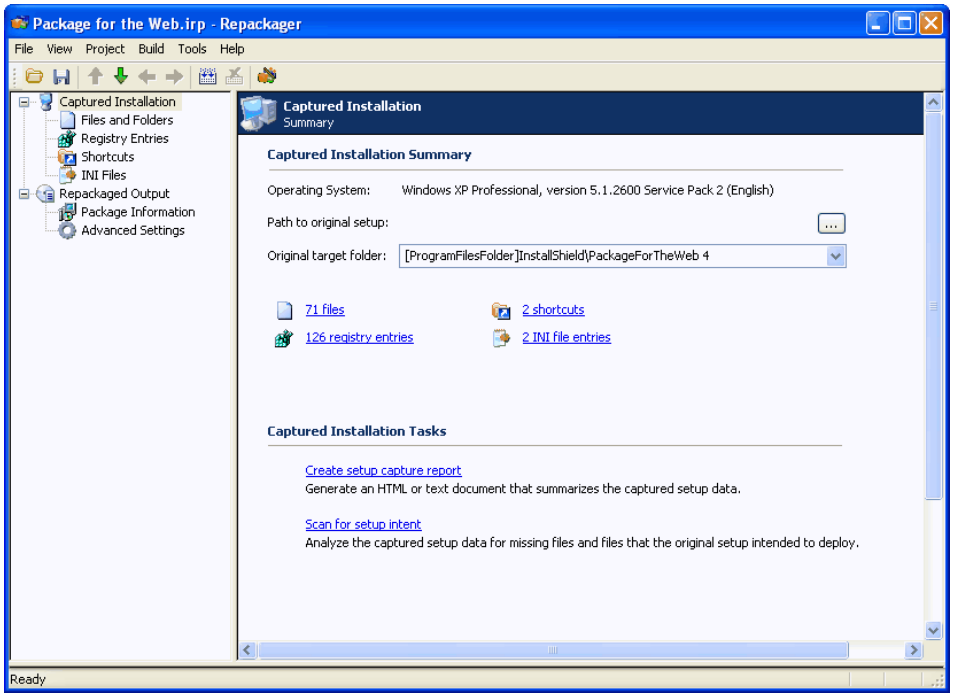
1. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the system state changes.



When the Repackaging Wizard has finished analyzing the system state changes and creating the Repackager project, the Summary Panel opens, providing confirmation that the repackaging was successful and listing the location of your new Repackager project.



2. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.



3. Continue with the instructions in Working With Repackager Projects.

Performing Single Step Snapshot Repackaging

To repackage an installation using the **Single Step Snapshot** method, perform the following steps:

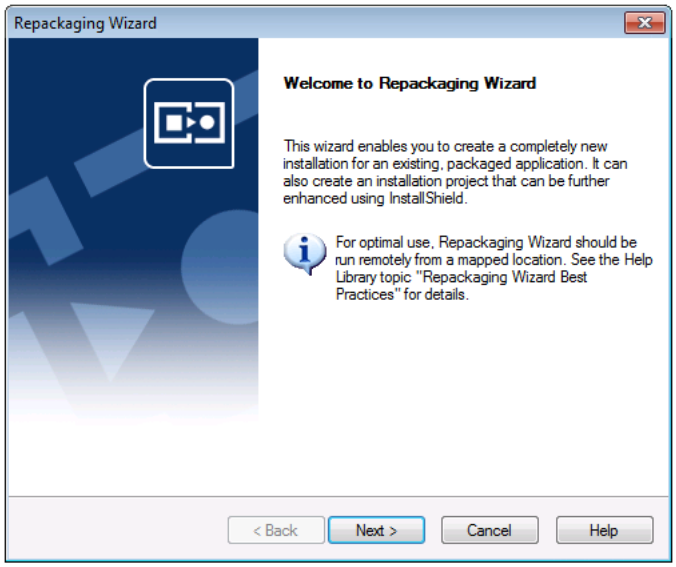
- Step 1: Selecting the Repackaging Method.
- Step 2: Collecting Product Information
- Step 3: Set Target Project Information
- Step 4: Set Capture Settings (Optional)
- Step 5: Beginning the Repackaging Process

Step 1: Selecting the Repackaging Method

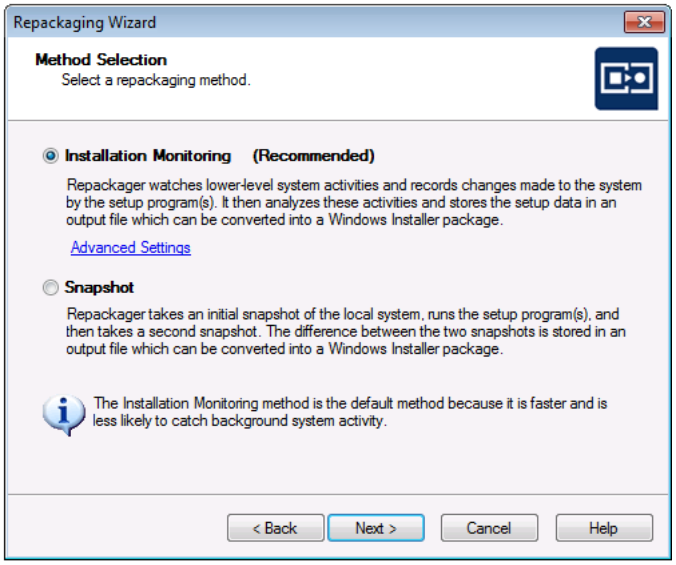
In this step, you launch the Repackaging Wizard and select the **Snapshot** repackaging method.

Task ***To select a repackaging method:***

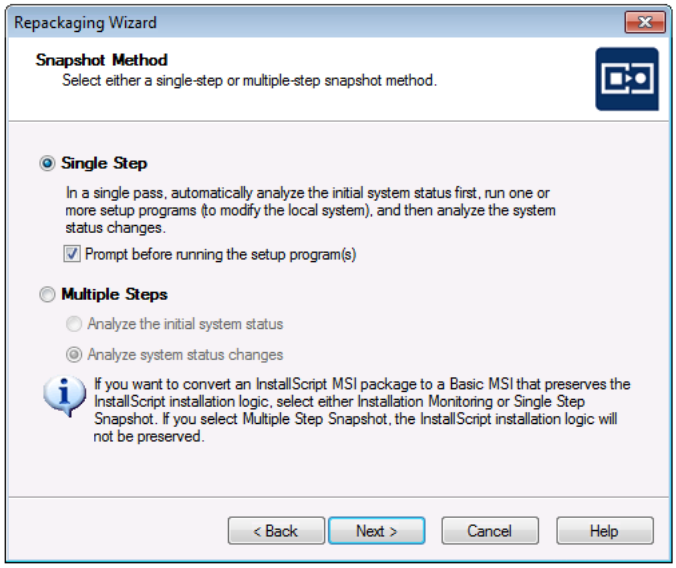
1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.



2. Click **Next**. The Method Selection Panel opens.



3. Select **Snapshot** and click **Next**. The **Snapshot Method** panel opens.



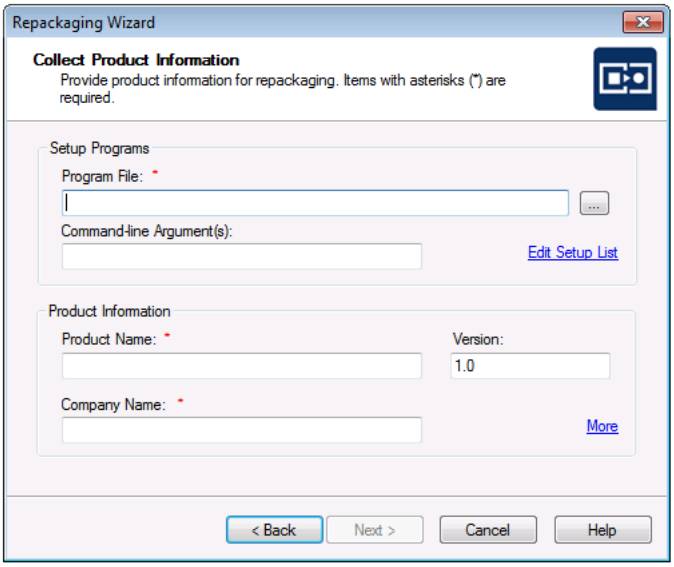
4. On the Snapshot Method panel, select **Single Step**.
5. If you want to be prompted before the selected setup program is launched, select the **Prompt before running the setup program(s) option**. If you do not select this option, the setup program will automatically be launched as soon as the Repackaging Wizard has finished analyzing the system status.
6. Continue with Step 2: Collecting Product Information.

Step 2: Collecting Product Information

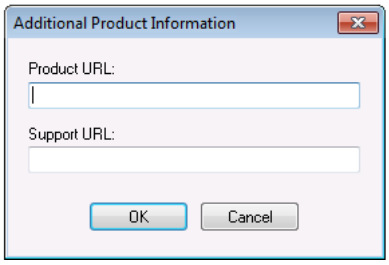
In this step, you will specify the installation you want to repackage and enter any command-line arguments to be used when the installation is run.

Task ***To enter product information:***

1. On the Snapshot Method panel, click **Next**. The Collect Product Information Panel opens with the **Setup Programs** and **Product Information** areas enabled.



2. Click the Browse (...) button next to the **Program File** field and select the installation program that you are repackaging.
3. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when the installation is run.
4. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
5. If you want to associate websites with this installation package, perform the following steps:
   a. Click the **More** link. The **Additional Product Information** dialog box opens.



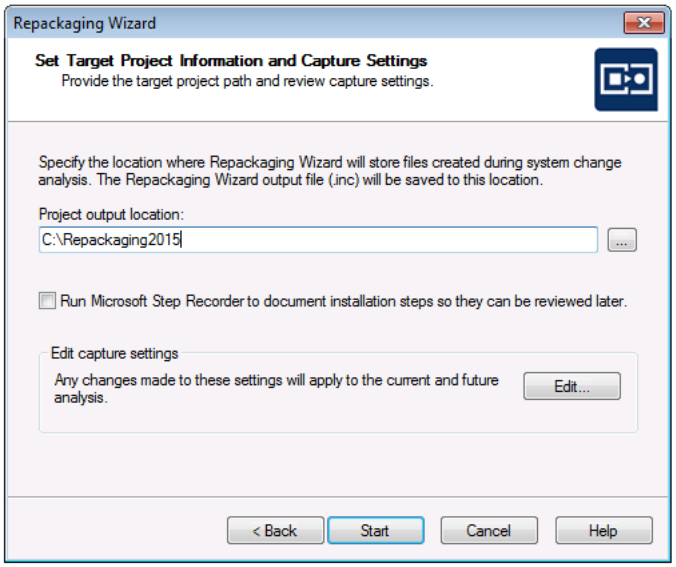
   b. Enter the **Product URL** and **Support URL** for the application you are repackaging.
   c. Click **OK**.
6. Continue with Step 3: Set Target Project Information.

Step 3: Set Target Project Information

In this step, you identify the location where you want files created by Repackager to be stored.

Task ***To set target project information:***

1. On the Collect Product Information Panel, click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 4: Set Capture Settings (Optional).

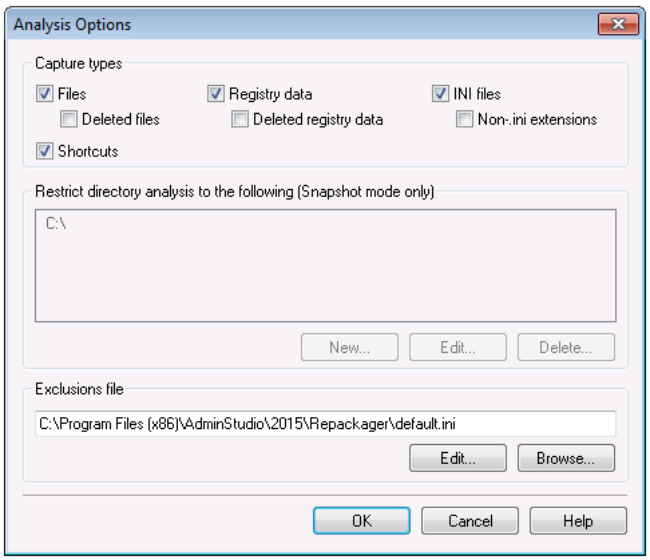
Step 4: Set Capture Settings (Optional)

From the **Set Target Project Information and Capture Settings Panel**, you can specify capture types for the repackaging session such as files, .ini files, shortcuts, and Registry data. You can also restrict directory analysis to specific directories, which can significantly improve repackaging performance.

- **If you want to modify the default capture settings**, perform the following steps.
- **If you do not want to modify the default capture settings**, click **Next** and continue with Step 7: Beginning the Repackaging Process.

Task ***To modify capture settings:***

1. On the **Set Target Project Information and Capture Settings Panel**, click **Edit**. The **Analysis Options** dialog box opens.



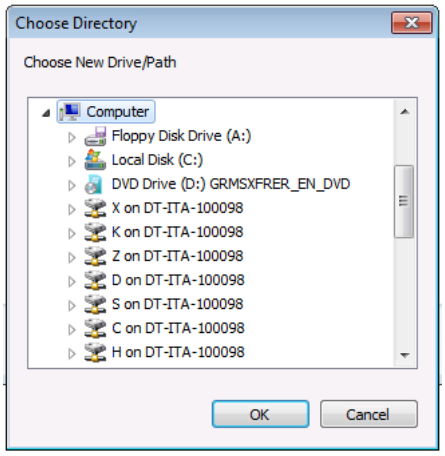
Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

2. Select the capture types that you want to use for this repackaging session:
   - **Files**
   - **Deleted files**

     Note • *If you select this option, deleted files will be displayed on the Deleted Files View of the Repackager interface.*
   - **INI files**
   - (INI files with) **Non-.ini extensions**
   - **Shortcuts**
   - **Registry data**
   - **Deleted registry data**

     Note • *If you select this option, deleted registry entries will be displayed on the Deleted Registry Entries View of the Repackager interface.*
3. If you want to restrict directory analysis to specific directories, first select the `C:\` in the **Restrict directory analysis to the following** list and click **Delete**. You will be prompted to confirm the deletion.

4. Next, to indicate the specific directories, click **New**. The **Choose Directory** dialog box opens.



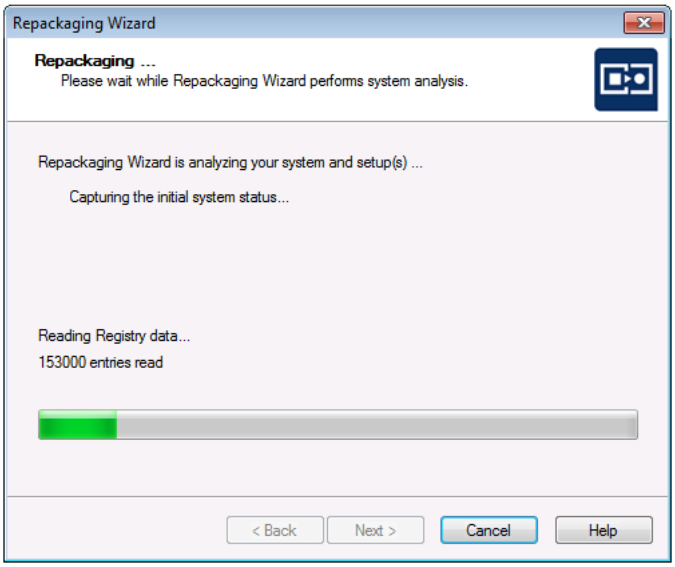
5. Select a directory to include and click **OK**. The selected directory is now listed on the **Analysis Options** dialog box. Repeat this process to add additional directories.
6. If you want to modify an existing restriction, or delete a restriction, select the listed directory and click **Edit** or **Delete**.
7. Click **OK** to return to the **Set Target Project Information and Capture Settings Panel**.
8. Continue with Step 7: Beginning the Repackaging Process.

Step 5: Beginning the Repackaging Process

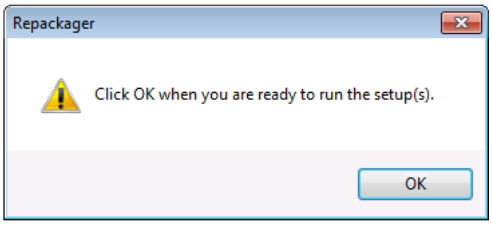
In this step you will begin the repackaging process.

Task ***To begin the repackaging process:***

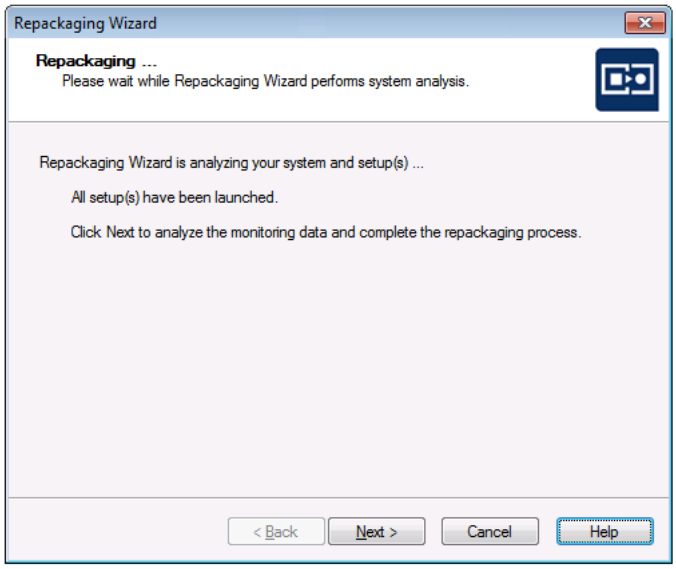
1. To begin the repackaging process, click **Start** on the **Set Target Project Information and Capture Settings Panel**. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status.



Depending upon whether you chose the **Prompt before running the setup program(s)** option on the Snapshot Method Panel, either the installation that you selected will start or you will be prompted to start it.

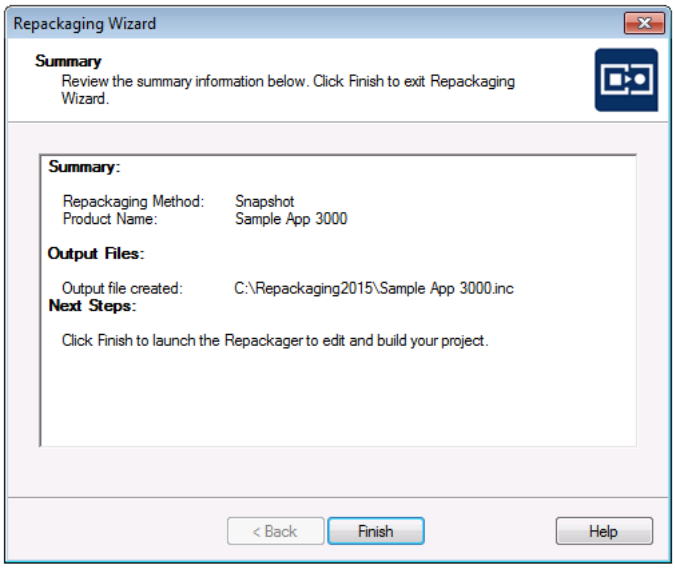


2. Install the application by following the prompts until the installation has completed.
3. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.



4. When you are ready to complete the repackaging process, click **Next**. The Repackaging Wizard then analyzes the system and setup data that it collected.

   Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful and listing the location of the Repackager project that was just created.



5. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.
6. Continue with the instructions in Working With Repackager Projects.

Repackaging an InstallScript MSI Setup to a Basic MSI Setup

InstallScript MSI installations use a Windows Installer database for storage of all file/registry information, but the actual user interface, and much of the installation logic is driven by the InstallScript engine via a `setup.exe` file. This type of installation architecture can cause difficulties during deployment, such as:

- inability to customize or transform the application
- inability to perform conflict detection
- inability to suppress the user interface
- difficulty patching or upgrading the application

Also, if an InstallScript MSI installation is repackaged using traditional methods (OS Snapshot or Installation Monitoring), significant platform-specific or custom installation, maintenance, and uninstallation logic, and user interface information is lost because those methods only record the installation activities for the specific platform used during repackaging.

Therefore, it is recommended that you use InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package.

Note • *If you want to convert an InstallScript MSI package to a Basic MSI package that preserves the InstallScript installation logic, and you are using the Snapshot method, you must select Single Step rather than Multiple Steps. If you select Multiple Steps, the InstallScript installation logic will not be preserved.*

Task ***To convert an InstallScript MSI Setup to a Basic MSI Setup with InstallScript support:***

1. Launch the **Repackaging Wizard** from Repackager. The **Welcome Panel** opens.
2. Click Next. The **Method Selection Panel** opens.
3. Select a **repackaging method**: Installation Monitoring or Snapshot.
4. Click Next. If you selected **Snapshot** on the **Method Selection Panel**, the **Snapshot Method Panel** appears. (If you selected **Installation Monitoring**, skip to Step 6.)
5. Select **Single Step** and click **Next**. The **Collect Product Information Panel** opens.

   Caution • *Because you are converting an InstallScript MSI package to a Basic MSI package with InstallScript support, you must select the Single Step Snapshot method (or use the Installation Monitoring method). If you select Multiple Step Snapshot, the InstallScript installation logic will not be preserved.*
6. On the **Collect Product Information Panel**, select the InstallScript MSI setup file and enter other product information.

   Caution • *While it is possible to click the Edit Setup List button and select additional setups, because you are converting an InstallScript MSI package, do not select additional setups.*
7. Click Next. Repackager will automatically determine if this is an InstallScript-based setup. If it is an InstallScript-based setup, the **InstallScript MSI Identified Panel** opens, informing you that the Repackaging Wizard has identified this setup as being an InstallScript MSI setup and prompting you to use InstallScript Scan to convert this setup.
8. Select Yes and click Next. The **InstallScript MSI Conversion Output Panel** opens.
9. In the Project Path to store files field, specify the location where you want the Repackaging Wizard to store files created during InstallScript Scan Analysis and where it will save the converted MSI package.

Note • *To specify capture types for the repackaging session, click the Edit button to access the* ***Analysis Options*** *dialog box.*

10. Click Next. The **Repackaging Panel** appears, displaying the progress of the repackaging operation.
11. Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.
12. Click Finish to launch the Repackager to edit and build your project. See Working With Repackager Projects.

Running the Repackaging Wizard from the Command Line

To run the Repackaging Wizard from the command line, perform the following steps.

Task ***To run the Repackaging Wizard from the command line:***

1. Open a command-line prompt.
2. Type `Repack.exe` followed by any command-line options you want to pass. See Repackaging Wizard Command-Line Options.
3. Press Enter.

An example of a typical command line is as follows:

```
Repack.exe -app Setup.exe -o C:\MyRepackagedApps\Output
     -pp SomeApp -cs Custom -cf MyOptions.ini -sb
```

In the above example, the following options are used:

Table 8-9 • Repackager Command-Line Options used in Example

Option	Description
-app	Specifies the name of the setup.
-o	Specifies the location of the output directory
-pp	Specifies the name of the product (and the name of the Repackager output file).
-cs	Specifies the name of the custom analysis options file to use.
-cf	Name of the analysis options file to use.
-sb	Allows you to run Repackager silently, with no user interaction.

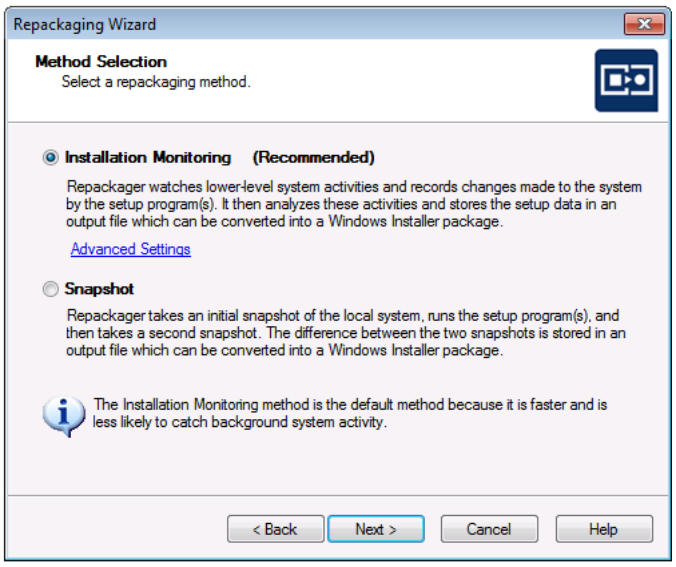
Repackaging a Windows Installer (.msi) Package

While it is not recommended that you repackage a Windows Installer (`.msi`) package, it sometimes may be necessary to repackage a Windows Installer package in order to convert it to a virtual package (perhaps due to the use of custom actions or other features that are not supported in application virtualization).

To repackage a Windows Installer (`.msi`) package, perform the following steps:

Task ***To repackage a Windows Installer package:***

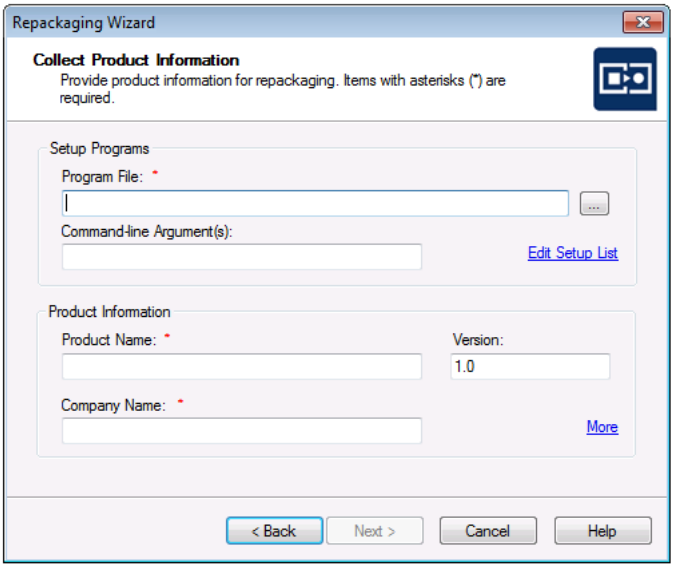
1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.
2. Click **Next**. The **Method Selection Panel** opens.



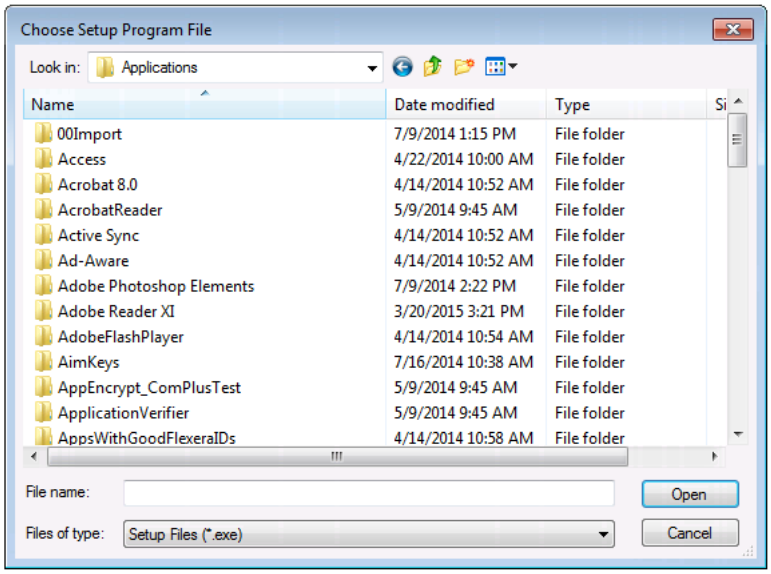
3. Select **Installation Monitoring** and click **Next**. The **Collect Product Information Panel** opens.

Note • *The **Installation Monitoring** method is recommended, but you may also choose the **Snapshot** method when repackaging a Windows Installer package. The **Installation Monitoring** method was used in the instructions that follow. For instructions on using the **Snapshot** method, see Repackaging Using the Snapshot Method*

Note • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.*

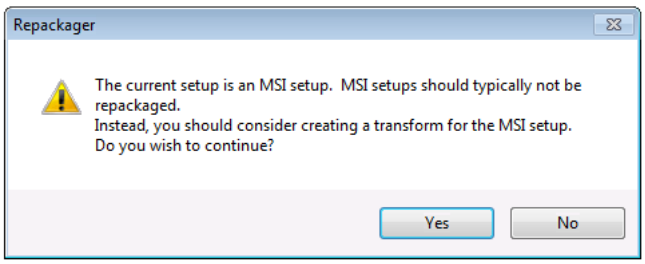


4. Click the Browse (...) button next to the **Program File** field to open the **Choose Setup Program File** dialog box.

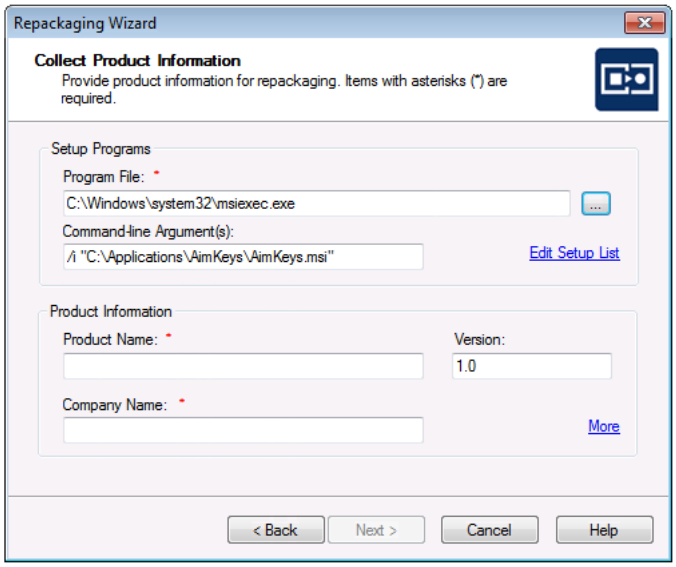


5. From the **Files of type** list, select **All Files (*.*)**. All files in the selected directory are listed.

6. Click **Open** and select the Windows Installer package (.msi) that you are repackaging. A message appears warning you that MSI setups should not typically be repackaged.



7. Click **Yes** to close the message. Several fields in the Collection Product Information panel have been populated with the commands necessary to repackage a Windows Installer package.

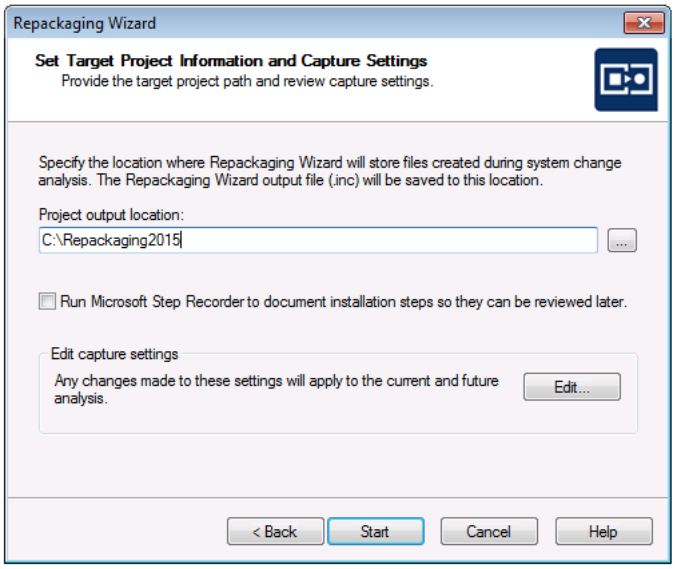


The following information was filled in:

Field	Entry
Program File	`C:\WINDOWS\system32\msiexec.exe`
Command line Argument(s)	`/i "C:\DIRECTORYPATH\PACKAGENAME.msi"`

Caution • *Do not edit the entries in the Program File or Command line Argument(s) fields.*

8. In the **Product Information** area, make entries in the **Product Name**, **Version**, and **Company Name** fields.
9. Click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.

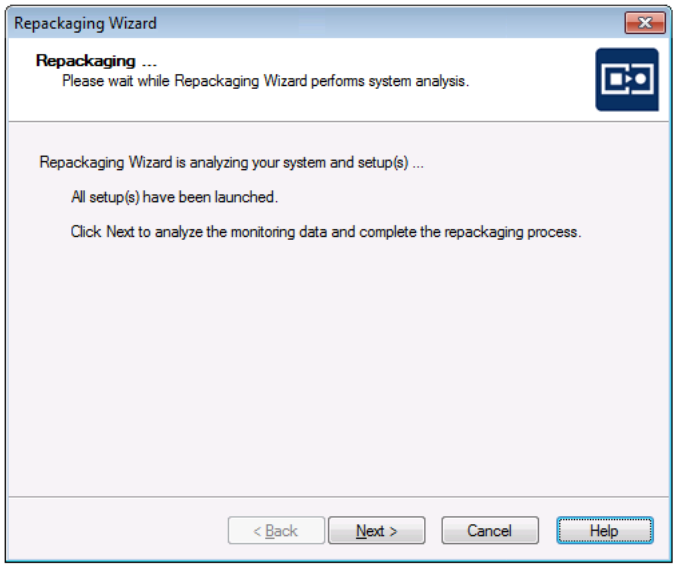


10. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

    You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

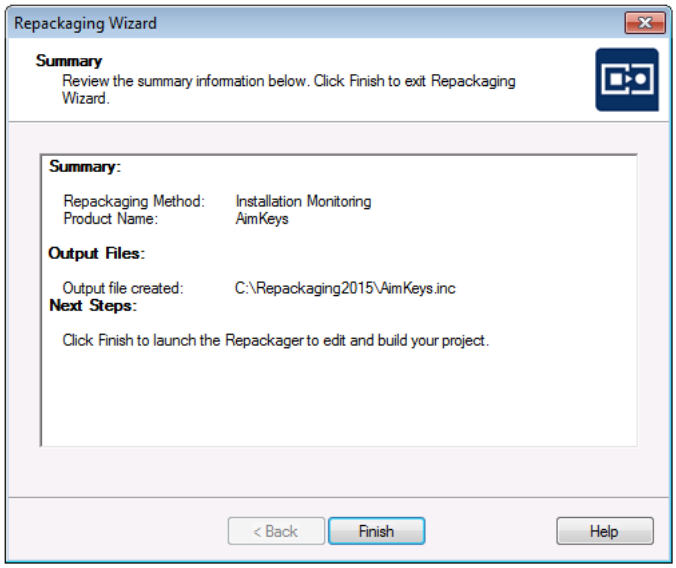
11. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status. Then, the selected setup program will be launched.

12. Follow the prompts until the installation has completed. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.



13. When you are ready to complete the repackaging process, click **Next**. The Repackaging Wizard then analyzes the system and setup data that it collected.

    Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.



14. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.

15. Continue with the instructions in Working With Repackager Projects.

Documenting Repackaging Steps Using the Microsoft Step Recorder Tool

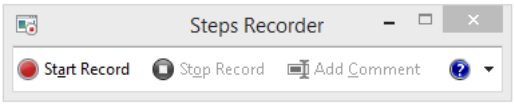
You can use the Microsoft Steps Recorder documentation tool with the Repackaging Wizard to automatically record the step-by-step actions that occur during repackaging. This information, which is saved in a web archive (`.mht`) file, includes a text description of where you clicked on each screen, along with a screen capture for each click.

To enable this option, select the **Run Microsoft Step Recorder to document installation steps so they can be reviewed later** option on the **Set Target Project Information and Capture Settings** panel of the Repackaging Wizard.

Task ***To use the Microsoft Steps Recorder during repackaging:***

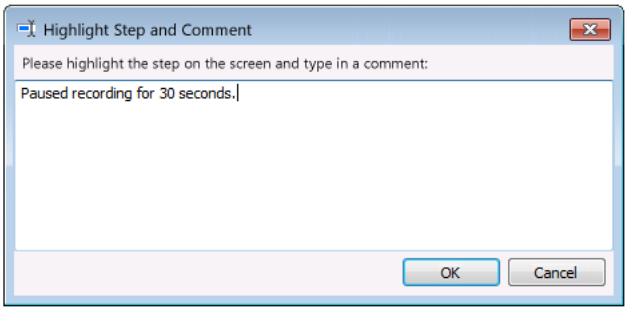
1. Launch the Repackaging Wizard.
2. On the **Method Selection** panel, select the desired method, as described in Installation Monitoring Method and Snapshot Method.
3. Proceed with repackaging, per the selected method, as described in the following help topics:
   - Repackaging Using the Installation Monitoring Method
   - Repackaging Using the Snapshot Method

4. On the **Set Target Project Information and Capture Settings** panel, select the **Run Microsoft Step Recorder to document installation steps so they can be reviewed later** option.
5. Click **Start** to begin the repackaging process. When the installer is launched, the **Steps Recorder** dialog box opens and recording automatically begins.



The **Steps Recorder** dialog box includes the following controls:

- **Pause Record**—Click to pause the recording. You would use this if you wanted to pause the repackaging process and use another application on your computer. If you do not pause the recording, all actions you take on the machine, whether or not they pertain to repackaging, will be recorded.
- **Stop Record**—Click to stop the recording.
- **Add Comment**—Click to pause the recording and open the **Highlight Step and Comment** dialog box where you can enter a comment. This comment will appear in the generated output file.



- **Elapsed time**—The time elapsed since the recording started is listed.

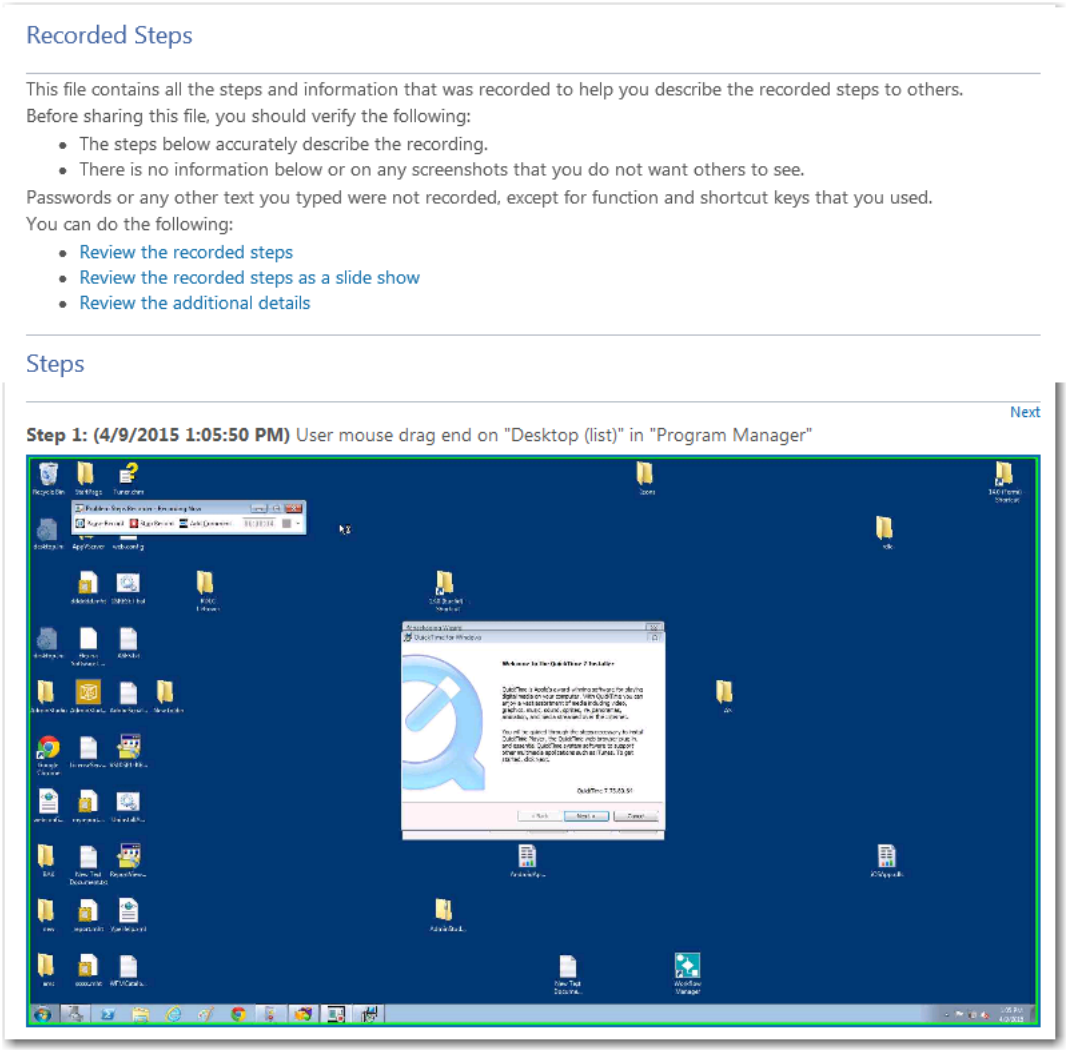
6. Click through the installer until it is completed.
7. Complete the panels on the Repackaging Wizard, per the selected method.
8. When repackaging is complete, open the Repackaged Output folder and locate the following web archive (`.mht`) file:

   `InstallerName``_Recording_`*`YYYYMMDD_TIME`*`.mht`

   For example:

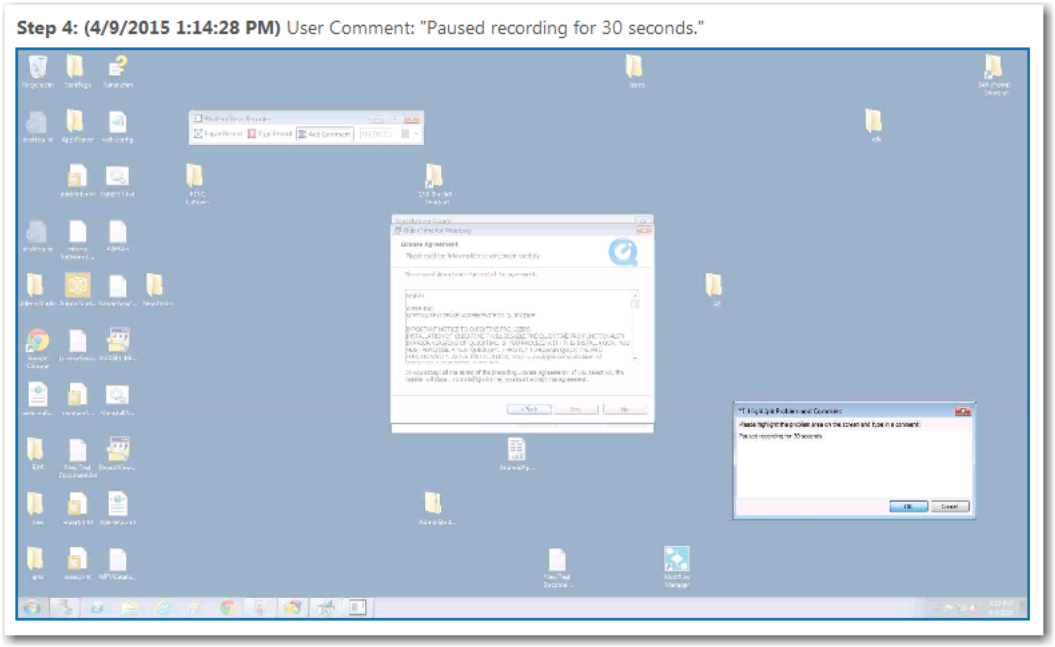
   `QuickTime_Recording_20150409_1015.mht`

9. Double-click the file to open it. The file opens in a browser window.
10. In the **Steps** section, scroll down to view all of the steps that you performed during repackaging along with screen captures of each step.

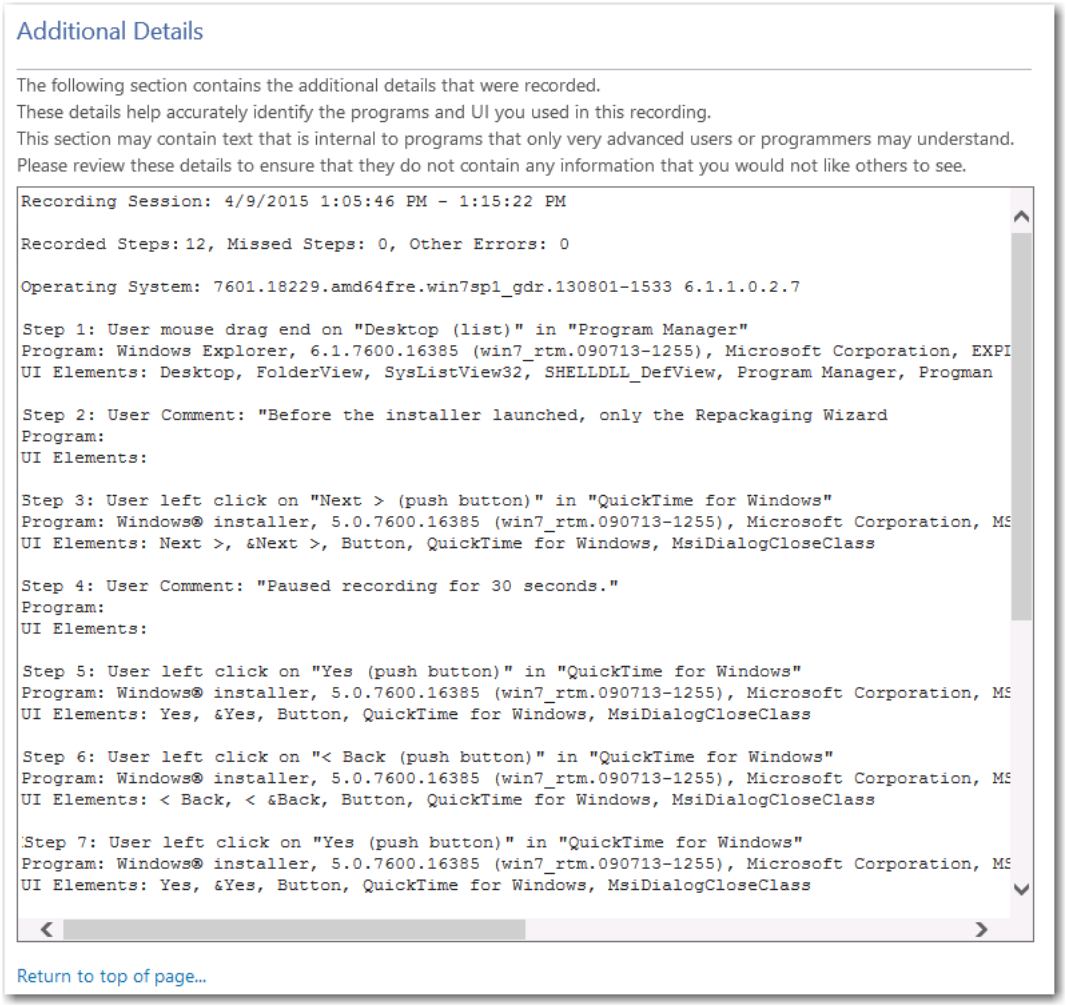


Tip • *If you want to view all of the screens as a slide show instead of scrolling through them, click* ***Review the recorded steps as a slide show****.*

If you entered any comments, they are listed in the file along with a screen capture of the **Highlight Step and Comment** dialog box.



11. Review the information in the **Additional Details** area, which contains a text description of the steps that were taken, along with information that is internal to the repackaged application.



Viewing the Recorded Archive File from Application Catalog

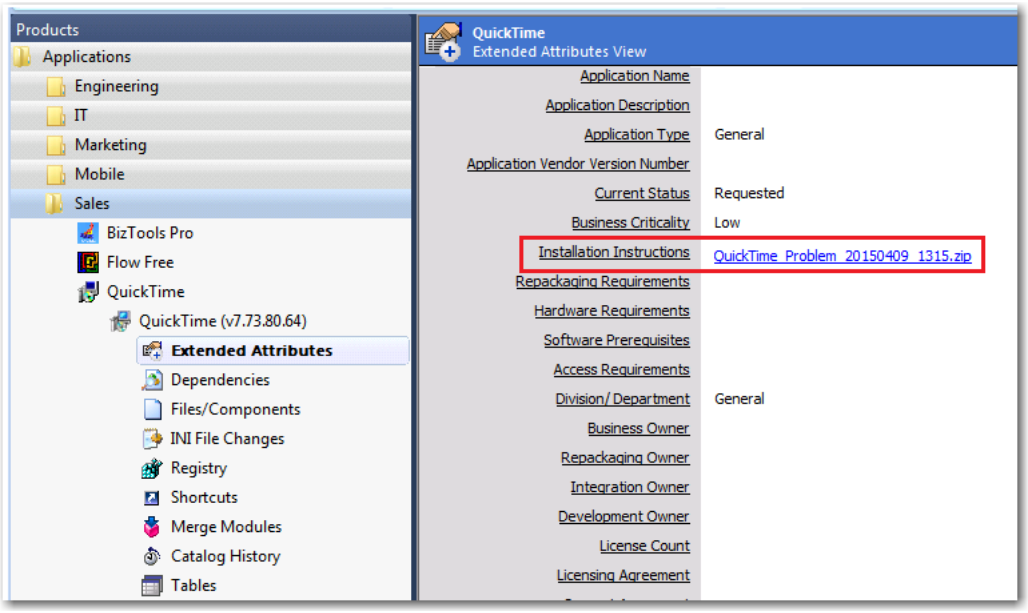
When you build a Windows Installer package using the Repackager project that was created during repackaging, this recorded web archive file will be copied to the Windows Installer package output folder.

Then, when you import this Windows Installer package into the Application Catalog, the recorded web archive file (or files) will also be imported.

To view a recorded web archive file in Application Catalog, perform the following steps:

Task ***To view a recorded web archive file in Application Catalog:***

1. Open Application Catalog.
2. Locate the Windows Installer package and expand its package node.
3. Select the **Extended Attributes** subnode. The **Extended Attributes View** opens.



4. Next to **Installation Instructions**, click the name of the listed ZIP file. The ZIP file will open and display a list of the `.mht` files it contains.

5. Double-click on the `.mht` file that you want to view.

Repackaging Wizard Reference

This section describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackaging Wizard. The help topics in the Repackager Reference are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

Reference information is organized as follows:

Table 8-10 • Organization of Repackager Reference Section

Section	Description
Repackaging Wizard	This section provides a panel-by-panel description of the Repackaging Wizard.
Additional Repackaging Wizard Dialog Boxes	This section describes the dialog boxes that can be accessed from the Repackaging Wizard.
Repackaging Wizard Command-Line Options	This section lists the command-line options that are supported by the Repackaging Wizard.
Reboot Handling in the Repackaging Wizard	This section describes how the Snapshot Method and Installation Monitoring Method handle required reboots during repackaging.

Repackaging Wizard

Repackager provides the Repackaging Wizard to convert a legacy setup into a Repackager project. Using this Wizard, you can select the repackaging method (either Snapshot or Installation Monitoring), specify the setup(s) you want to repackage, and run the setup(s). When the Repackaging Wizard has finished its analysis, Repackager automatically creates a Repackager project (`.irp`) file, which can be modified in Repackager. You can then convert this file to an InstallShield Editor project (`.ism`) for further editing, or convert it directly to a Windows Installer package (`.msi`).

The Repackaging Wizard includes the following panels:

- Welcome Panel
- Method Selection Panel
- Snapshot Method Panel
- Collect Product Information Panel
- InstallScript MSI Identified Panel
- Set Target Project Information and Capture Settings Panel
- InstallScript MSI Conversion Output Panel
- Repackaging Panel
- Summary Panel
- Additional Repackaging Wizard Dialog Boxes

Welcome Panel

The Welcome panel appears when you first launch the Repackaging Wizard, providing some introductory information about the use of the Wizard, including that it is for use with traditional (non-Windows Installer-based) installations.

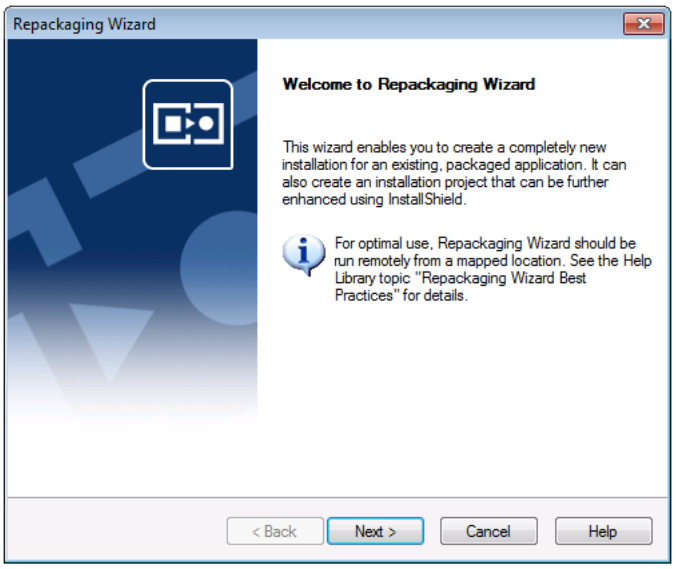


Figure 8-1: Repackaging Wizard Welcome Panel

Method Selection Panel

From the Method Selection panel, select the method(s) you want to use for repackaging.

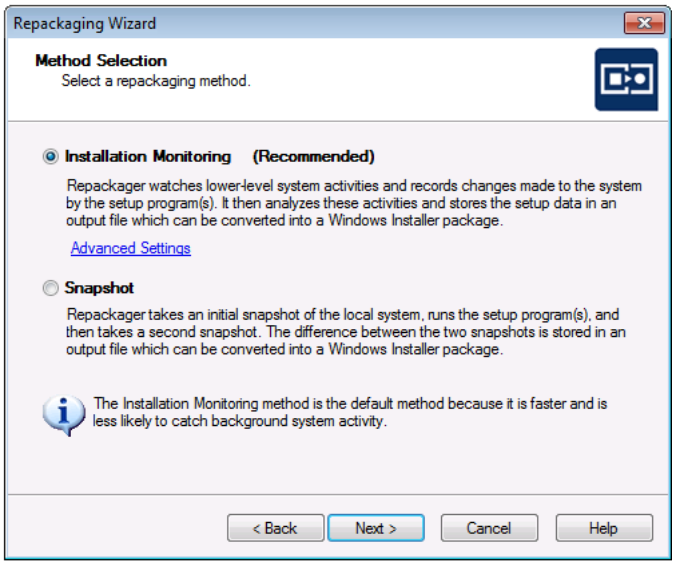


Figure 8-2: Repackaging Wizard Method Selection Panel

The available choices include:

Table 8-11 • Method Selection Panel Options

Options	Description
Snapshot	The Snapshot method involves taking system snapshots before and after an installation, and then creating the Windows Installer package from the difference between them. Any configurations you make between snapshots is also included in the generated Windows Installer package.

Table 8-11 • Method Selection Panel Options

Options	Description
Installation Monitoring	Installation Monitoring watches all activities generated by an installation, and then determines the files, .ini files, registry entries and shortcuts that should be included in the generated Windows Installer package. Installation Monitoring is significantly faster than the Snapshot repackaging method. If there are services running on the machine that have nothing to do with the installation being repackaged, click the **Advanced Settings** link to open the Excluded Processes Dialog Box, where you can choose to exclude those processes. ***Edition*** • *The Installation Monitoring Method is included with AdminStudio Professional and Enterprise Editions.* ***Tip*** • *If you know that the installation that you are capturing is from a self-extracting* **.exe** *file and if you want to use the* ***Installation Monitoring*** *method, you should click* ***Advanced Settings*** *and add the name of that* **.exe** *file to the excluded processes list.*

System Changes Captured by Repackager

Regardless of the repackaging method used, Repackager captures system changes made to the following:

- Application Paths
- Environment Variables
- Files
- INI Files
- NT Services
- ODBC Data Sources
- ODBC Drivers
- Printer Drivers
- Registry Entries
- Shortcuts

Snapshot Method Panel

The Snapshot Method Panel, which is only displayed if you use the snapshot technology, allows you to specify the way in which you perform repackaging.

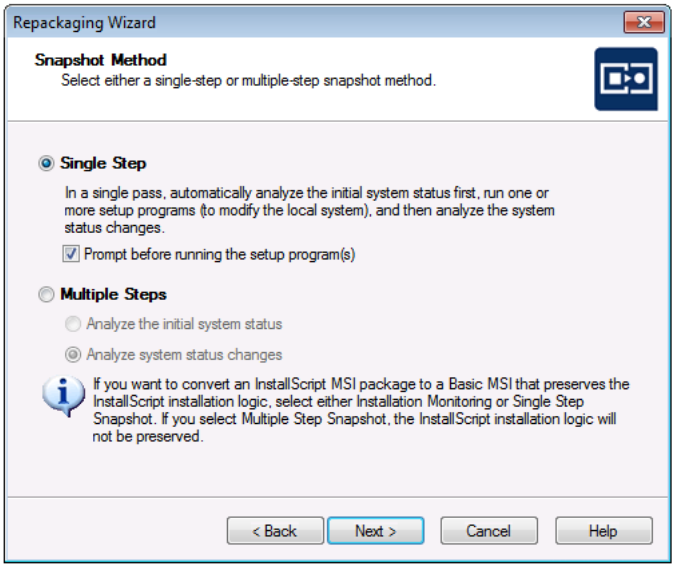


Figure 8-3: Repackaging Wizard Snapshot Method Panel

On the Snapshot Method Panel, you have the following two options:

Table 8-12 • Snapshot Method Panel Options

Option	Description
Single Step	Repackaging in a single step requires you specify at least one setup program to repackage. The Repackager first takes an initial system snapshot, then runs the setup program(s) you specify, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package. You also have the option of requiring the Repackager to prompt you before running the setup program(s), allowing you the opportunity to make changes to your system that you want included in the final package.
Multiple Steps	Repackaging in multiple steps allows you to run the Repackager to obtain an initial system snapshot, after which the Repackager exits. You can then perform any modifications to the system, such as changing configurations, running installations, and so forth. After making the necessary modifications, run the Repackager again to analyze system status changes. The difference between the second Repackager execution and the first results in the script file that ultimately can be converted into a Windows Installer package.

The single step method is very straightforward if you are repackaging applications and not performing many system changes. The multiple step method allows greater flexibility because a setup is not required. This allows you to capture system configurations within the Repackager output, and ultimately within a Windows Installer package. For example, you could modify the screen color depth and create an MSI package for just that configuration.

If Single Step is selected, the **Collect Product Information Panel** is displayed when you click Next. If Multiple Steps is selected and you are performing the initial snapshot, the Collect Product Information panel is displayed, but the Setup Programs area is disabled. If you are performing a system status change analysis, the **Repackaging Panel** appears when you click **Next**.

Collect Product Information Panel

The **Collect Product Information** panel gathers information necessary for repackaging the installation(s).

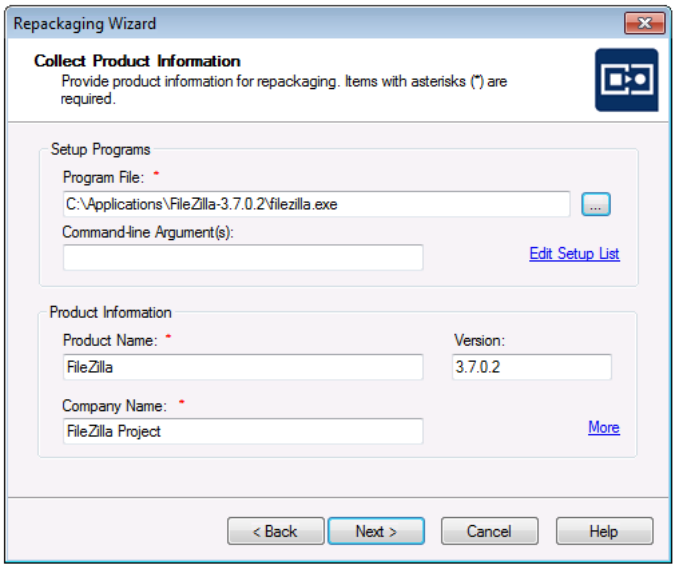


Figure 8-4: Repackaging Wizard Collect Product Information Panel

The information on the Collect Product Information Panel is divided into two sections: **Setup Programs** and **Product Information**.

Setup Programs Area

The Setup Programs area contains information about the setup you are repackaging. Repackager uses this information to launch the setup correctly following pre-analysis. The information collected includes:

Table 8-13 • Setup Programs Options

Properties	Description
Program File	The name and location of the setup executable. Click the Browse ... button to locate this file. This is a required field.
Command-Line Argument(s)	Any command-line arguments to be used when the setup is run.
Edit Setup List	Click to display the **Additional Setup Programs** dialog box, from which you can enter additional installations to repackage together with this installation. Additional setups share the same product name, version number, and company name in the repackaged installation. However, as you locate each additional setup to repackage, you can specify command-line parameters pertaining only to that setup. You can also specify the order in which the installations are run, should it be necessary.

Product Information Area

In the Product Information area, you identify the repackaged installation's **Product Name**, **Version Number**, and **Company Name**.

Table 8-14 • Product Information Options

Field	Description
Product Name	Enter the name for final repackaged installation. This could be the name of the original installation (for example, Tuner), the name of a collective group of products (for example, Microsoft Applications), or another name of your selection (for example, My Apps). This is a required field.
Version Number	Enter the version of the product.
Company Name	Enter the name of the company.

Product Support Information

If you want to associate websites with this installation, click the **More** link in the Product Information area to open the Additional Product Information dialog box, where you can enter the **Product URL** and **Support URL** for the application you are repackaging.

InstallScript MSI Identified Panel

This panel opens if the Repackaging Wizard identifies an installation as an InstallScript MSI installation created with InstallShield Editor, InstallShield DevStudio, or InstallShield Developer.

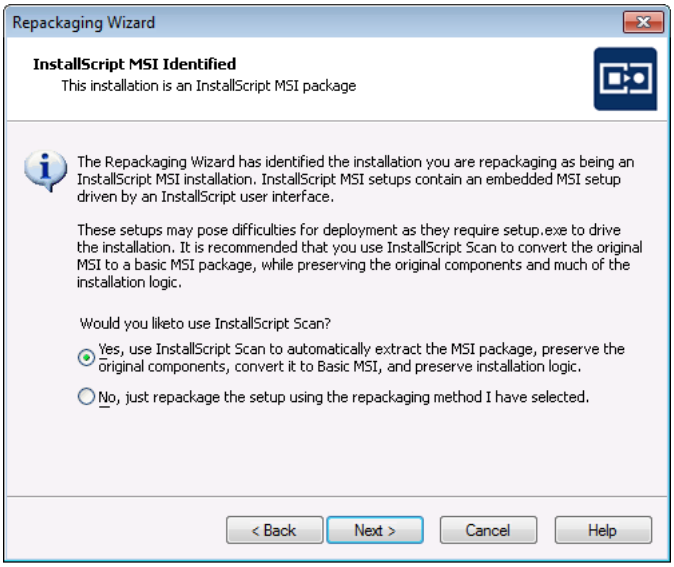


Figure 8-5: Repackaging Wizard InstallScript MSI Identified Panel

InstallScript MSI installations use a Windows Installer database for storage of all file/registry information, but the actual user interface, and much of the installation logic is driven by the InstallScript engine via a `setup.exe` file. This type of installation architecture can cause difficulties during deployment, such as:

- inability to customize or transform the application
- inability to conflict detect
- inability to suppress the user interface
- difficulty patching or upgrading the application

Also, if an InstallScript MSI installation is repackaged using traditional methods (OS Snapshot or Installation Monitoring), significant platform-specific or custom installation, maintenance, and uninstallation logic, and user interface information is lost because those methods only record the installation activities for the specific platform used during repackaging.

Therefore, it is recommended that you use InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package.

Select one of the following options:

- **Yes**—Use InstallScript Scan to automatically extract the MSI package and convert it to Basic MSI, while preserving the original components and installation logic. This is the default selection.
- **No**—Repackage the installation using the repackaging method selected on the **Method Selection Panel** (Installation Monitoring or Snapshot).

Click **Next** to proceed.

Set Target Project Information and Capture Settings Panel

The location where you want files created by Repackager stored is defined in the **Project path to store files** field on the **Set Target Project Information Panel**.

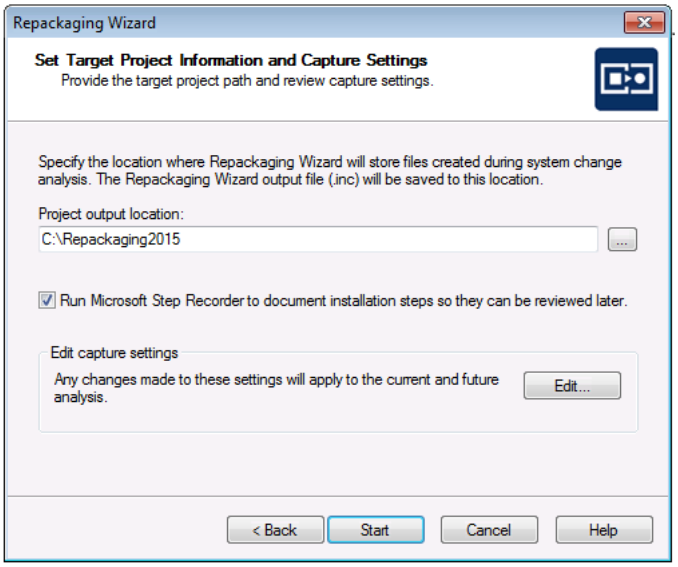


Figure 8-6: Repackaging Wizard Set Target Project Information and Capture Settings

It is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

You can also review or edit current capture settings by clicking **Edit**, which displays the **Analysis Options** dialog box. See Analysis Options Dialog Box for more information.

You can use the Microsoft Steps Recorder documentation tool with the Repackaging Wizard to automatically record the step-by-step actions that occur during repackaging. This information, which is saved in a web archive (`.mht`) file, includes a text description of where you clicked on each screen, along with a screen capture for each click. To enable this option, select the **Run Microsoft Step Recorder to document installation steps so they can be reviewed later** option. For more information, see Documenting Repackaging Steps Using the Microsoft Step Recorder Tool.

Click **Start** to begin repackaging and display the **Repackaging Panel**.

InstallScript MSI Conversion Output Panel

On this panel, specify the location where you want Repackager to store the files it creates during InstallScript Scan analysis. The converted Windows Installer MSI package will be saved to this location.

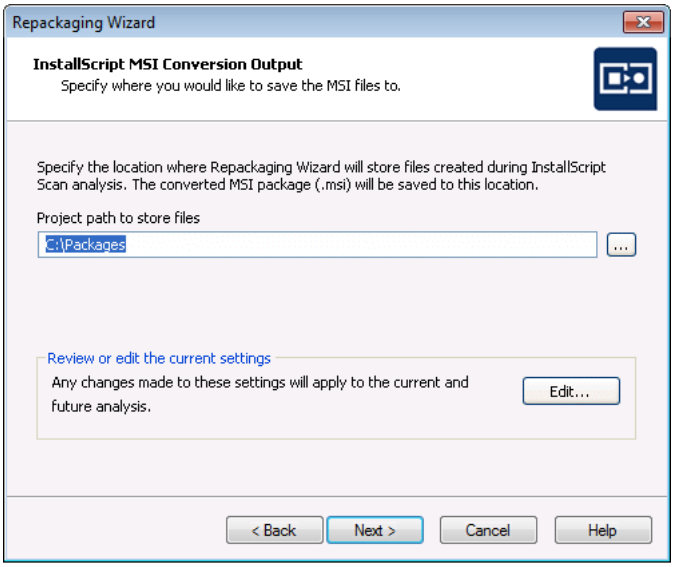


Figure 8-7: Repackaging Wizard InstallScript MSI Conversion Output Panel

It is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

You can also review or edit current settings by clicking **Edit** to open the **Analysis Options** dialog box. On the **Analysis Options** dialog box, you can specify capture types for the repackaging session, and, for snapshot-mode captures, you can restrict directory analysis to specific directories.

Click **Start** to begin repackaging and display the **Repackaging Panel**.

Repackaging Panel

The Repackaging panel appears while Repackager analyzes your system.

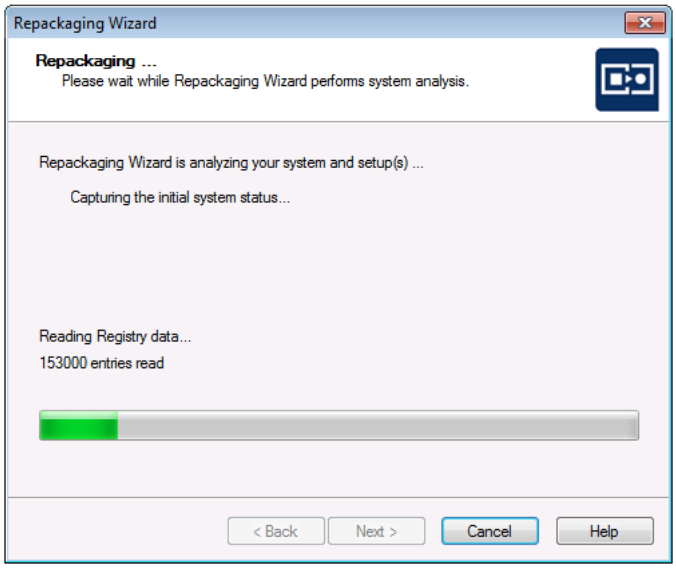


Figure 8-8: Repackaging Wizard Repackaging Panel 1

Depending on settings configured before starting repackaging, the analysis may stop following the initial phase, and again after setup has been run.

After the setups have been completed, you are prompted to click the **Next** button to complete the repackaging process.

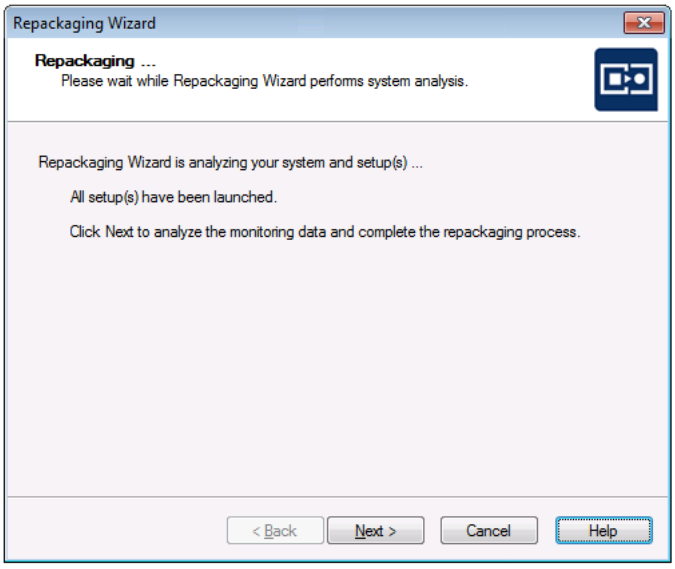


Figure 8-9: Repackaging Wizard Repackaging Panel 2

When you click **Next**, the repackaging is performed and its progress is displayed.

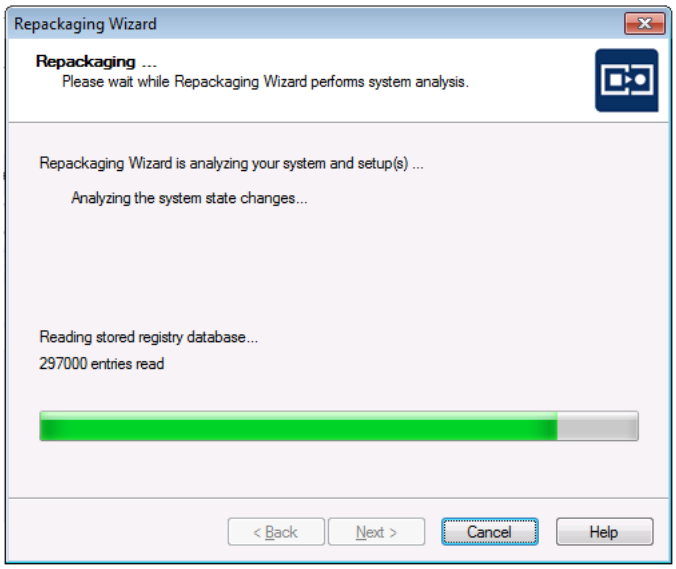


Figure 8-10: Repackaging Wizard Repackaging Panel 3

Following repackaging, the **Summary Panel** is displayed.

Summary Panel

The final panel displayed by Repackager is the Summary panel.

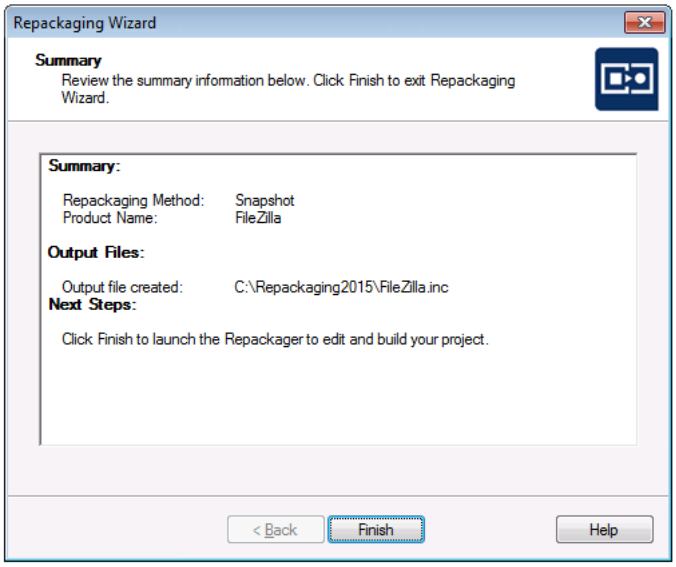


Figure 8-11: Repackaging Wizard Summary Panel

This panel provides confirmation that repackaging was successful, and provides the location of the source setup program(s), the Windows Installer package, and the InstallShield Editor project.

Additional Repackaging Wizard Dialog Boxes

The following dialog boxes can be accessed from the Repackaging Wizard:

- Additional Setup Programs Dialog Box
- Setup Information Dialog Box
- Excluded Processes Dialog Box
- Analysis Options Dialog Box

Additional Setup Programs Dialog Box

This dialog box, which is accessed by clicking the **Edit Setup List** button on the **Collect Product Information Panel** of the Repackaging Wizard, displays a list of additional setup programs you want to add to the final Windows Installer package.

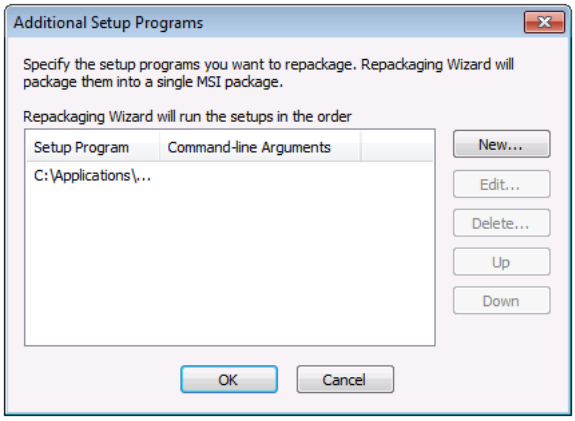


Figure 8-12: Repackaging Wizard's Additional Setup Programs Dialog Box

Essentially, this is a list of the other executables to run, in the order they are to be run, prior to final analysis. The following buttons are available:

Table 8-15 • Additional Setup Programs Dialog Box Buttons

Button	Description
New	Brings up the **Setup Information** dialog box to enter information about the setup programs.
Edit	Displays the **Setup Information** dialog box to edit information about the currently selected setup.
Delete	Removes the currently selected setup.
Up	Moves the selected setup up in the setup programs list.
Down	Moves the selected setup down in the setup programs list.

Setup Information Dialog Box

The **Setup Information** dialog box allows you to enter or edit information pertaining to the installations you are repackaging.

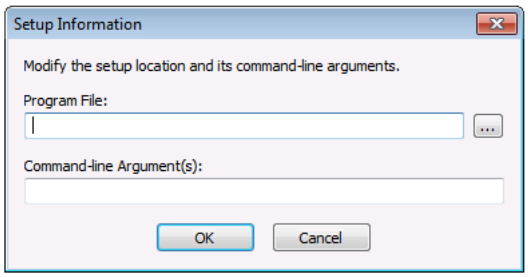


Figure 8-13: Setup Information Dialog Box

Accessible from the **Additional Setup Programs** dialog box, you can provide the name and location of an additional setup program, and any command-line arguments for the setup.

Excluded Processes Dialog Box

During Installation Monitoring, Repackager captures all of the activity of each service or process running on the machine, and then processes this collected data. However, many services running on a machine may have nothing to do with the installation being repackaged. Therefore, you may choose to exclude those processes by adding them to the list on the **Excluded Processes** dialog box.

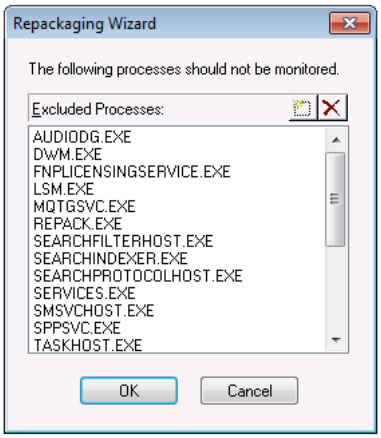


Figure 8-14: Repackaging Wizard Excluded Processes Dialog Box

You can open the **Excluded Processes** dialog box by clicking the **Advanced Settings** link on the Repackaging Wizard Method Selection Panel. The **Excluded Processes** dialog box initially lists a default set of processes.

- **To add a process to this list**, click the New ( ) button to add a new blank line to this list, and enter the name of the process that you want to exclude.
- **To delete a process from this list**, select the process and click the Delete ( ) button.

Analysis Options Dialog Box

The Analysis Options dialog box, accessible by clicking **Edit** from the **Set Target Project Information and Capture Settings Panel** or the **InstallScript MSI Conversion Output Panel**, allows you to specify capture types for the repackaging session. You can also edit or change the exclusions file that will be used for this repackaging session.

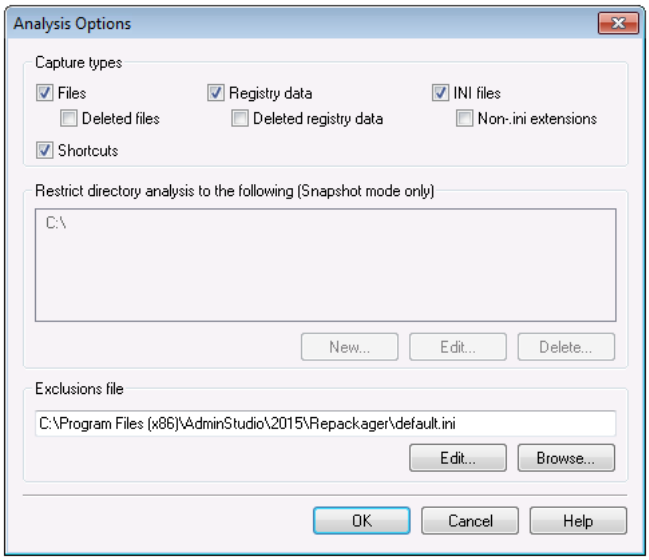


Figure 8-15: Analysis Options Dialog Box

Note • *Options set in this dialog box apply to the recurrent and subsequent repackaging sessions.*

Capture Types

To specify the type of files that will be captured during this repackaging session, make selections in the Capture types area:

Table 8-16 • Capture Types

Type	Description
Files	Capture file names during repackaging.
Deleted files	Capture deleted file names during repackaging. ***Note*** • *If you select this option, deleted files will be displayed on the Deleted Files View of the Repackager interface.*
Registry data	Capture registry data during repackaging.

Table 8-16 • Capture Types

Type	Description
Deleted registry data	Capture deleted registry data during repackaging. ***Note*** *• If you select this option, deleted registry entries will be displayed on the Deleted Registry Entries View of the Repackager interface.*
INI files	Capture .ini files during repackaging.
Non-ini extensions	Capture .ini files with non-.ini extensions during repackaging.
Shortcuts	Capture shortcuts during repackaging.

Restricting Directory Analysis to Specific Directories

For snapshot-mode captures, you can restrict directory analysis to specific directories, which can significantly improve repackaging performance. Selected directories are listed in the **Restrict directory analysis to the following** box.

- To add a directory restriction, click **New**.
- To modify an existing restriction, click **Edit**.
- To remove a restriction, click **Delete**.

Important *• By default, `C:\` is listed in the **Restrict directory analysis to the following** list. If you want to restrict directory analysis to specific directories on a machine, you must not only add the specific directories, but you must also delete `C:\` from this list.*

Editing or Changing Exclusions File

In the **Exclusions file** field, the path to the file that contains the default configuration for Repackager, including default exclusion information, is listed. By default, the location is:

```
[AdminStudioInstallDirectory]\Repackager\default.ini
```

- Click **Edit** to edit the listed exclusion file in the Exclusions Editor.
- Click **Browse** to browse to a different exclusion file.

Repackaging Wizard Command-Line Options

The following command-line options are supported by the Repackaging Wizard:

Table 8-17 • Repackaging Wizard Command-Line Options

Option	Description
`-?`	Displays a dialog box containing usage information for all Repackager command line options: • **If a option name is provided**, detailed help for the specified option will be displayed. • **If no option name is provided,** a dialog box containing general usage information for all options is displayed.
`-app <setup program list>`	Enables you to provide a pipe (|) delimited list of setups to run during repackaging. You can also pass command-line arguments to the setup by separating them from the setup name with a semicolon. If entering multiple setups, proper double quoting—including escaping nested quotes—is necessary: `-app "\"exe1path\";cmdline1|\"exe2path"|...exeN; cmdlineN`
`-cf <config.ini>`	This option allows you to select your own configuration template containing exclusions. A sample of this type of file (named `Default.ini`) can be found in the following directory: `[AdminStudioInstallDirectory]\Repackager` This particular file contains the default exclusion information.
`-cs <configuration type>`	This option allows you to select the configuration file type for exclusions. Possible values are: • **Shared**—Use shared settings from those stored in the `AdminStudio Shared` directory. • **Custom**—Use a custom configuration file (in conjunction with `-cf`).

Table 8-17 • Repackaging Wizard Command-Line Options (cont.)

Option	Description
`-is`	Regarding the Repackaging an InstallScript MSI Setup to a Basic MSI Setup procedure, use this parameter in the command line using the following syntax: `Repack.exe -app "c:\setup.exe" -o C:\apps\output -mm -is` In the above example, the user wants to repackage `c:\setup.exe` using the Installation Monitoring repackaging method (as specified by `-mm`) and InstallScript conversion (as specified by `-is`). Repackager would perform the InstallScript conversion process and produce a Basic MSI package with InstallScript support as output. Without the -is parameter, Repackager would perform repackaging without performing InstallScript conversion, and would only create a Repackager `.inc` file as the output. ***Note*** *• The command line parameter -is will be considered only if the setup to be repackaged is a InstallScript MSI setup. If user specified any other legacy setup that is not a InstallScript MSI setup then -is will be ignored.* ***Note*** *• If user chooses to use the Multiple Step Snapshot repackaging method, then the -is parameter will be ignored. Even if the setup is an InstallScript MSI setup, -is will still be ignored when using the Multiple Step Snapshot repackaging method.*
`-mode <snapshot mode>`	Repackager supports the following repackaging modes for snapshots: • **single**—Single step repackaging that creates an INC file as its output. • **pre**—Pre-scanning only scans the local drive for a baseline snapshot of the system. • **post**—Post-scanning only scans the local drive and compares the result with the pre-scan. The differences are written to the INC file as output.
`-mm`	Instructs Repackager to use installation monitoring as the repackaging technology.
`-ms`	Instructs Repackager to use snapshots as the repackaging technology.
`-o <inc path name>`	Specifies a folder path not including the filename. The file name is derived from the Product Name unless overridden with the -of switch.
`-of <inc file name>`	Specifies the .inc file name that should be used instead of the product name. Use -o to specify the path.
`-onp`	When using the **Installation Monitoring** method via command line to perform repackaging on a 64-bit operating system, you can use the `-onp` command line option to cause the **Installation Monitoring** method to only monitor new processes created on the system and to ignore any existing/ running ones. This option is useful to optimize the monitoring process on a 64-bit operating system.

Table 8-17 • Repackaging Wizard Command-Line Options (cont.)

Option	Description
`-pc <company name>`	Allows you to set the company name.
`-pp <product name>`	Allows you to set the product name. This will be the same name as the generated Repackager output file (`.inc`).
`-pv <product version>`	Allows you to set the product version.
`-sb`	This option allows you to run Repackager silently, with no user interaction. A progress dialog box is displayed. If no .ini file is specified using the -i parameter, Repackager uses `Repack.ini` as the default input file. If an output folder is not specified using -o, the default output folder is `C:\Packages`.
`-sn`	This option allows you to run Repackager silently, with no user interaction and no progress dialog box. If no .ini file is specified using the -i parameter, Repackager uses `Repack.ini` as the default input file. If an output folder is not specified using -o, the default output folder is `C:\Packages`.
`-version`	This option displays standard version information for Repackager, including the full version and copyright information.

Tip *• To open a help topic from the command line that lists command line options, enter the following:*

```
repack.exe /?
```

Note *• In addition to the - sign for command-line arguments, you can also use the / symbol.*

Reboot Handling in the Repackaging Wizard

During repackaging, a setup may require a reboot. For example, some operations may require a file which is in use be replaced, which can only be done after a reboot. Some nuances exist depending on the repackaging technology you are using (Snapshot or Installation Monitoring). In either case, when the Repackaging Wizard detects that a reboot is necessary, the Repackaging Wizard saves the appropriate data and waits until you confirm that you are ready to reboot the machine.

For Snapshot repackaging, the operating system completes the reboot operation. During startup, the operating system restarts all applications and processes and performs any pending file operations. One of the applications that restarts is Repackager. Before you continue processing in Repackager, be patient and ensure all processes and applications have restarted. This may take a minute or two. After the applications and processes have been launched, you can continue repackaging by clicking **Next**.

For Installation Monitoring, on reboot the operating system launches the Repackaging Wizard, which in turn launches applications and processes and waits until these are finished before prompting you to continue repackaging. However, in some cases the processes or applications launched by the Repackaging Wizard will launch other applications and processes. As in Snapshot repackaging, it is generally a good idea to wait a minute or two before clicking **Next**.

In both circumstances, waiting helps ensure the setup is fully installed and that captured data contains the necessary information to properly rebuild the setup as an MSI installation.

Note • *On Windows Vista and newer, system reboots are almost instantaneous and do not allow running applications to properly shut down, which may result in a loss of data. When using the* ***Installation Monitoring*** *method, Repackager successfully handles a system reboot and delays it until you click the* ***Reboot*** *button on the Repackaging Wizard.*

9

Converting Legacy Installations Using the Repackager Interface

A Repackager project file (.irp) can be built into an InstallShield Editor project (.ism) or a Windows Installer package (.msi). You can use the Repackager interface to create and modify Repackager project files. You can also use it to build an isolated Windows Installer package and to configure the exclusions used when repackaging a legacy installation.

Information about the Repackager interface is presented in the following sections:

Table 9-1 • Using the Repackager Interface

Section	Description
About the Repackager Interface	Explains how to launch the Repackager interface and how to set options.
Creating Repackager Projects	Explains how to create a Repackager project file (.irp), which can then be built into an InstallShield Editor project (.ism) or a Windows Installer package (.msi).
Working With Repackager Projects	Explains how to build an InstallShield Editor project and Windows Installer package from a Repackager project. The topics in this section include: • Building an InstallShield Editor Project • Building a Windows Installer Package • Automatically Generating a Virtual Application During Repackager Project Build • Viewing Repackager Project Properties • Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project • Creating a Setup Capture Report for a Project • Generating Software ID Tag Files During Repackaging • Saving Repackager Projects • Opening InstallShield Editor from Repackager

Table 9-1 • Using the Repackager Interface (cont.)

Section	Description
Isolating Windows Installer Packages	Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. This section reviews isolation concepts and options, and explains how to build an isolated Windows Installer package.
Configuring Exclusions	Explains how to use Repackager and the Exclusions Editor to configure the exclusions used when repackaging a legacy installation.
Creating an InstallShield Editor Template to Use Within Repackager	Explains how to create an InstallShield Editor template to use to speed up the Repackaging process.
Repackager Interface Reference	Describes each of the views and dialog boxes that you might encounter when using the Repackager interface. The help topics in this section are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

Note • *For information on other Repackager features, see Repackaging Legacy Installations Using the Repackaging Wizard.*

About the Repackager Interface

Information about using the Repackager interface is presented in this section:

- Launching the Repackager Interface
- Setting Repackager Options

Launching the Repackager Interface

Repackager can be launched from within the AdminStudio interface. Additionally, if you install Repackager on a network, use Windows Explorer to browse to the `islc.exe` executable on the shared drive.

Task ***To launch Repackager from the AdminStudio interface:***

1. Launch AdminStudio.
2. Click the **AdminStudio Tools** button.
3. From the Tools Gallery, click the **Repackager** icon on the left side.

The Repackager Start Page opens and you can begin the repackaging process.

Note • *You can also launch Repackager directly from the Windows **Start** menu by pointing to **All Programs**, AdminStudio, **AdminStudio 2022 R2 | 24.0 Tools**, and clicking **Repackager**.*

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

Setting Repackager Options

On the Options Dialog Box, which is opened by selecting **Options** from the **Tools** menu, you can specify the following Repackager options:

- Selecting Data Display Colors
- Specifying Additional Merge Module Directories
- Controlling the Display of ICE Validation Warnings
- Suppressing Build Output Folder Overwrite Warnings

Selecting Data Display Colors

On the **Colors** tab of the Repackager Options dialog box, you can configure the color of scanned items and deleted items in Repackager's exclusion views (Files, `.ini` Files, Registry Data, and Shortcuts).

Task ***To change the way excluded and included data is displayed in Repackager:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Options**. The **Colors** tab of the **Options** dialog box opens.
3. Configure the display colors for **Excluded** and **Setup Intent** items.
4. Click OK.

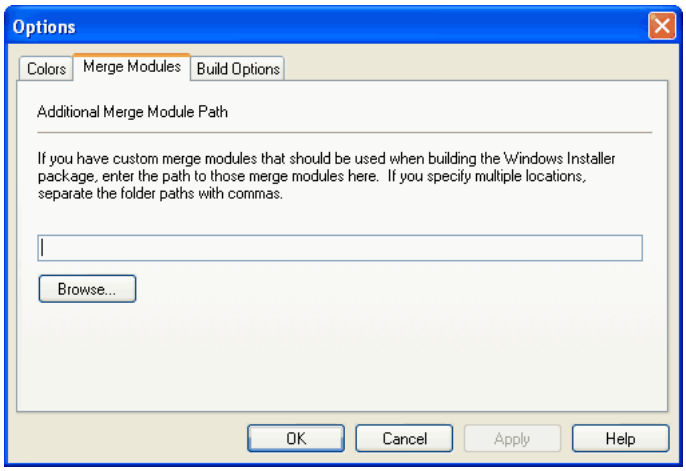
Specifying Additional Merge Module Directories

If you have custom merge modules that should be used when building a Windows Installer package, you need to specify the directories that contain those custom merge modules on the **Merge Modules** tab of the Options dialog box.

Task ***To specify directories of additional Merge Modules:***

1. Open the Repackager interface.
2. From the Tools menu, select **Options**. The **Colors** tab of the Options dialog box opens.

3. Open the **Merge Modules** tab.



4. Enter the directory paths to the custom merge modules. To specify multiple directories, separate the folder paths with commas.

Note • *You can click **Browse** and navigate to a directory, but if you browse to a second directory, its directory path will replace the one you initially selected. Therefore, if you want to specify multiple directories separated by commas, you need to manually enter the directory paths.*

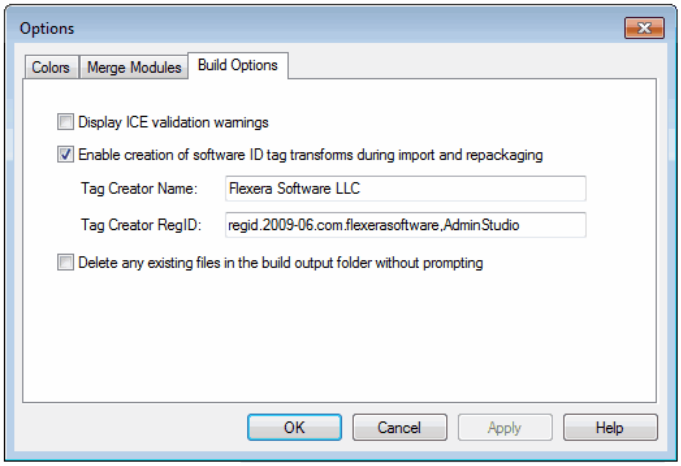
5. Click **OK**.

Controlling the Display of ICE Validation Warnings

On the **Build Options** tab of the Options Dialog Box, you can specify whether or not you want to list ICE validation warnings in the Repackager output window during the Build process.

Task ***To set the display of ICE validation warnings during builds:***

1. From the Repackager interface, select **Options** from the **Tools** menu. The Options dialog box opens.
2. Open the **Build Options** tab.



3. To display any ICE validation warnings that occur during the Repackager Build process, select the **Display ICE validation warnings** option. By default, this option is not selected.

Note • *For information on the software ID tag options on the **Build Options** tab of the **Options** dialog box, see Enabling Software ID Tag Generation During Repackaging.*

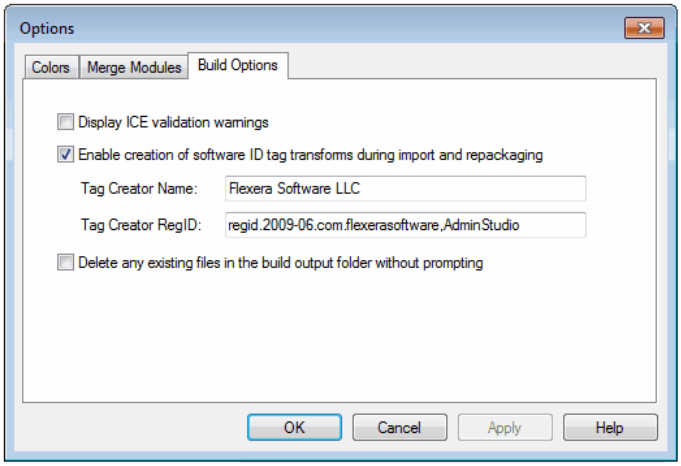
Suppressing Build Output Folder Overwrite Warnings

By default, Repackager will build the Repackager project's associated Windows Installer package in a directory named `MSI_Package`, which is a subdirectory of the directory containing the Repackager project. If you have edited the Repackager project's associated InstallShield Editor project to change this default location, each time you rebuild the Repackager project, Repackager will prompt you to confirm that you want to overwrite the existing files.

If you are repeatedly building from the Repackager interface into the same build output folder (which is not the default `MSI_Package` folder) and you do not want to be prompted to confirm that you will be overwriting the existing content, you can select an option on the Repackager **Options** dialog box to suppress the confirmation prompts.

Task ***To suppress build output folder overwrite warnings:***

1. On the **Tools** menu, click **Options**. The Repackager **Options** dialog box opens.
2. Open the **Build Options** tab.



3. Select the **Delete any existing files in the build output folder without prompting** option.
4. Click **OK**.

Creating Repackager Projects

Repackager projects (`.irp`) allow you to visually analyze the files, `.ini` files, shortcuts, and registry entries captured or changed during the conversion of a legacy setup into a Windows Installer package. You can also exclude files, shortcuts, registry entries, and `.ini` files from the resulting Windows Installer package, without affecting the original setup data.

There are two methods of creating Repackager projects:

Table 9-2 • Methods of Creating Repackager Projects

Method	Installation Source
Repackaging Wizard	You can use the Repackaging Wizard to convert the following installations: ● InstallShield Professional 1.x to 5.1.x ● InstallShield Professional 5.5 to 7.x ● InstallShield InstallScript MSI ● InstallShield DevStudio 9.x InstallScript ● InstallShield Editor InstallScript See Repackaging Legacy Installations Using the Repackaging Wizard.

Table 9-2 • Methods of Creating Repackager Projects (cont.)

Method	Installation Source
Repackager Interface	You can use the Repackager interface to convert the following installations: • Repackager 3.x output (.inc) • Microsoft SMS projects (.ipf) • Microfocus ZENworks projects (.axt/.aot) • WinINSTALL projects (.txt) (6.0, 6.5, 7.x) • Wise installation projects (.wse) • InstallShield Professional log files (.isl) See Converting Legacy Installations Using the Repackager Interface.

Converting Legacy Installations Using the Repackager Interface

In addition to repackaging a legacy installation using the Repackaging Wizard, you can also convert many setup types directly to Repackager projects (`.irp`)—and ultimately to InstallShield Editor projects (`.ism`) and Windows Installer packages (`.msi`). Repackager can directly convert the following setup types:

- Converting Repackager 3.x Output Files
- Converting a Microsoft SMS Project to a Repackager Project
- Converting Microfocus ZENworks Projects
- Converting WinINSTALL Projects
- Converting Wise Installation Projects
- Converting InstallShield Professional Log Files

Converting Repackager 3.x Output Files

To convert a Repackager 3.x output file to a Repackager project, perform the following steps.

Task ***To convert a Repackager 3.x output file (.inc) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. On the **File** menu, click **Open**. The **Open** dialog box opens.
3. Change the **Files of type** filter to **Legacy Repackager Files (*.inc)**.
4. Browse to locate the Repackager 3.x output file you want to convert.
5. Select the file and click **OK**.

The Repackager 3.x project is updated to the Repackager project (`.irp`) format. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting a Microsoft SMS Project to a Repackager Project

To convert a Microsoft SMS project to a Repackager project, perform the following steps.

Task ***To convert a Microsoft SMS project (.ipf) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to SMS Installer (*.ipf).
4. Browse to locate the SMS project you want to convert.
5. Select the project, and click OK.

The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting Microfocus ZENworks Projects

You can convert Microfocus ZENworks projects (`.axt/.aot`) to Windows Installer packages (`.msi`) one at a time or in bulk:

- **Repackager Interface**—You can convert a ZENworks project to a Windows Installer package using the Repackager interface. See Converting a Microfocus ZENworks Project Using the Repackager Interface.
- **Command Line**—You can use the Command Line to bulk convert multiple ZENworks projects to Windows Installer packages. See Converting Multiple Microfocus ZENworks Projects Using the Command Line.

Note • *In order to convert an* ***.aot*** *file, the ZENworks Desktop Management Agent 6.5 or later (zenlite.dll) must be installed on the workstation where Repackager is installed. If this agent is not installed, Repackager can only convert ZENworks .axt files. See About .axt and .aot Application Object Template Files for more information.*

Converting a Microfocus ZENworks Project Using the Repackager Interface

Using Repackager, you can convert Microfocus ZENworks projects (`.axt/.aot`) to Windows Installer packages (`.msi`).

About .axt and .aot Application Object Template Files

In ZENworks Desktop Management, the snAppShot utility generates application object template files—with either an `.axt` or `.aot` extension—that contain the details that are required for the Application Launcher to be able to distribute an application to a workstation:

- registry entries to be added
- files to be copied

- changes to be made in the `.ini` files and system text files (`autoexec.bat` and `config.sys`)

Because an `.axt` file is a text file that can be edited with a text editor in order to modify it after it has been created, it can be opened and converted by Repackager.

However, in order to convert an `.aot` file (which is not a text file), the ZENworks Desktop Management Agent 6.5 or later (`zenlite.dll`) must be installed on the workstation where Repackager is installed. If this agent is not installed, Repackager can only convert ZENworks .axt files.

Note • *For information on installing the ZENworks Desktop Management agent (version 6.5 or later) to a workstation, see Novell ZENworks 6.5 Desktop Management Installation Guide.*

Task ***To convert a Microfocus ZENworks project (.axt/.aot) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. On the **File** menu, click **Open**.
3. In the **Open** dialog, change the **Files of type** filter to **Microfocus ZENworks (*.axt)** or **Microfocus ZENworks (*.axt/*.aot)**.

   Note • *If the ZENworks Desktop Management Agent 6.5 or later (zenlite.dll) is installed on the workstation where Repackager is installed, the Files of type filter will be Microfocus ZENworks (*.axt/*.aot). If this agent is not installed, the Files of type filter will be Microfocus ZENworks (*.axt) and you will be unable to select .aot files as the legacy setup source. See About .axt and .aot Application Object Template Files for more information.*
4. Browse to locate the ZENworks project you want to convert.
5. Select the project, and click **OK**.

The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting Multiple Microfocus ZENworks Projects Using the Command Line

To perform a bulk conversion of ZENworks projects to Windows Installer packages, you use the `-Z` command line switch.

Task ***To convert multiple Microfocus ZENworks projects (.axt/.aot) to a Windows Installer package, a Repackager project, or an InstallShield Editor project:***

1. Create an `.ini` file using the following format:

```
[General]
OutputFormat=MSI|INC|ISM

[AXT]
C:\myData\Project1.axt
C:\myData\Project2.axt
```

```
C:\myData\Project3.axt

[AOT]
C:\myData\Project1.aot
C:\myData\Project2.aot
C:\myData\Project3.aot
C:\myData\Project4.aot
```

The following table describes the elements of this file:

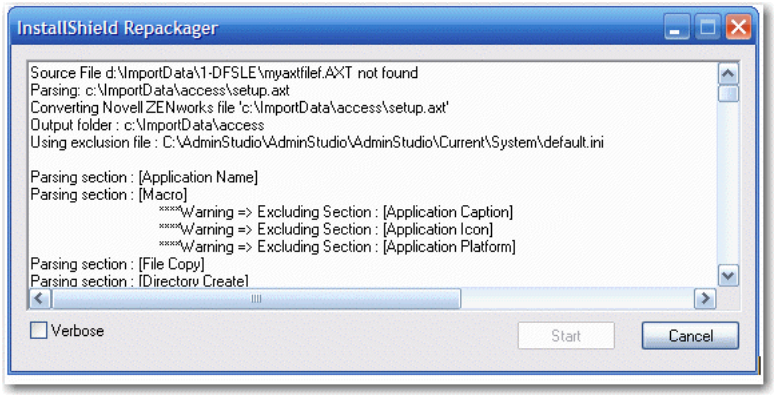
Section	Description
[General]	Controls the output format of the entire conversion process. Select one of the following to identify the output format: • **MSI**—Windows Installer package • **INC**—Repackager output file • **ISM**—InstallShield Editor project file
[AXT]	List the names and locations of the legacy ZENworks projects (`.axt`) you want to convert. Include the paths (absolute or relative) to the `.axt` files.
[AOT]	List the names and locations of the ZENworks `.aot` projects you want to convert. Include the paths (absolute or relative) to the `.aot` files.

2. Run the repackaging process from the command line using the `-Z` parameter:

```
ISLC.exe -Z"C:\DirectoryName\FileName.ini"
```

Caution • *You must enter a fully qualified path to identify the location of your .ini file.*

Repackager loads the `.ini` file and begins the conversion process. A dialog box opens to display progress messages.



To limit the volume of messages listed, clear the **Verbose** check box.

3. When the repackaging process is complete, the **Cancel** button changes to a **Close** button. Click **Close** to close this dialog box.

You will find the converted files in the location specified in the `.ini` file as the location of the `.aot`/`.axt` input files.

Converting WinINSTALL Projects

Task ***To convert a WinINSTALL 6.0, 6.5, or 7.x project (.txt) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the **File** menu, select **Open**.
3. In the **Open** dialog box, change the **Files of type** filter to **WinINSTALL (*.txt)**.
4. Browse to locate the WinINSTALL project you want to convert.
5. Select the project, and click **OK**.
6. If the **WinINSTALL Conversion** dialog box opens, fill in the WinINSTALL-specific variables and click **OK**.

   The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Note • *WinINSTALL projects must be converted to .txt files prior to conversion to Repackager projects.*

Converting Wise Installation Projects

Task ***To convert a Wise Installation project (.wse) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to Wise Projects (*.wse).
4. Browse to locate the Wise Installer project you want to convert.
5. Select the project, and click OK.

   The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting InstallShield Professional Log Files

You can convert an InstallShield Professional log file (`.isl`) to a Repackager project if you have access to the original setup media. When you open the log file, following the steps below, Repackager will try to find the original setup media automatically (in the location specified in the log file), but if it cannot, it will allow you to browse to it before continuing. If you do not have access to the original setup media, the conversion will fail.

Task ***To convert an InstallShield Professional Log File (.isl) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to InstallShield Pro Log Files (*.isl).
4. Browse to locate the InstallShield Professional log file you want to convert.
5. Select the file, and click OK.

   The log file is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Working With Repackager Projects

After creating a Repackager project—by Repackaging Legacy Installations Using the Repackaging Wizard or by Converting Legacy Installations Using the Repackager Interface—you can perform the following tasks:

- Building an InstallShield Editor Project
- Building a Windows Installer Package
- Automatically Generating a Virtual Application During Repackager Project Build
- Viewing Repackager Project Properties
- Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project
- Creating a Setup Capture Report for a Project
- Generating Software ID Tag Files During Repackaging
- Saving Repackager Projects
- Opening InstallShield Editor from Repackager

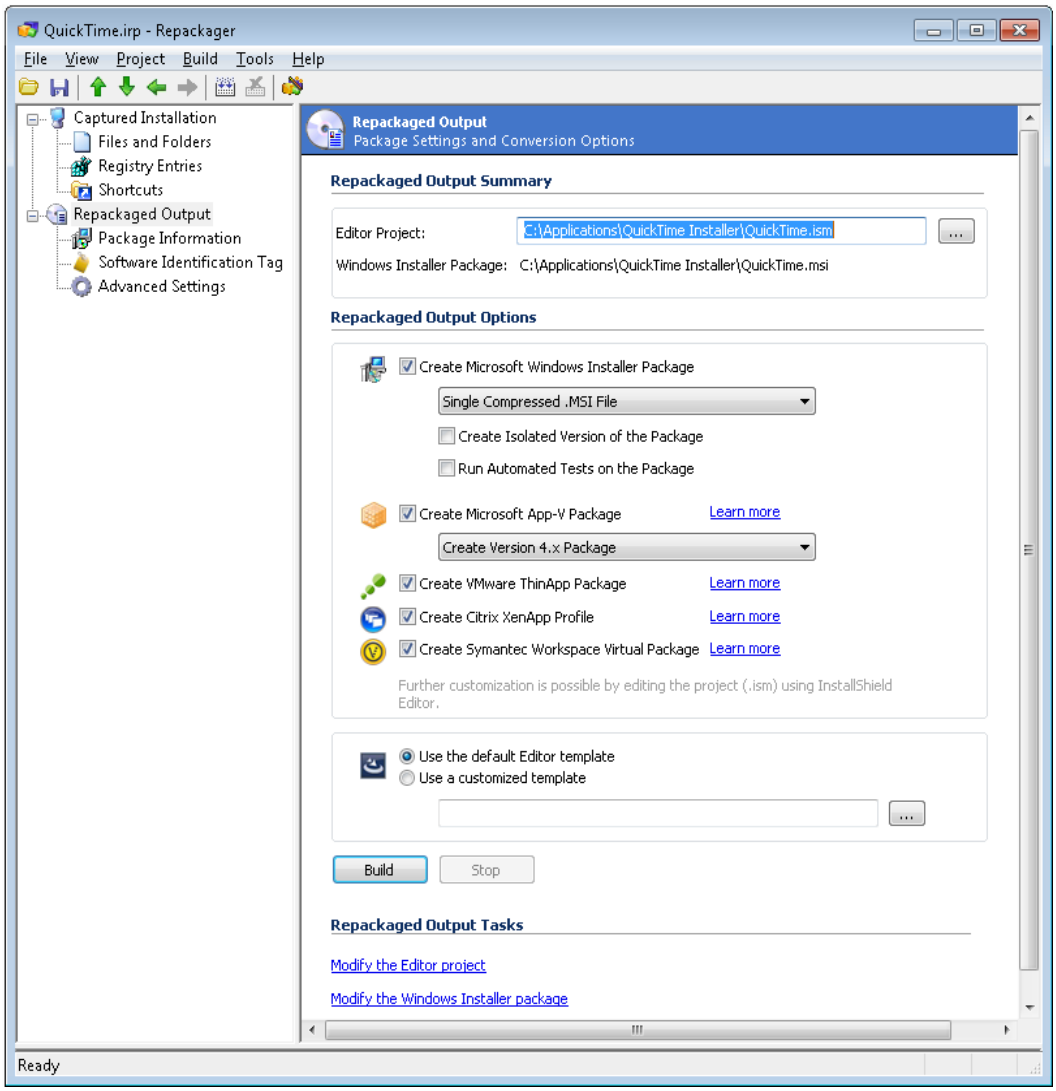
Building an InstallShield Editor Project

You can build an InstallShield Editor project (`.ism`) from your Repackager project (`.irp`).

You can also choose to build just an InstallShield Editor project, so that you can open it in InstallShield Editor and make some modifications prior to building.

Task ***To build an InstallShield Editor project (.ism):***

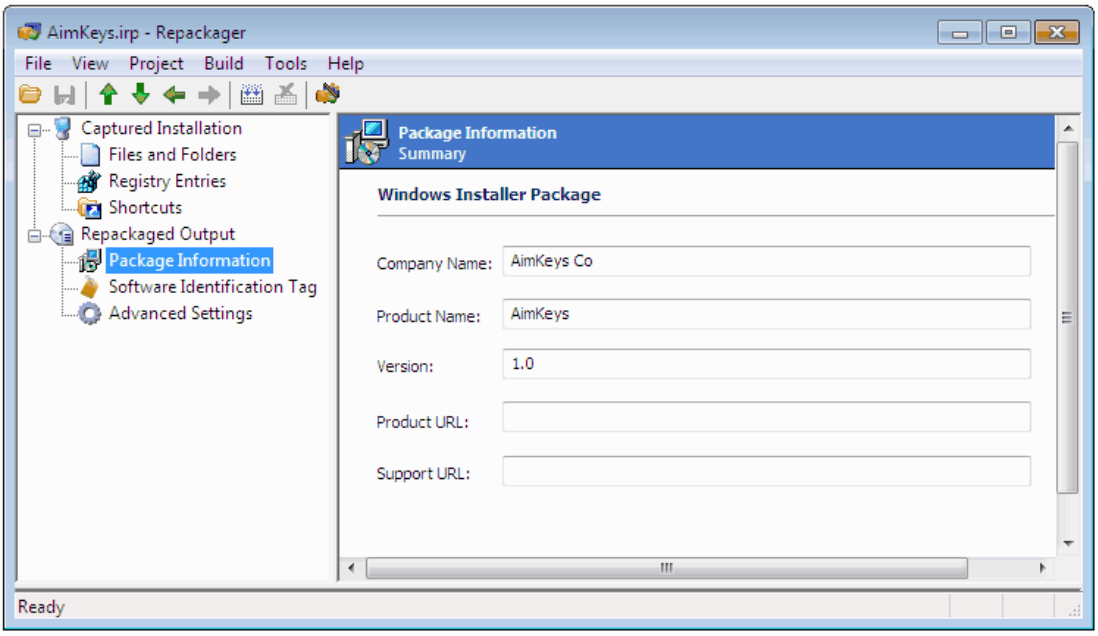
1. In the Repackager interface, open the Repackager project that you want to convert to an InstallShield Editor project.
2. Select **Repackaged Output** from the View List. The **Repackaged Output** view opens.



3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. If you do not want to **Create Microsoft Windows Installer Package**, clear this option. If you want to create a Windows Installer Package, see Building a Windows Installer Package.
5. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project. In the **Repackaged Output Options** area, select the InstallShield Editor Project Template you want to use when creating the project:
   - **Use the default Editor template**—Select this option to use the default InstallShield Editor Project Template.
   - **Use a customized template**—Select this option to use a customized InstallShield Editor Project Template.

     For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.

6. Select **Package Information** from the View List. The **Package Information** view opens, where you can specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.



7. Enter the following information:

   a. **Company Name**—The name of the company that developed the product you are repackaging.

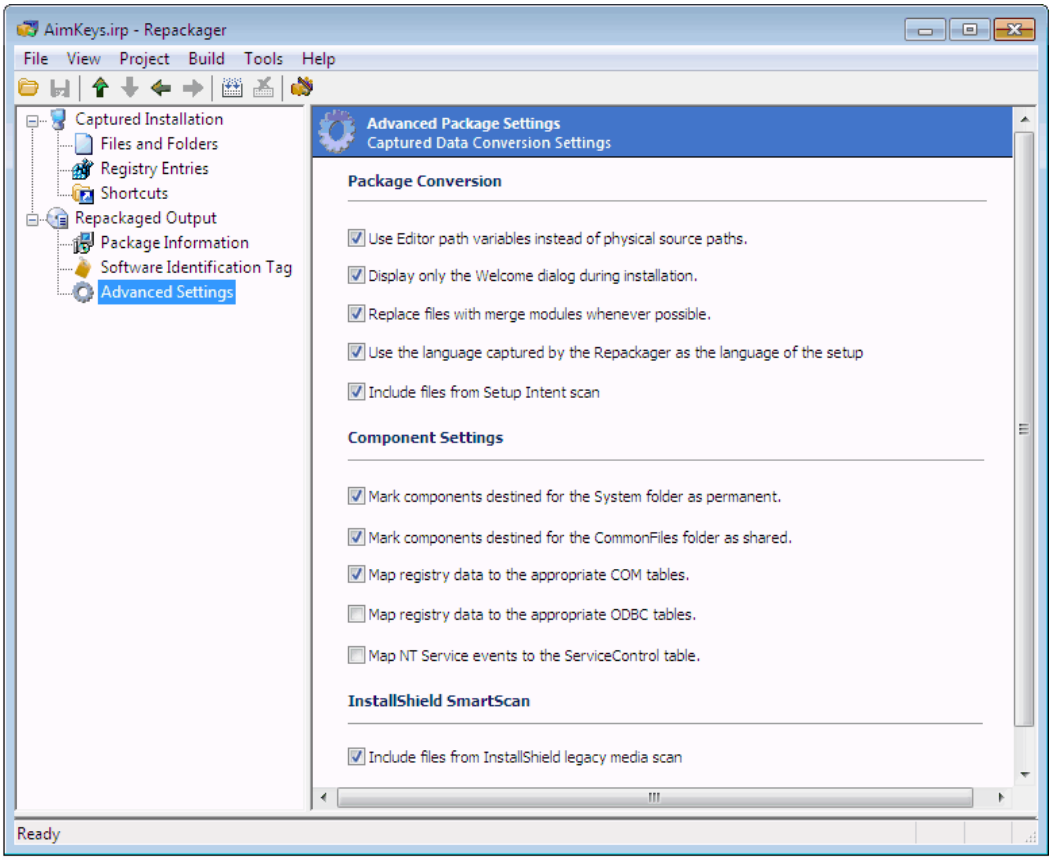
   b. **Product Name**—The name of the product you are repackaging.

   c. **Version**—The product's version number.

   d. **Product URL**—The URL for product information. This appears in **Add/Remove Programs** in the Control Panel.

   e. **Support URL**—A URL for support information. This also appears in **Add/Remove Programs** in the Control Panel, and is often changed during repackaging to provide an internal support URL.

8. Select **Advanced Settings** from the View List. The **Advanced Package Settings** view opens.



9. Select the options that you want to use, as described in Configuring Advanced Conversion Options.
10. Select **Repackaged Output** on the View List. The **Repackaged Output** view opens.
11. Click the **Build** button. The build process begins, and its progress is reported in the output window.

    When the build process is complete, a `Conversion completed` message appears in the output window, and a link to the build log file is provided.

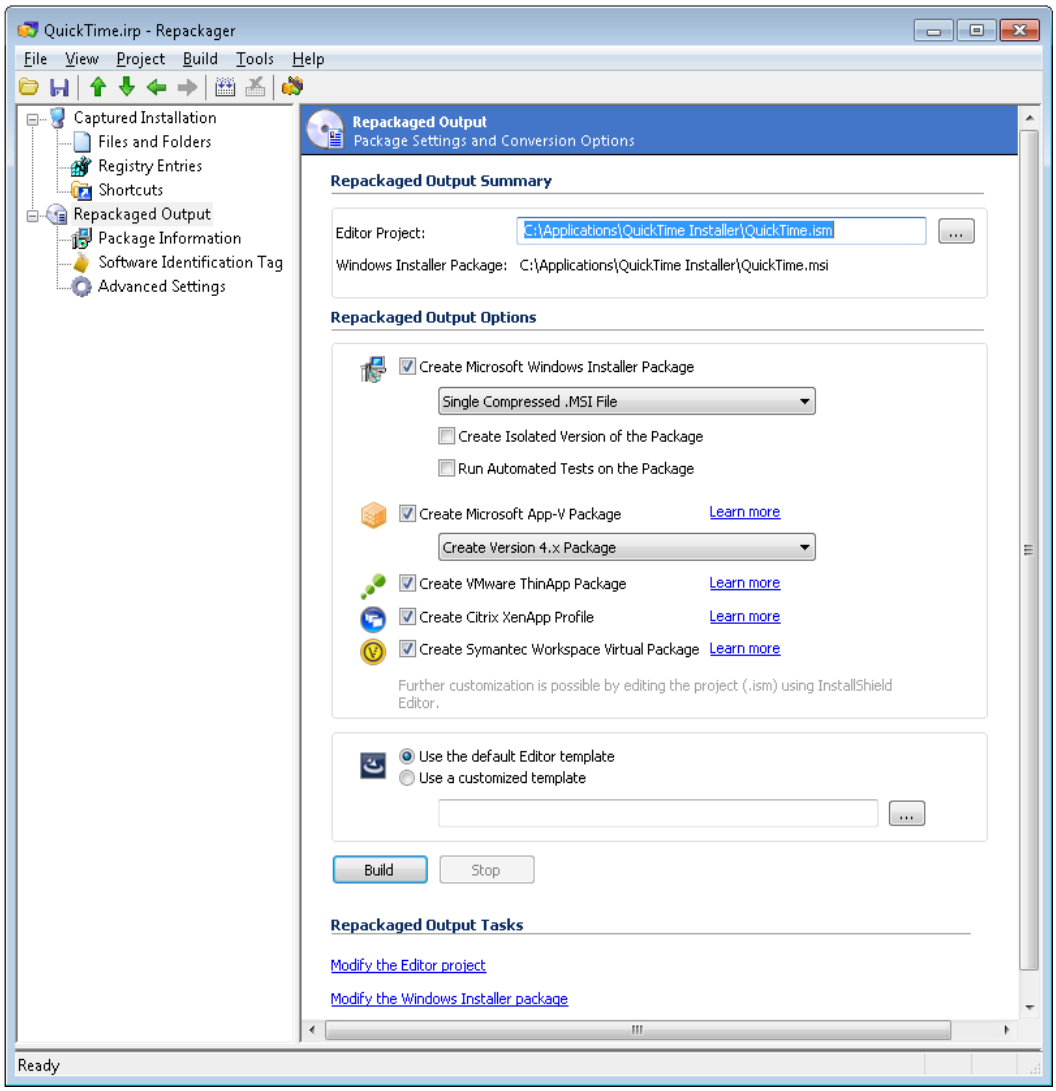
Building a Windows Installer Package

You can simultaneously build an InstallShield Editor project (`.ism`) and a Windows Installer package (`.msi`) from your Repackager project (`.irp`). However, before you do so, you must configure options in your Repackager project necessary for the build.

Note • *For information on building a virtual application, see Automatically Generating a Virtual Application During Repackager Project Build.*

Task ***To build an InstallShield Editor project (.ism) and a Windows Installer package (.msi):***

1. In the Repackager interface, open the Repackager project that you want to convert to an InstallShield Editor project and build a Windows Installer package.
2. Select **Repackaged Output** from the View List. The **Repackaged Output** view opens.



3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. Select the **Create Microsoft Windows Installer Package** option, and select the following additional options:
   a. The compression option that you select for this package depends upon the size of your application's installation and your delivery method.

      Neither `Setup.exe` nor your `.msi` file can be spanned across multiple disks. So, if the source files associated with your Windows Installer package cannot fit on the same disk as the `setup.exe` and `.msi` file, you will need to include them in `.cab` files on other disks. But if you are performing a network installation and have unlimited space, there is no need to compress files or include additional files in .cab files.

From the list, select one of the following options:

Option	Description
Single Compressed .MSI File	Select this option if you want to compress all necessary files inside the .msi package, as opposed to storing them outside of the .msi database.
Single Compressed Setup.exe File	Select this option if you want to compress all files inside a setup.exe file, including the .msi file and all other necessary files.
.MSI File With External .CAB File	Select this option if you want to create an .msi file and want to compress the rest of the necessary files in an external .cab file. For example, you might have an installation that contains three features—each containing a 1.5 MB file, Setup.exe, and the installation files for Windows NT—and you want to create a custom media type that is 2 MB in size. The build will span multiple disks. ● Disk one will contain Setup.exe, InstMsiW.exe (which contains the logic to install the Windows Installer service on Windows NT machines), Setup.ini (which is required for installations that include Setup.exe), and your .msi file. ● The remaining disks will contain .cab files that store compressed copies of all your source files.
.MSI File With External .CAB File and Setup.exe	Select this option if you want to create an .msi file and a setup.exe file, and want to compress all the rest of the necessary files in an external .cab file.
Uncompressed .MSI File	Select this option if you want to create an uncompressed .msi file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi file.
Uncompressed .MSI File With Setup.exe	Select this option if you want to create an uncompressed .msi file along with a setup.exe file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi and setup.exe files.

b. To reduce versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested, select the **Create Isolated Version of the Package**. An additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of:

appname`.isolated.msi`

For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages.

c. Select the **Run Automated Tests on the Package** option to automatically run best practice tests against the newly built Windows Installer package to determine if it is built according to Windows Installer standards, and if it is in compliance with the installation requirements of the Windows operating system.

d. To build a virtual application, select the **Create Microsoft App-V Package**, **Create VMware ThinApp Package**, **Create Citrix XenApp Profile**, and/or **Create a Symantec virtual application** option. See Automatically Generating a Virtual Application During Repackager Project Build.

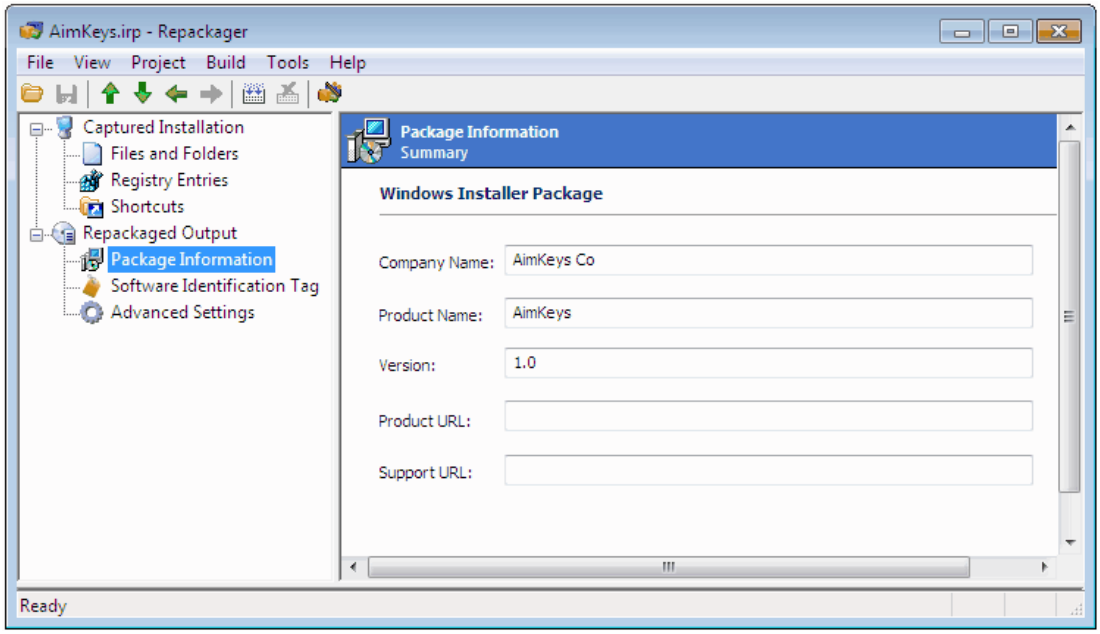
Note • *In order to select one of these virtualization options, you must have already selected the* ***Create Microsoft Windows Installer Package*** *option.*

5. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project. In the **Repackaged Output Options** area, select the InstallShield Editor Project Template you want to use when creating the project:

   - **Use the default Editor template**—Select this option to use the default InstallShield Editor Project Template.
   - **Use a customized template**—Select this option to use a customized InstallShield Editor Project Template.

     For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.

6. Select **Package Information** from the **View List**. The **Package Information** view opens, where you can specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.



7. Enter the following information:

   a. **Company Name**—The name of the company that developed the product you are repackaging.

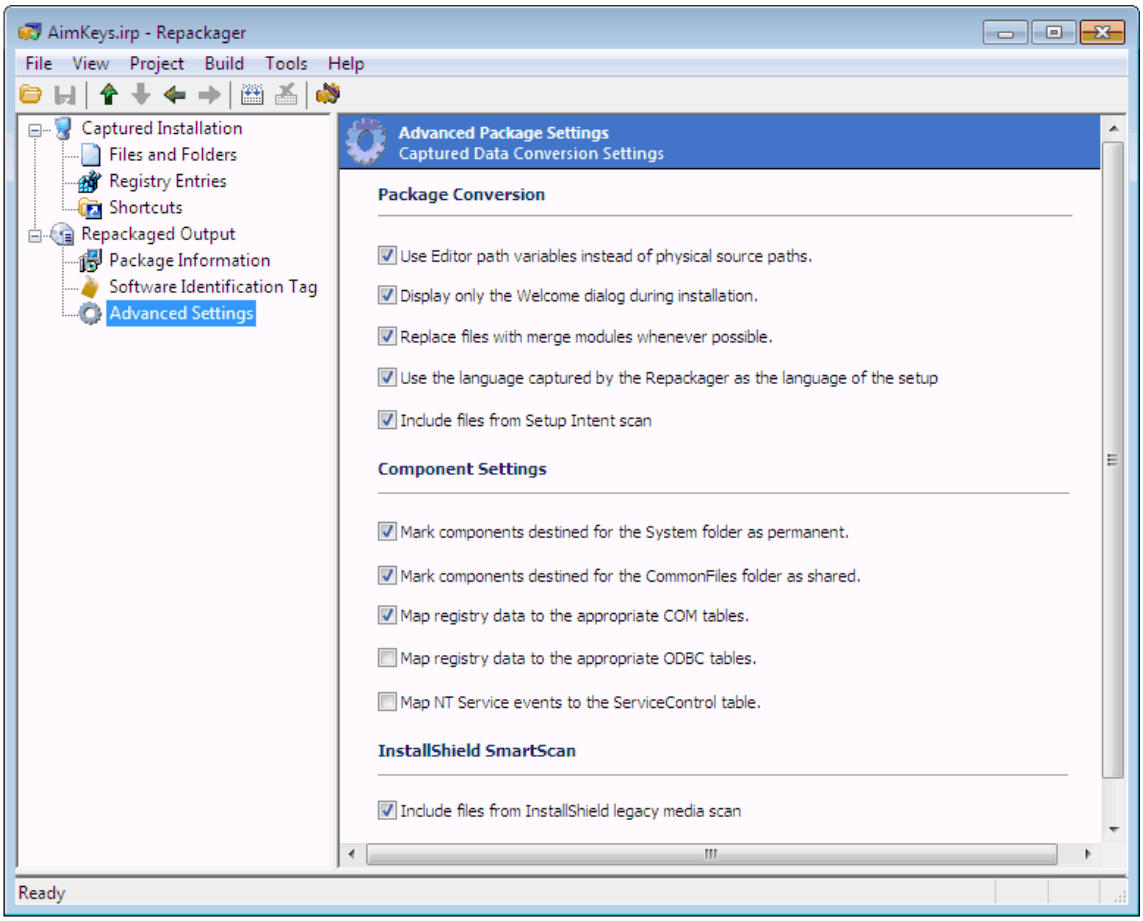
   b. **Product Name**—The name of the product you are repackaging.

   c. **Version**—The product's version number.

   d. **Product URL**—The URL for product information. This appears in **Add/Remove Programs** in the Control Panel.

   e. **Support URL**—A URL for support information. This also appears in **Add/Remove Programs** in the Control Panel, and is often changed during repackaging to provide an internal support URL.

8. Select **Advanced Settings** from the View List. The **Advanced Package Settings** view opens.



9. Select the options that you want to use, as described in Configuring Advanced Conversion Options.
10. Select **Repackaged Output** on the View List. The **Repackaged Output** view opens.
11. Click the **Build** button. The build process begins, and its progress is reported in the output window.

    When the build process is complete, a `Conversion completed` message appears in the output window, a link to the build log file is provided, and the location of the `.msi` file is listed. For example:

    ```
    Output file: C:\1516261\WinZip.msi
    ```

About the Context.msi File

When some Windows Installer packages are repackaged, some of their data (such as files or registry entries) are excluded according to the normal Repackager exclusion settings. For example, files destined for the `\Windows\Installer` folder are typically excluded. However, this type of information is occasionally necessary in order to successfully convert a Windows Installer package to a virtual package.

To address this issue, when Repackager builds a Windows Installer package, it now produces two `.msi` files: `packagename.msi` and `packagename.context.msi`.

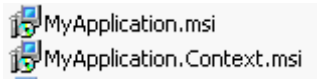


Figure 9-1: Repackaged Output: application.msi and application.context.msi

The `.context.msi` file contains context data that is necessary in order to convert a `.msi` file to a virtual package. When creating a virtual package, Repackager combines the data in both the main `.msi` file and the `.context.msi` file to produce the final virtual package.

Note • *For more information on the `.context.msi` file, see Capturing Virtualization Context in the AdminStudio Help Library.*

Important • *If you are not converting a package to a virtual package, you can ignore its `.context.msi` file.*

Note • *Context data is not displayed in the Repackager interface when viewing captured Files/Registry details.*

Configuring Advanced Conversion Options

To set package conversion and component settings in your Repackager project, perform the following steps.

Task ***To configure advanced conversion options:***

1. Select **Advanced Settings** from the Repackager View List. The **Advanced Package Settings** view opens.
2. Under **Package Conversion**, select the package conversion options you want to use during conversion:

Option	Description
Use Editor path variables instead of physical source paths	When storing files in the InstallShield Editor project (`.ism`), the Wizard uses path variable locations whenever possible.
Display only the Welcome dialog box during installation	Only the Welcome dialog box is displayed when the Windows Installer package is run on a target machine. If this option is unchecked, the default UI sequence is displayed when the setup is installed.
Replace files with merge modules wherever possible	Following best practice rules, Repackager replaces components with comparable merge modules whenever possible.
Use the language captured by the Repackager as the language of the setup	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view).
Include files from Setup Intent scan	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project).

3. Under **Component Settings**, select the component settings options you want to use during conversion:

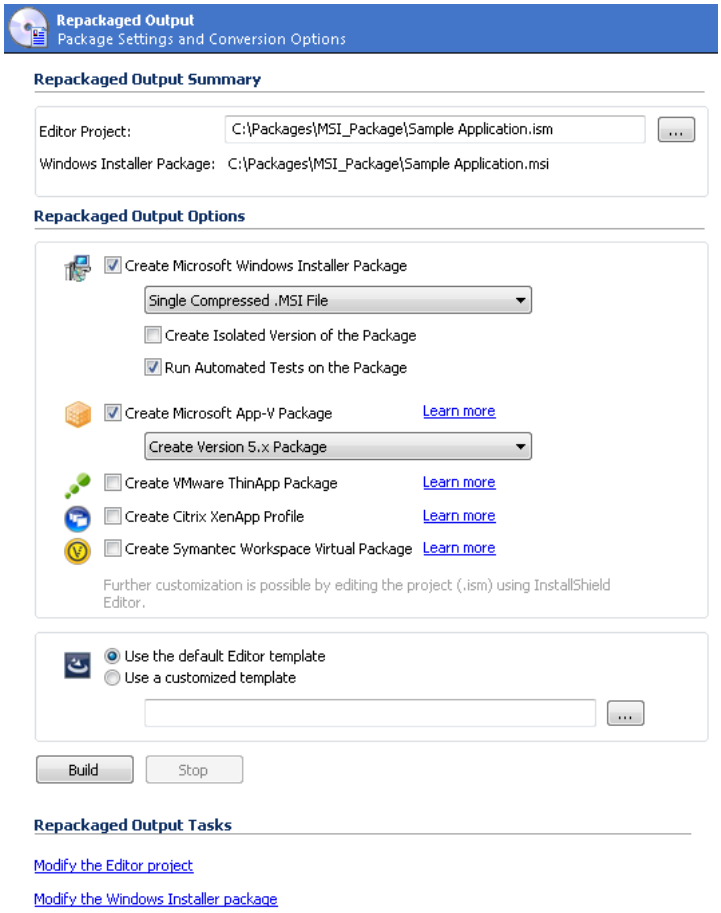
Option	Description
Mark components destined for the System folder as permanent	Executable files installed to the system folder (System32Folder) are marked as Permanent files and will not be uninstalled when the package is uninstalled. This eliminates ICE09 validation errors.
Mark components destined for the CommonFiles folder as shared	Executable files installed to the CommonFilesFolder (or a subfolder of CommonFilesFolder) are marked as shared files. This ensures that these components can coexist with DLLs installed by previous setups.
Map registry data to the appropriate COM tables	Setting this option reduces the number of ICE33 warnings that can occur during package validation, resulting from data not being mapped to the appropriate MSI tables.
Map registry data to the appropriate ODBC tables	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer.
Map NT Service events to the ServiceControl table	If selected, NT Service–related registry data is mapped to ServiceControl table instead of the Registry table.

Automatically Generating a Virtual Application During Repackager Project Build

You can simultaneously build an InstallShield Editor project (`.ism`), a Windows Installer package (`.msi`), a Microsoft App-V application (4.x or 5.x), a ThinApp application, a Citrix profile, and/or a Symantec virtual application from your Repackager project (`.irp`). To do this, you need to select options on the Repackager **Repackaged Output** view.

Task ***To automatically generate a virtual application during Repackager project build:***

1. In the Repackager interface, open a Repackager project.
2. Select **Repackaged Output** from the View List. The **Repackaged Output** view opens.



3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. Select the **Create Microsoft Windows Installer Package** option, and select the associated compression, isolation, and automated test options as described in Building a Windows Installer Package.

Important • *When building a virtual package, the* ***Create Microsoft Windows Installer Package*** *option* ***must*** *be selected. If it is not selected, the virtualization options are disabled.*

5. Select one or more of the virtual application options:
   - **Create Microsoft App-V Package**, and then also select one of the following from the list:
     - **Create Version 4.x Package**
     - **Create Version 5.x Package**
   - **Create VMware ThinApp Package**
   - **Create Citrix XenApp Profile**

Note • *If you would like to further customize the virtual application using the InstallShield Microsoft App-V Assistant, ThinApp Assistant, or Citrix Assistant, you can click the* ***Modify the Editor Project*** *link below to open this project in InstallShield Editor. This option is not available until after you build the Repackager project the first time.*

Note • *You can also use the Automated Application Converter to convert a Windows Installer package to a virtual package. See Performing Virtualization and Repackaging Using the Automated Application Converter in the AdminStudio Help Library.*

6. Select whether to use the default Editor template or a customized template, as described in Building a Windows Installer Package.
7. Select **Package Information** from the View List and set **Package Information** options as described in Building a Windows Installer Package.
8. Select **Advanced Settings** from the View List and select the options that you want to use, as described in Configuring Advanced Conversion Options.
9. Select **Repackaged Output** on the View List. The **Repackaged Output** view opens.
10. Click the **Build** button. The build process begins, and its progress is reported in the output window.

    When the build process is complete, a `Conversion completed` message appears in the output window, and a link to the build log file is provided.

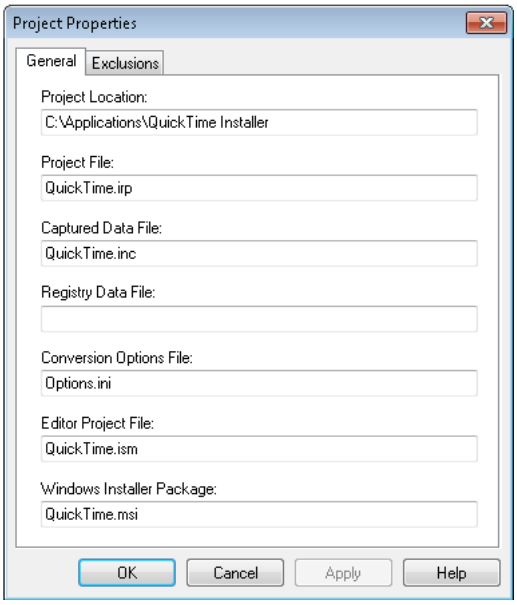
    - **If you chose the App-V 4.x or 5.x application option**, a folder named `App-VPackage` was created in the location you specified in the **Editor Project** field. This folder contains the App-V application for this package and all of its associated files, as described in Components of an App-V 4.x Package (.sft) or Components of an App-V 5.0 Package (.appv) in the AdminStudio Help Library.
    - **If you chose the ThinApp application option**, a folder named `ThinAppPackage` was created in the location you specified in the **Editor Project** field. This folder contains the ThinApp application for this package and all of its associated files, as described in Components of a ThinApp Application in the AdminStudio Help Library.
    - **If you chose the Citrix XenApp profile option**, a folder named `CitrixProfile` was created in the location you specified in the **Editor Project** field. This folder contains the Citrix profile for this package and all of its associated files, as described in About Citrix Profiles (.profile) in the AdminStudio Help Library.

Viewing Repackager Project Properties

You can view the properties of the currently open Repackager project by opening the Project Properties dialog box.

Task ***To view properties for the current Repackager project:***

1. Open a project in the Repackager interface.
2. From the **Project** menu, select **Properties**. The **General** tab of the **Project Properties** dialog box opens.



The following properties are listed:

- **Project Location**—The full path of the directory where the current Repackager project file (`.irp`) is located.
- **Project File** —The name of the current Repackager project file.
- **Captured Data File**—The name and location of the captured data file (`.inc`), which was either created by the Repackaging Wizard or during conversion of a Microfocus ZENworks project, Microsoft SMS project, or WinINSTALL project. The path is relative to the current Repackager project file.
- **Registry Data File**—The name and location of the file containing captured registry data. The path is relative to the current Repackager project file.
- **Conversion Options File**—The name and location of the `Options.ini` file, which contains an exhaustive list of all options you can use during conversion of the Repackager project to an InstallShield Editor project and Windows Installer package.
- **Editor Project File**—The name and location of the InstallShield Editor project file as set in the Product View (MSI Package). The path is relative to the current Repackager project file.
- **Windows Installer Package**—The name and location of the Windows Installer package. The path is relative to the current Repackager project file.

3. When finished viewing properties in the **General** tab, click OK.

Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project

Although an installation may have intended to install certain files, these files sometimes may not be installed—often because the files already exist on the target machine (either as the same version or a newer version). These files, although not installed or updated, are needed for the product to execute properly when the setup is run on a system that does not already have these files.

You can use the **Setup Intent Wizard** to detect file dependencies that may not be included in your Repackager project (`.irp`). The **Setup Intent Wizard** scans a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files.

To use the **Setup Intent Wizard**, perform the following steps:

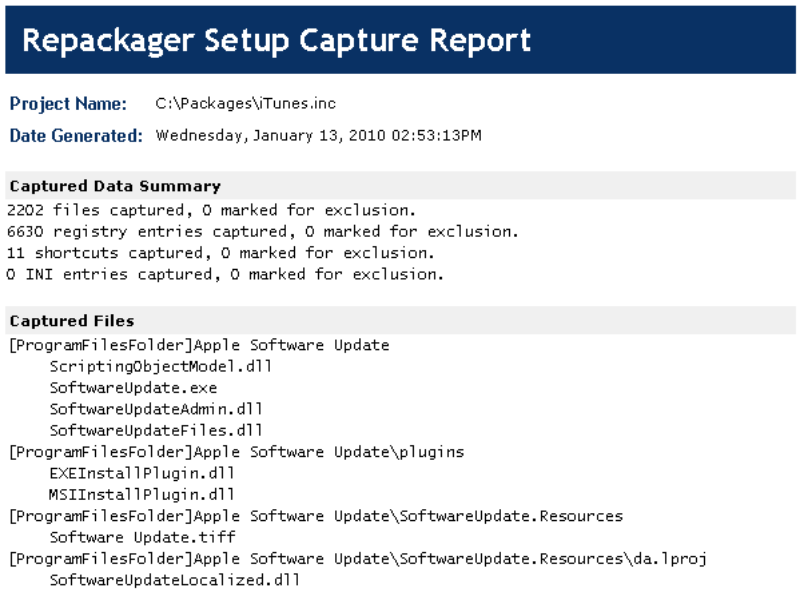
Task ***To detect file dependencies:***

1. From the Project menu, select Setup Intent Wizard. The **Welcome Panel** opens.
2. From the **Welcome** panel, click **Next**. The **Scanning Project Panel** opens.
3. Once scanning is finished, the **Results Panel** opens, listing new files that your setup requires.
4. From the **Results Panel**, select the files you want added to your Repackager project and click **Finish**.
5. Save your Repackager project.

Note • *Because the Setup Intent Wizard analyzes files in the Repackager project and searches for dependent files, you must run the Setup Intent Wizard from the same machine where repackaging was performed (with the Repackaging Wizard). You can then save the Repackager project and transfer it to another machine.*

Creating a Setup Capture Report for a Project

You can generate an HTML or text document that summarizes the data that was captured when a setup was repackaged.



Repackager Setup Capture Report

Project Name: C:\Packages\iTunes.inc

Date Generated: Wednesday, January 13, 2010 02:53:13PM

Captured Data Summary

2202 files captured, 0 marked for exclusion.
6630 registry entries captured, 0 marked for exclusion.
11 shortcuts captured, 0 marked for exclusion.
0 INI entries captured, 0 marked for exclusion.

Captured Files

[ProgramFilesFolder]Apple Software Update
    ScriptingObjectModel.dll
    SoftwareUpdate.exe
    SoftwareUpdateAdmin.dll
    SoftwareUpdateFiles.dll
[ProgramFilesFolder]Apple Software Update\plugins
    EXEInstallPlugin.dll
    MSIInstallPlugin.dll
[ProgramFilesFolder]Apple Software Update\SoftwareUpdate.Resources
    Software Update.tiff
[ProgramFilesFolder]Apple Software Update\SoftwareUpdate.Resources\da.lproj
    SoftwareUpdateLocalized.dll

Figure 9-2: Sample Repackager Setup Capture Report

The following information is available to be displayed in this report:

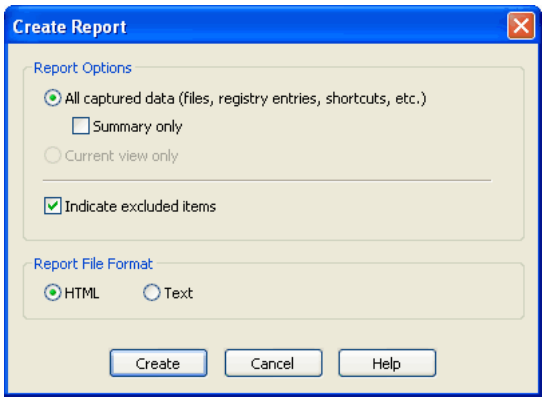
- Captured files

- Captured shortcuts
- Captured .ini file entries
- Captured Registry entries

The report also indicates which of the files, shortcuts, `.ini` file entries, or Registry entries, if any, have been marked for exclusion. Those marked for exclusion are not included in the Repackager project.

Task ***To create a report detailing captured data:***

1. From the **Project** menu, select **Create Report**. The **Create Report** dialog box opens.



2. Select whether you want the report to contain **All captured data** (all of the data collected during the entire capture), or just the **Current view**.
3. If you want the report to contain data from the entire capture, specify whether you want to just display summary information.
4. Specify whether you want to display excluded items in the report.
5. Select the file format for the report. You can generate an **HTML** report or a **Text** report.
6. Click **Create**. A Save As dialog box opens.
7. From the resulting Save As dialog box, browse to the location where you want to save the file, and provide a name for the report.
8. Click **Save**. The report is saved to the specified location and automatically opens.

Generating Software ID Tag Files During Repackaging

AdminStudio includes ISO/IEC 19770-2 software tagging support. ISO/IEC 19770-2 is an international standard for the creation of software identification tags.

AdminStudio adds software ID tag files—which contain both ISO 19770-2 compliant tag information and AdminStudio's extended tag information—to Windows Installer packages that it processes in two locations:

- **Packages built by Repackager**—By default, whenever Repackager builds a Windows Installer package (even when building one silently), a software ID tag file is created for that package.

- **Packages imported into the Application Catalog**—By default, tag files are created for each package that is imported into the Application Catalog. When Application Catalogs from versions of AdminStudio prior to 11.0 are upgraded, AdminStudio will, upon your approval, create tag files for all packages during upgrade. For more information, see Generating Software ID Tag Files During Package Import in the AdminStudio Help Library.

In both of these cases, AdminStudio stores the ISO tag file in an external transform file.

In this section, Repackager's support for creating and editing software ID tag files is described in the following topics:

- Enabling Software ID Tag Generation During Repackaging
- Viewing and Editing Software ID Tag Information in the Repackager Interface

Note • *For detailed information on AdminStudio's support for software ID tag files, see About Software ID Tag File Generation in the AdminStudio Help Library.*

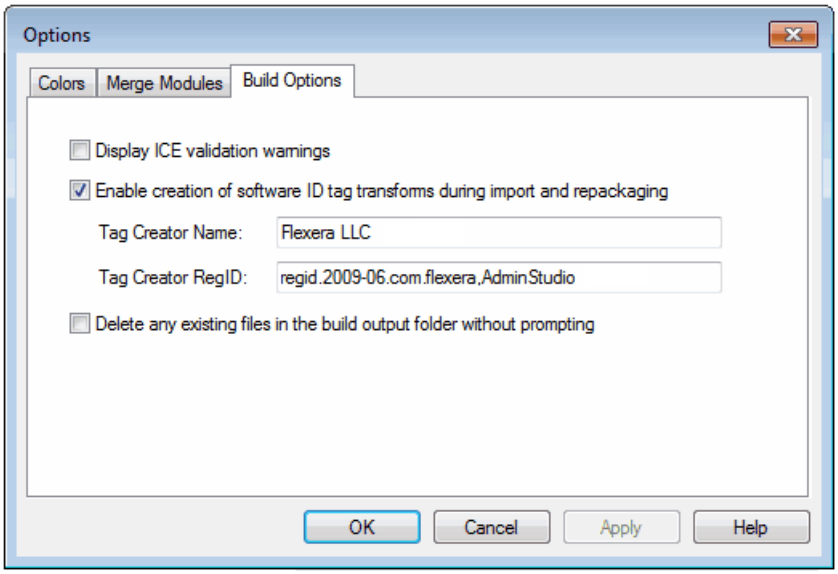
Enabling Software ID Tag Generation During Repackaging

On the **Build Options** tab of the Repackager **Options** dialog box, you can enable or disable automatic software tag file creation and can set the default values for **Tag Creator Name** and **Tag Creator RegID**.

Important • *Any changes that you make to the software tagging options on the **Build Options** tab of the Repackager **Options** dialog box will also automatically be made to the options on the **General Options > Import Options > Software Tagging** tab of the Application Catalog **Options** dialog box.*

Task ***To set software tagging options in Repackager:***

1. Launch Repackager.
2. On the **Tools** menu, click **Options**. The **Options** dialog box opens.
3. Open the **Build Options** tab.



4. To instruct AdminStudio to automatically create a transform file containing software tag file(s) for Windows Installer packages that are imported into the Application Catalog or built using Repackager, make sure that the **Enable creation of software ID tag transforms during import and repackaging** option is selected. By default, this option is selected.

Note • *Whenever a Windows Installer package is imported into the Application Catalog or built using Repackager, AdminStudio creates a software ID tag file (which is stored in the Application Catalog), but if the* ***Enable creation of software ID tag transforms during import and repackaging*** *option it not selected, AdminStudio does not create the transform.*

5. In the **Tag Creator Name** field, enter a name to identify the creator of the software ID tag files that will be created by AdminStudio. By default, the value is `Flexera`.

Note • *For more information, see About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields in the AdminStudio Help Library.*

6. In the **Tag Creator RegID** field, enter an ID to uniquely identify the creator of the software ID tag files that will be created by AdminStudio, using the following format:

```
regid.YYYY-MM.ReversedDomainName,optional_division
```

For example:

```
regid.2009-06.com.yourcompany,GlobalProductDivision
```

By default, the value is AdminStudio's RegID:

```
regid.2009-06.com.flexera,AdminStudio
```

Note • *For more information on RegIDs, see About Software Tagging RegIDs in the AdminStudio Help Library.*

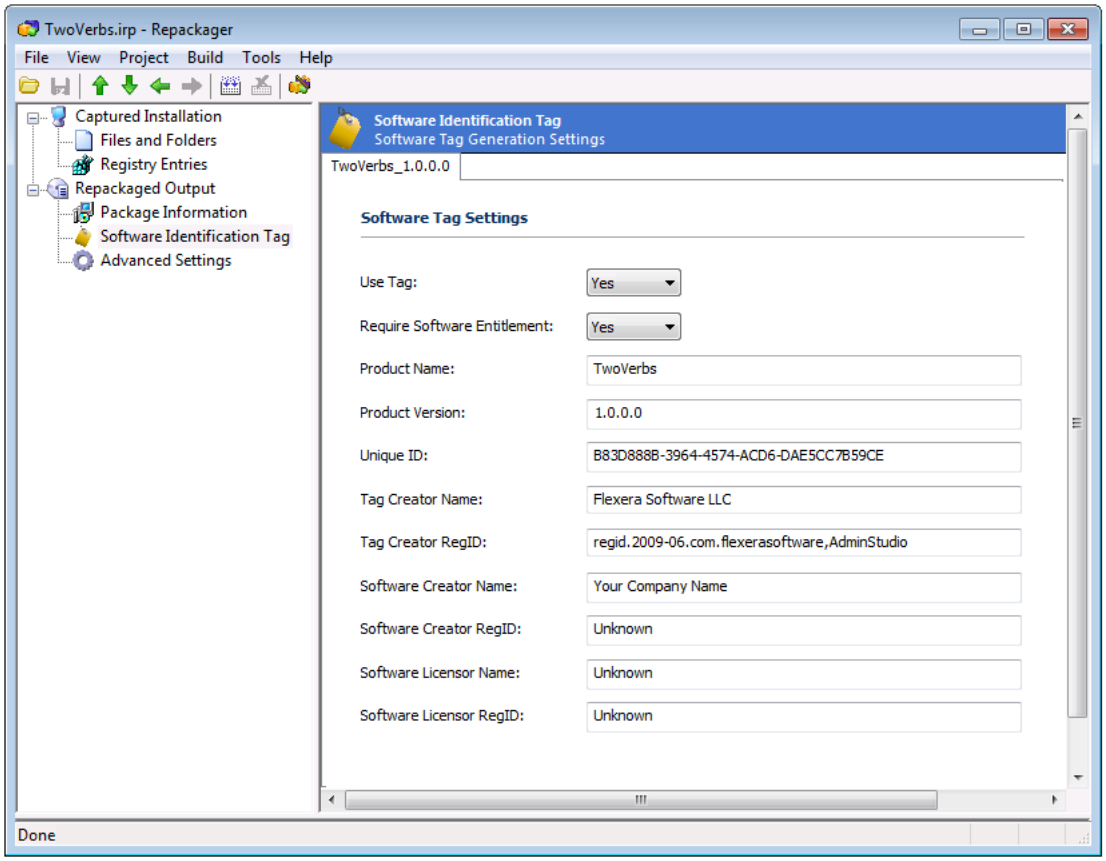
Viewing and Editing Software ID Tag Information in the Repackager Interface

When you use Repackager to convert a legacy package to a Windows Installer package, by default a tag file is generated for each package when the Windows Installer package is built.

You can view and edit tag information in the Repackager interface's **Software Identification Tag** view.

Task ***To view and edit software ID tag file information in Repackager:***

1. Open a Repackager project in the Repackager interface.
2. Under **Repackaged Output**, select the **Software Identification Tag** node. The **Software Identification Tag** view opens.



3. Edit the view and edit the information in the fields, as described in Software Identification Tag View.
4. To save your edits, select **Save** on the **File** menu.

Saving Repackager Projects

Edition •

To save a Repackager project, perform the following steps:

Task ***To save the current Repackager project:***

1. Select **Save** from the **File** menu.

   or

2. Click the Save button () on the toolbar.

Task ***To save the current Repackager project under a different name:***

Select **Save As** from the **File** menu.

Opening InstallShield Editor from Repackager

After building your Repackager project into a Windows Installer package and/or an InstallShield Editor project, you may want to launch InstallShield Editor for additional modifications.

Task ***To launch the generated InstallShield Editor project (.ism) in InstallShield Editor:***

From the Repackager **Project** menu, select **Edit InstallShield Project**. If installed, InstallShield Editor opens the project file.

Task ***To launch the generated Windows Installer package (.msi) in InstallShield Editor:***

From the Repackager **Project** menu, select **Edit Windows Installer Package**. If installed, InstallShield Editor opens the package in Direct MSI Edit mode.

Isolating Windows Installer Packages

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

When building a Windows Installer package from your Repackager project, you can also choose to create an isolated version of that package by selecting an option on the **Repackaged Output** view.

Information about application isolation is presented in the following topics:

- About Application Isolation
- About Assemblies
- About Manifests
- About Digital Certificates

- Setting Isolation Options
- Building an Isolated Windows Installer Package

About Application Isolation

Application isolation, which is a technique used to minimize the dependencies of an application on system components or dynamic elements, is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components and dynamic elements with which it was originally developed and tested.

Isolation is accomplished by:

- Providing DLLs and other shared components for specific applications, *and*
- Placing information traditionally stored in the Registry into other files that specify the locations of these isolated components.

Application isolation provides increased stability and reliability for applications because they are unaffected by changes caused by installation and ongoing maintenance of other applications on the system.

Depending on the isolation options chosen, you can partially or totally isolate an application. When using assemblies and manifests to isolate applications, the assemblies can be updated following deployment without necessitating application reinstallation.

Reasons to Isolate Applications

You would want to isolate an application if:

- You want to resolve incompatibilities between different versions of shared components.
- You want to reduce the complexity of the installation by storing COM activation data in a manifest instead of the registry.
- You want to insulate the application from changes to shared components.

Reasons Not to Isolate an Application

You would not want to isolate an application if, following application isolation, you discover that the application no longer works because of an internal dependency on a component that has been moved during the isolation process.

Tip • *Following isolation, you can use the Dynamic Dependency Scanner in InstallShield Editor to verify isolated files are loaded from a different directory.*

Isolating Windows Installer Packages Using Application Isolation Wizard

In addition to being able to generate an isolated version of a repackaged setup immediately after the build step in Repackager, you can also use Application Isolation Wizard to isolate a Windows Installer package.

Application Isolation Wizard is a stand-alone tool which accepts a Windows Installer package as input and outputs a new, isolated Windows Installer package.

The Application Isolation Wizard provides a user interface experience that allows the user to extend the initial "dependency scanning" process for identifying file isolation candidates, while in Repackager you specify your assembly and digital signing isolation options on the Isolation Options dialog box, and then those selections are applied to all isolated packages created by Repackager.

For more information, see Isolating Applications Using Application Isolation Wizard in the AdminStudio Help Library.

About Assemblies

Assemblies are DLLs or other portable executable files that applications require to function. These can be either shared or private. Private assemblies are typically stored in the same directory as the application they support. Shared assemblies are stored in the `WinSxS` directory, and are digitally signed.

By creating manifests for assemblies, Repackager allows you to create self-contained applications that can use different versions of the same DLL or other portable executable, without any version conflicts.

Shared Assemblies

Shared assemblies are assemblies available to multiple applications on a computer. Applications that require these assemblies specify their dependence within a manifest. Multiple versions of shared assemblies can be used by different applications running simultaneously.

These assemblies are stored in the `WinSxS` directory, and must be digitally signed for authenticity. After deployment, the version of shared assemblies can be changed, allowing for changes in dependencies.

Private Assemblies

Private assemblies are assemblies created for exclusive use by an application. They are accompanied by an assembly manifest, which contains information normally stored in the registry. Private assemblies allow you to totally isolate an application, eliminating the possibility that dependent files may be overwritten by other applications.

These assemblies are always stored in the same location as their associated executable.

About Manifests

Manifests, which are used during isolation, are XML files that describe an application. Repackager can create two types of manifests: application manifests and assembly manifest.

Application Manifests Describe an Isolated Application

Application manifests are XML files that describe an isolated application. This descriptive information includes the relationship between the application and its dependent files.

Typically, the naming convention for a manifest is:

```
ApplicationName.Extension.manifest
```

For example, if the application was `HelloWorld.exe`, the manifest file is called:

```
HelloWorld.exe.manifest
```

Assembly Manifests Describe an Application's Assemblies

Assembly manifests are XML files that describe an application's assemblies. This includes components such as DLLs.

Information stored in the assembly manifest, such as COM registration information, ProgIDs, etc., is usually stored in the Registry. However, by making it independent from the Registry, only that application can use the dependent files described in the manifest. This enables you to have multiple versions of the same DLL or other portable executable file on a system without generating compatibility conflicts.

Typically, the naming convention for a manifest is:

```
AssemblyName.Extension.manifest
```

For example, if the component was `Goodbye.dll`, the manifest file is called:

```
Goodbye.dll.manifest
```

Manifests as New Components

When you create manifests, Repackager supports putting them into new components. If you do not select the **Create new component for each assembly** option on the **Manifest Options** tab of the **Isolation Options** dialog box, the manifest will be added to the same component as the assembly.

About Digital Certificates

Digital certificates identify you and/or your company to end users to assure them the assembly they are about to use has not been altered. They are issued by a certification authority such as VeriSign, or created using a combination of software publishing credentials (`.spc`) and a private key (`.pvk`), both also issued by a certification authority. The certificate includes the public cryptograph key, and, when used in combination with a private key, can be used by end users to verify the authenticity of the signor.

The following digital certificate concepts are defined in this topic:

- Private Keys
- Software Publishing Credentials
- Using a Certificate Store
- Creating a Certificate File

Private Keys

A private key (a file with the extension .pvk) is granted by a certification authority. Repackager uses the private key you enter in the **Digital Signature** tab of the Isolation Options dialog box to digitally sign your shared assembly and ensure end users of its content's authenticity.

The .spc (Software Publishing Credentials) file and .pvk file you enter in the Digital Signature tab compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Software Publishing Credentials

You must supply a certification authority with specific information about your company and software to obtain software publishing credentials in the form of an .spc file. Your software publishing credentials are used to generate a digital signature for your assembly.

The `.spc` file and `.pvk` (private key) file you enter as in the Digital Signature tab of the Advanced Options dialog box compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Using a Certificate Store

To perform code signing, both private key and software publishing credential information must be supplied. This must occur each time a package is signed. Most server operating systems store a certificate locally on the computer that the user used to request the credential information.

Instead of having to store credential files on each of the user computers, you can create a Certificate Store, a storage location which will have numerous certificates, which enables all users or computers with adequate permissions to retrieve the certificate as needed.

Using a Certificate Store allows you to associate the same credentials and private key files with multiple packages. This simplification is particularly useful when isolating applications, as typically the code signing information will be identical for all shared assemblies. Ultimately, the Certificate Store removes the burden of managing private key and software publishing credential information.

Creating a Certificate File

You can create a certificate file from the constituent PVK and SPC files and import it into the Certificate Store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the Certificate Store.

Caution • *Certificate files must be 2048-bit or higher. For more information, see the article: Assembly Signing Example on the Microsoft Developer Network website.*

Setting Isolation Options

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

On the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following Repackager isolation options:

- **Assembly Options**—Specify the type of assemblies Repackager will create: private side-by-side assemblies in the application folder or shared side-by-side assemblies in the `WinSxS` folder. You can also specify the assembly naming conventions. See Specifying Manifest Options.
- **Digital Signature Options**—You can configure the certificate information required when using shared assemblies. This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly. See Setting Digital Signature Options for Shared Assemblies.

Specifying Manifest Options

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following options:

- Selecting the Assembly Type
- Specifying the Assembly Naming Conventions

Note • *For more information on assemblies and manifests, see About Assemblies and About Manifests.*

Selecting the Assembly Type

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the type of assemblies Repackager will create: private side-by-side assemblies in the application folder or shared side-by-side assemblies in the `WinSxS` folder.

Task ***To select the assembly type:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.
3. Select one of the following **Assembly Type** options:
   - Create private side-by-side assemblies in the application folder.
   - Create shared side-by-side assemblies in `WinSxS` directory.

Note • *Manifests for shared assemblies must be digitally signed. See Setting Digital Signature Options for Shared Assemblies.*

Note • *The modifications you make on the **Isolation Options** dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Specifying the Assembly Naming Conventions

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the type of naming conventions Repackager will use when creating assemblies.

Task ***To set the default naming convention for assemblies:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.

3. In the **Assembly Naming Conventions** area, enter your **Company** name and **Division**. These two fields create the default assembly naming convention (in the form `Company.Division.Assembly` followed by a number).
4. If you want to create a new component for each assembly, select the **Create new component for each assembly** option.

Assemblies created during application isolation will follow the naming convention as specified.

Note • *The modifications you make on the **Isolation Options** dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Setting Digital Signature Options for Shared Assemblies

You can configure the certificate information required when using shared assemblies on the **Digital Signature** tab of the **Isolation Options** dialog box.This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

Note • *The modifications you make on the Isolation Options dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Note • *For more information, see About Digital Certificates.*

Task ***To set digital signature options:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.
3. Open the **Digital Signatures** tab.
4. Click the Browse (...) button next to the **Certificate File** field and navigate to the certificate file you are using to sign assemblies.

   A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.

5. In the **Code Signing Technology** area, select the type of code signing technology you want to use for the digital signature. You can use one of the following technologies:

Technology	Description
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: ● **SPC File**—Specify the name and location of your software publishing credentials file (`.spc`). ● **PVK**—Specify the name and location of your private key file (`.pvk`). In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as VeriSign with specific information about your company and software.
Certificate Name in the store	Select this option to use the name of an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary. As an alternative to providing `.spc` and `.pvk` files, you can specify the certificate name as it appears in the certificate store.

Building an Isolated Windows Installer Package

To reduce versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested, select the **Create Isolated Version of the Package** option on the Repackager **Repackaged Output** view. An additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of:

appname`.isolated.msi`

For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages.

Configuring Exclusions

Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, .ini files, or shortcuts excluded at this point are not included in the Repackager project.

There are two methods of configuring exclusions:

- Configuring Exclusions Using Repackager
- Configuring Exclusions Using the Exclusions Editor

Configuring Exclusions Using Repackager

There are three types of exclusions used when repackaging a legacy installation:

Table 9-3 • Repackager Exclusion Types

Exclusion Type	Description
Repackaging Exclusions	Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, .ini files, or shortcuts excluded at this point are not included in the Repackager project. Therefore, if you exclude a directory you later need, you need to repackage the legacy setup again. The Repackager best practice is to capture everything using the Repackaging Wizard, and then exclude visually in the Repackager Interface. This way, you avoid having to run the Repackaging Wizard again if you accidentally exclude necessary files. In some cases, you may want to avoid capturing specific data types during repackaging. For example, your organization may never want to capture shortcuts. You can disable capture of shortcuts during repackage time, thereby eliminating the need to exclude them later. In Snapshot mode, you may want to limit the analysis to a certain directory to reduce the time it takes to capture the initial and final snapshot.
Project Exclusions	Each Repackager project can use a project exclusion list which marks files, registry entries, shortcuts, and .ini files as excluded in the Repackager project. If your process dictates that you capture everything and only exclude items in the Repackager Interface, then you should set up commonly captured but unnecessary items from the project by default. Because all the data from the original capture is intact, if you accidentally exclude necessary files, you can always reinclude them from the Repackager Interface and quickly rebuild your Windows Installer package.
Individual Project Exclusions	Because each project is different, and may require you to make decisions as to whether certain captured data is necessary, you can also selectively exclude or reinclude items on a per-package basis. These individual project exclusions allow you a fine-level of control as you prepare to build your Windows Installer package from the Repackager project.

Excluding Files

To exclude a captured file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a captured file from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders View** opens.
2. Expand the directory tree and select the directory containing the file you want to exclude.
3. In the file list, right-click the file and then click **Exclude**.

Excluding All Files in a Directory

To exclude all captured files in a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude all captured files in a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders** view opens.
2. Expand the directory tree and select the directory containing the files you want to exclude.
3. Right-click the directory and then click **Exclude**.

Excluding Directories and Subdirectories

To exclude all captured files and subdirectories within a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude all captured files and subdirectories within a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders** view opens.
2. Expand the directory tree to display the directory containing the files and subdirectories you want to exclude.
3. Right-click the directory and then click **Exclude All**.

Adding Files and Folders to the Global Exclusions List from the Files and Folders View

You can add a file or a directory of files to the Repackager global exclusion list (`isrepackager.ini`) from the **Files and Folders** view of the Repackager interface.

Note • *You can also configure exclusions using the Exclusions Editor, as described in Configuring Exclusions Using the Exclusions Editor.*

To add items to the global exclusions list from the **Files and Folders** view, perform the following steps.

Task ***To add files and/or folders to the exclusions list:***

1. Select **Files and Folders** from the View List. The **Files and Folders** view opens.
2. Expand the directory tree to display the directory containing the files and subdirectory you want to exclude.
3. Right-click on a directory or file and then select **Add to Exclusions** from the shortcut menu.

Note • *If you selected a directory, you are then prompted to click **Yes** or **No** to indicate whether you want to also add the directory's subdirectories to the exclusion list.*

The directory name and/or file name are now displayed in red to indicate that they are excluded.

4. To remove a previously selected item from the global exclusion list, right-click on the item and then select **Remove from Exclusions** from the shortcut menu.

Excluding Registry Keys

To exclude a registry key from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a registry key from the InstallShield Editor project and Windows Installer package:***

1. Select **Registry Entries** from the View List. The **Registry Entries** view opens.
2. Expand the Registry tree to display the registry key you want to exclude.
3. Right-click the registry key and then click **Exclude**.

Excluding Registry Values

To exclude a captured registry value from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a captured registry value from the InstallShield Editor project and Windows Installer package:***

1. Select **Registry Entries** from the View List. The **Registry Entries** view opens.
2. Expand the Registry tree and select the registry key containing the value you want to exclude.
3. In the **Registry Value** list, right-click the value and then click **Exclude**.

Excluding .ini Files

To exclude a captured .ini file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a captured .ini file from the InstallShield Editor project and Windows Installer package:***

1. Select **INI Files** from the View List. The **INI Files** view opens.
2. Expand the **INI Files** tree to display the .ini file you want to exclude.
3. Right-click the .ini file and then click **Exclude**.

Excluding .ini File Sections

To exclude a section in a captured .ini file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a section in a captured .ini file from the InstallShield Editor project and Windows Installer package:***

1. Select **INI Files** from the View List. The **INI Files** view opens.
2. Expand the **INI Files** tree to display the .ini file containing the section you want to exclude.
3. Right-click the section and then click **Exclude**.

Excluding Shortcuts

To exclude a captured shortcut from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude a captured shortcut from the InstallShield Editor project and Windows Installer package:***

1. Select **Shortcuts** from the View List. The **Shortcuts** view opens.
2. Expand the Shortcuts tree to display the shortcut you want to exclude.
3. Right-click the shortcut and then click **Exclude**.

Excluding All Shortcuts in a Directory

To exclude all captured shortcuts in a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude all captured shortcuts in a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Shortcuts** from the View List. The **Shortcuts** view opens.
2. Expand the Shortcuts tree to display the directory containing the shortcuts you want to exclude.
3. Right-click the directory and then click **Exclude**.

Excluding Shortcuts from Subdirectories

To exclude all captured shortcuts within a directory or its subdirectories from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task ***To exclude all captured shortcuts within a directory or its subdirectories from the InstallShield Editor project and Windows Installer package:***

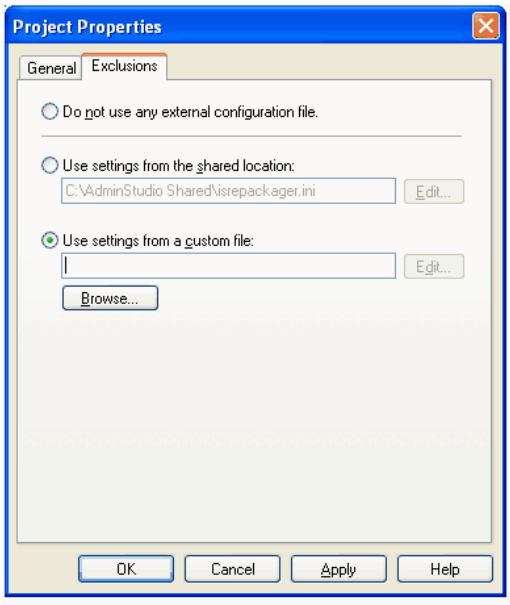
1. Select **Shortcuts** from the View List. The **Shortcuts** view opens.
2. Expand the **Shortcuts** tree to display the directory containing the shortcuts and/or subdirectories containing shortcuts you want to exclude.
3. Right-click the directory and then click **Exclude All**.

Specifying the External Configuration File

To specify an external configuration file which you want to use as a filter when converting legacy setups, perform the following steps.

Task ***To specify an external configuration file which you want to use as a filter when converting legacy setups:***

1. From the Repackager **Project** menu, select **Properties**. The **General** tab of the **Project Properties** dialog box opens.
2. Open the **Exclusions** tab.



3. Select the **Use settings from the shared location** or the **Use settings from a custom file** option. The **Browse** button for that option is activated.
4. Click **Browse** and select the configuration file you want to use.

Tip • *After you select a configuration file, the **Edit** button is activated, enabling you to open the file in the Exclusions Editor.*

5. Click **OK**.

When you apply a configuration file, Repackager automatically updates all views to reflect the configuration file's exclusions. However, if you have already excluded items using Repackager, item states are retained.

Modifying External Configuration Files

To configure an external configuration file, perform the following steps.

Task ***To configure an external configuration file:***

1. From Repackager's **Project** menu, click **Properties**. The **General** tab of the **Project Properties** dialog box opens.
2. Select the **Exclusions** tab.
3. Select the file you want to modify in either the **Use settings from the shared location** option or **Use settings from a custom file** options.
4. Click **Edit**. The **Exclusions Editor** opens.
5. Make necessary modifications using the **Exclusions Editor**.
6. When you finish editing the configuration file, click **OK**.
7. Click **OK** to close the **Project Properties** dialog box.

When you apply a configuration file, Repackager automatically updates all views to reflect the configuration file's exclusions. However, if you have already excluded items using Repackager, item states are retained.

Configuring Exclusions Using the Exclusions Editor

The Exclusions Editor allows you to configure three types of exclusions: Repackaging, Project, and OS Snapshot.

Repackaging Exclusions

Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, `.ini` files, or shortcuts excluded at this point are not included in the Repackager project. Therefore, if you exclude a directory you later need, you need to repackage the legacy setup again.

The Repackager best practice is to capture everything using the Repackaging Wizard, and then exclude visually in the Repackager Interface. This way, you avoid having to run the Repackaging Wizard again if you accidentally exclude necessary files.

In some cases, you may want to avoid capturing specific data types during repackaging. For example, your organization may never want to capture shortcuts. You can disable capture of shortcuts during repackage time, thereby eliminating the need to exclude them later. In Snapshot mode, you may want to limit the analysis to a certain directory to reduce the time it takes to capture the initial and final snapshot.

Project Exclusions

Each Repackager project can use a project exclusion list which marks files, registry entries, shortcuts, and `.ini` files as excluded in the Repackager project. If your process dictates that you capture everything and only exclude items in the Repackager Interface, then you should set up commonly captured but unnecessary items from the project by default. Because all the data from the original capture is intact, if you accidentally exclude necessary files, you can always reinclude them from the Repackager Interface and quickly rebuild your Windows Installer package.

OS Snapshot Exclusions

Like pre-capture repackaging exclusions, you can use the Exclusions Editor to configure exclusions to apply during the capture of OS snapshots. However, to maximize the usefulness of OS snapshots, you should avoid editing the default snapshot exclusion list (ISSnapshot.ini).

Exclusions and Repackager

Exclusions in Repackager refer to files, registry entries, shortcuts, and `.ini` files that are marked as excluded in the Repackager Interface by default when you open a Repackager project or if you change your exclusions file. The captured data is only marked as excluded and not ignored or discarded during capture. You can create an exclusion file in the Exclusions Editor, and link it to Repackager from the **Exclusions Tab** of the **Project Properties** dialog box in Repackager.

Exclusions and the OS Snapshot Wizard

When using the Exclusions Editor to configure analysis options for capturing OS snapshots, you are creating an exclusion list for files, directories, `.ini` files, `.ini` file sections, and registry data. Items in the exclusion list are not captured during the OS snapshot process, and will not be included in the OS snapshot file which is created.

Launching Exclusions Editor

The Exclusions Editor can be launched either within the Repackager interface or outside of Repackager. You can edit the default exclusions file, `isrepackager.ini`, using either interface.

Note • *You can also add items to the default exclusions file from the* ***Files and Folders*** *view of the Repackager interface. See Adding Files and Folders to the Global Exclusions List from the Files and Folders View.*

However, if you want to create a new, custom exclusions file, you must launch the Exclusions Editor outside of Repackager.

- Launching Exclusions Editor Outside of Repackager
- Launching Exclusions Editor Within Repackager

Launching Exclusions Editor Outside of Repackager

To launch the Exclusions Editor outside of the Repackager interface, perform the following steps.

Task ***To add a file to the exclusion list:***

1. Launch the Exclusions Editor by locating and executing the following file:

   `[AdminStudioInstallDirectory]\Repackager\AnalysisOptions.exe`

   The **Files** tab of the Exclusions Editor opens.

2. Perform one of the following to open an exclusions file:

   - **Shared Exclusions**—To edit the shared exclusions file, on the **Files** menu, point to **Open** and click **Shared Exclusions**. The exclusions in the shared exclusions file are now listed on the **Files** tab.
   - **Custom Exclusions**—To create a new custom exclusions file, on the **Files** menu, click **New**. A default set of exclusions is listed.

3. Make edits to the file.
4. Save the file by selecting **Save** on the **File** menu.
5. If you were creating a custom exclusions file, specify a name and location for this exclusions file and click **Save**.

Launching Exclusions Editor Within Repackager

To add a file to the exclusion list, perform the following steps.

Task ***To add a file to the exclusion list:***

1. Launch Repackager and open a project.
2. On the **Project** menu, click **Properties**. The **Project Properties** dialog box opens.
3. On the **Exclusions** tab, do one of the following:

   - To edit the default exclusions file, select **Use settings from the shared location** and click **Edit**.
   - To edit a custom exclusions file, select **Use settings from a custom file**, browse to the file you want to open (if it is not listed), and click **Edit**.

   The **Files** tab of the Exclusions Editor opens, with the appropriate exclusions file open.

4. Make edits to the file.
5. Save the file and close the Exclusions Editor by clicking **OK**.

Note • *Note that when opening the Exclusions Editor from within Repackager, there is no* ***File*** *menu displayed, meaning that you can only edit an existing exclusions file; you cannot create a new exclusions file.*

Excluding Files

You use the Exclusions Editor to create an exclusion list for files so that those files are not captured during the OS snapshot process, and will not be included in the OS snapshot file.

Task ***To add a file to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor.
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. Enter or browse to the directory **Path** containing the file you want to exclude.
4. Enter the name of the file you want to exclude, or browse to it by clicking the Browse (...) button to the right of the **Excluded Files** field. If you want to exclude multiple files from the same directory, separate them with pipes (|). If you want to exclude all files in a directory, enter an asterisk (*).
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor.

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding Files with Specific Extensions

To exclude files with specific extensions from the exclusion list, perform the following steps.

Task ***To exclude files with specific extensions from the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, enter or browse to the directory containing the files you want to exclude. If you want to exclude files with a certain extension from all directories, enter an asterisk (*) for the **Path** value.
4. Enter an asterisk followed by the extension you want excluded in the **Excluded Files** field. For example, if you want to exclude all .bak files, enter *.bak. If you want to exclude multiple file types from the same directory (or from all directories), separate each exclusion with a pipe (|).
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding Directories

To add a directory to the exclusion list, perform the following steps.

Task ***To add a directory to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, enter or browse to the directory **Path** containing the files you want to exclude.
4. Enter an asterisk (*) in the **Excluded Files** field.
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Editing Existing File Exclusions

To edit an existing file exclusion, perform the following steps.

Task ***To edit an existing file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Select the appropriate exclusion and click **Edit**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, modify the **Path** and **Excluded Files** information.
4. Click **OK** to close the **File Exclusion Information** dialog box. The edited exclusion is listed in the **Files** tab.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Removing File Exclusions

To remove an existing file exclusion, perform the following steps.

Task ***To remove an existing file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Select the appropriate exclusion and click **Delete**.
3. Confirm the exclusion by clicking **OK**. The deleted exclusion is removed from the list.
4. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding .ini Files

To add an .ini file to the exclusion list, perform the following steps.

Task ***To add an .ini file to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Click **New**. The **INI File Exclusion Information** dialog box opens.
4. Enter or browse to the `.ini` file you want to exclude.
5. If there are specific sections you want to exclude from the `.ini` file, put the section names in brackets ([]) and separate them with pipes (|) in the **Excluded Sections** field. If you want to exclude all sections, put an asterisk (*) in the **Excluded Sections** field.
6. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the list on the **INI Files** tab.
7. Save the exclusions file as described in Launching Exclusions Editor.

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Excluding Sections from .ini Files

To add a specific .ini file section to the exclusion list, perform the following steps.

Task ***To add a specific .ini file section to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Click **New**. The **INI File Exclusion Information** dialog box opens.
4. Enter or browse to the `.ini` file containing the section you want to exclude.
5. To exclude a specific `.ini` file section, enter the section name in brackets ([]) in the **Excluded Sections** field. If there are multiple sections, separate them with pipes (|).
6. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the INI Files and Sections Excluded During Analysis dialog box.
7. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the list on the **INI Files** tab.
8. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Editing Existing .ini File Exclusions

To edit an existing .ini file exclusion, perform the following steps.

Task ***To edit an existing .ini file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor.
2. Open the **INI Files** tab.
3. Select the appropriate exclusion and click **Edit**. The **INI File Exclusion Information** dialog box opens.
4. In the **File Exclusion Information** dialog box, modify the **INI File** and **Excluded Sections** information.
5. Click **OK** to close the **INI File Exclusion Information** dialog box. The edited exclusion appears in the list on the **INI Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Removing .ini File Exclusions

To delete an existing .ini file exclusion, perform the following steps.

Task ***To delete an existing .ini file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Select the appropriate exclusion and click **Delete**.
4. Confirm the exclusion by clicking **OK**. The deleted exclusion is removed from the list.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Excluding Registry Data

To add registry data to the exclusion list, perform the following steps.

Task ***To add registry data to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Click **New**. The **Choose Registry Key** dialog box opens.
4. Enter or browse to the registry key you want to exclude and click **OK**. The key is added to the list on the **Registry** tab.
5. If you want to exclude a certain value in the key, select it from the list and click Edit. The **Edit Registry Key** dialog box opens.
6. Provide the **Value Name** you want to exclude, and click **OK** to close the dialog box. The exclusion information is reflected in the list on the **Registry** tab.
7. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

Editing Existing Registry Exclusions

To edit existing registry exclusions, perform the following steps.

Task ***To edit an existing registry exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Select the registry key that you want to edit and click **Edit**. The **Edit Registry Key** dialog box opens.
4. Modify the exclusion as necessary and click **OK**. The edited information is reflected in the list on the **Registry** tab.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

Removing Registry Exclusions

To delete an existing registry exclusion, perform the following steps.

Task ***To delete an existing registry exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Select the registry key that you want to delete and click **Delete**.
4. Confirm the deletion by clicking **OK**. The deleted exclusion is removed from the list.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

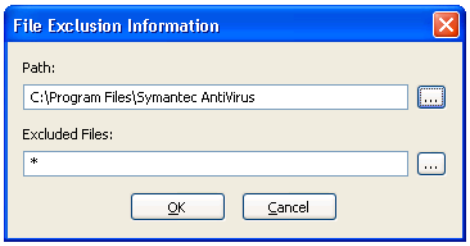
Repackaging and Anti-Virus Software

Any machine that you use to repackage most likely has anti-virus software installed on it, even a “clean” machine. During repackaging, the real-time virus detection feature of anti-virus software could automatically update various cached files in its directories.

Therefore, in order to avoid repackaging errors when using the Snapshot repackaging method, you should exclude the software directories containing your anti-virus software.

Task ***To exclude anti-virus software directories:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.



3. Enter or browse to the directory **Path** containing the anti-virus files that you want to exclude. For example, if you wanted to exclude Symantec AntiVirus software, you would select the following directory:

   `C:\Program Files\Symantec AntiVirus`
4. Enter an asterisk (*) in the **Excluded Files** field.
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusions appear on the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Important • *It is strongly recommended that you leave your anti-virus software running during repackaging.*

Creating an InstallShield Editor Template to Use Within Repackager

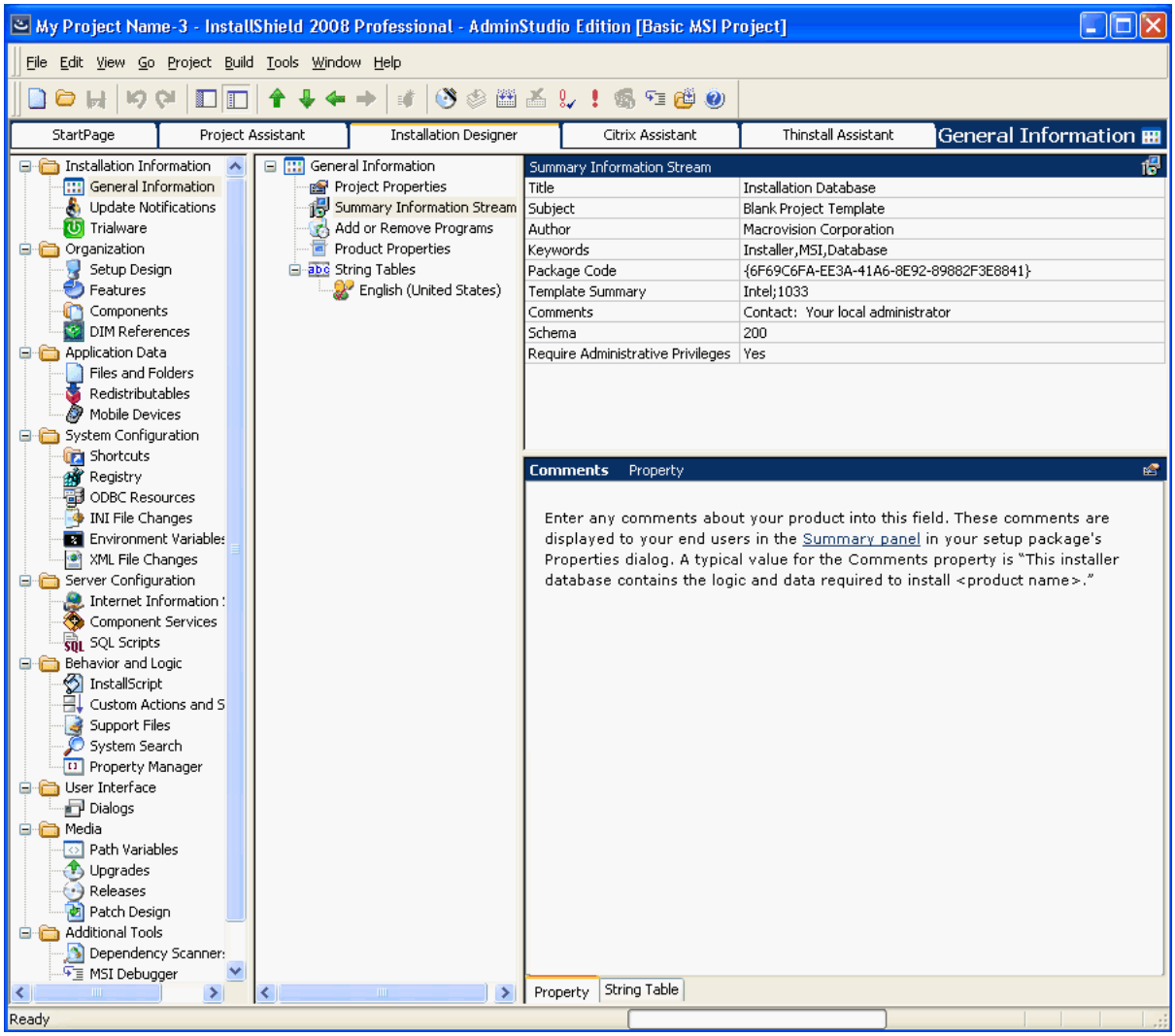
One of the main reasons you use AdminStudio is to significantly reduce the time it takes to package an application for deployment. You can use the following procedure to speed up the packaging process even more.

You can create an InstallShield Editor template that you can use within the Repackager interface to save additional time when customizing a package. By using this template, all future InstallShield Editor `.ism` project files generated by Repackager will contain the company-specific default settings that were specified in the template. Using a template is also beneficial for organizations with multiple packagers, since it helps enforce consistency by enabling all packagers to make the same standard customizations to packages.

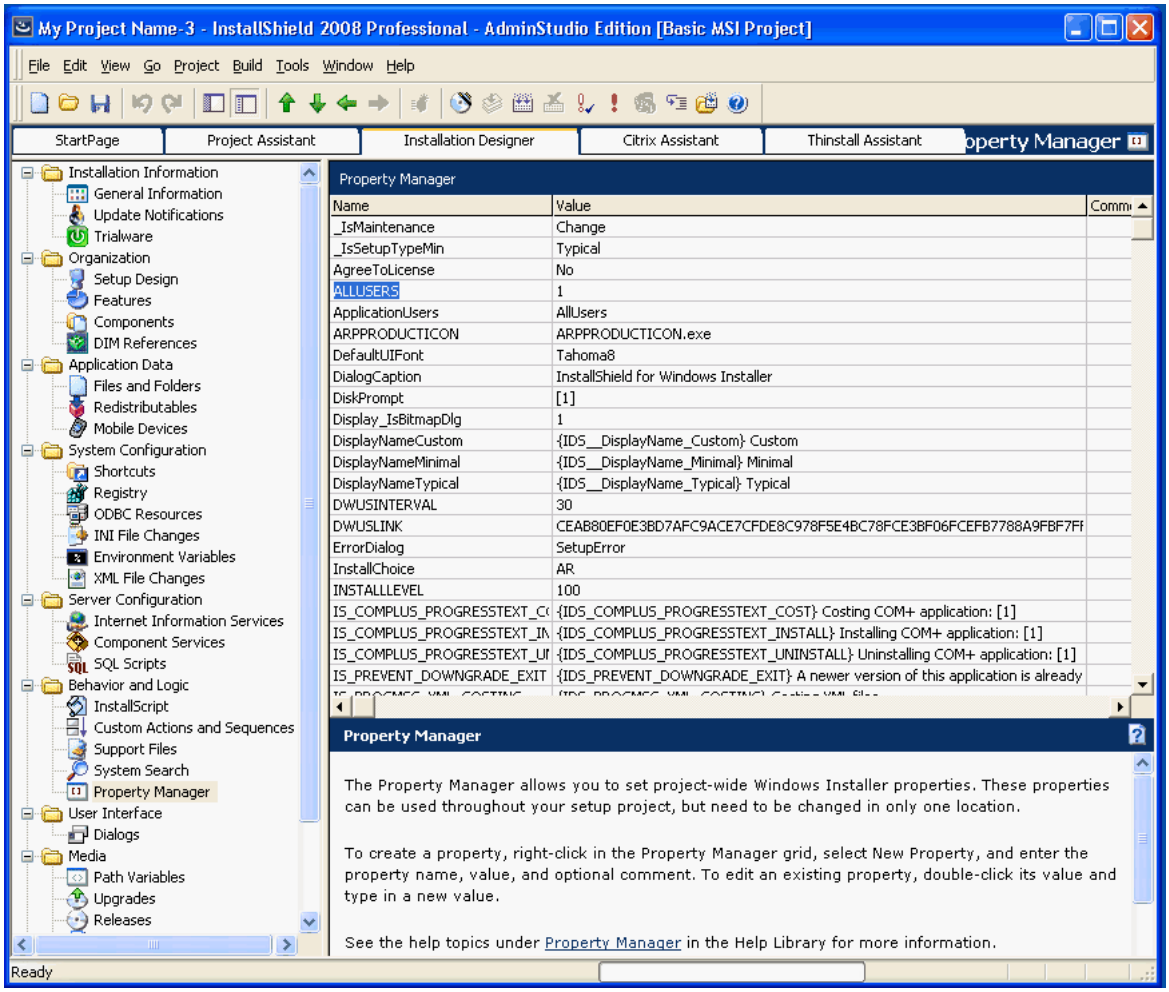
To create an InstallShield Editor template to use within Repackager, perform the following steps.

Task ***To create a customized InstallShield Editor template:***

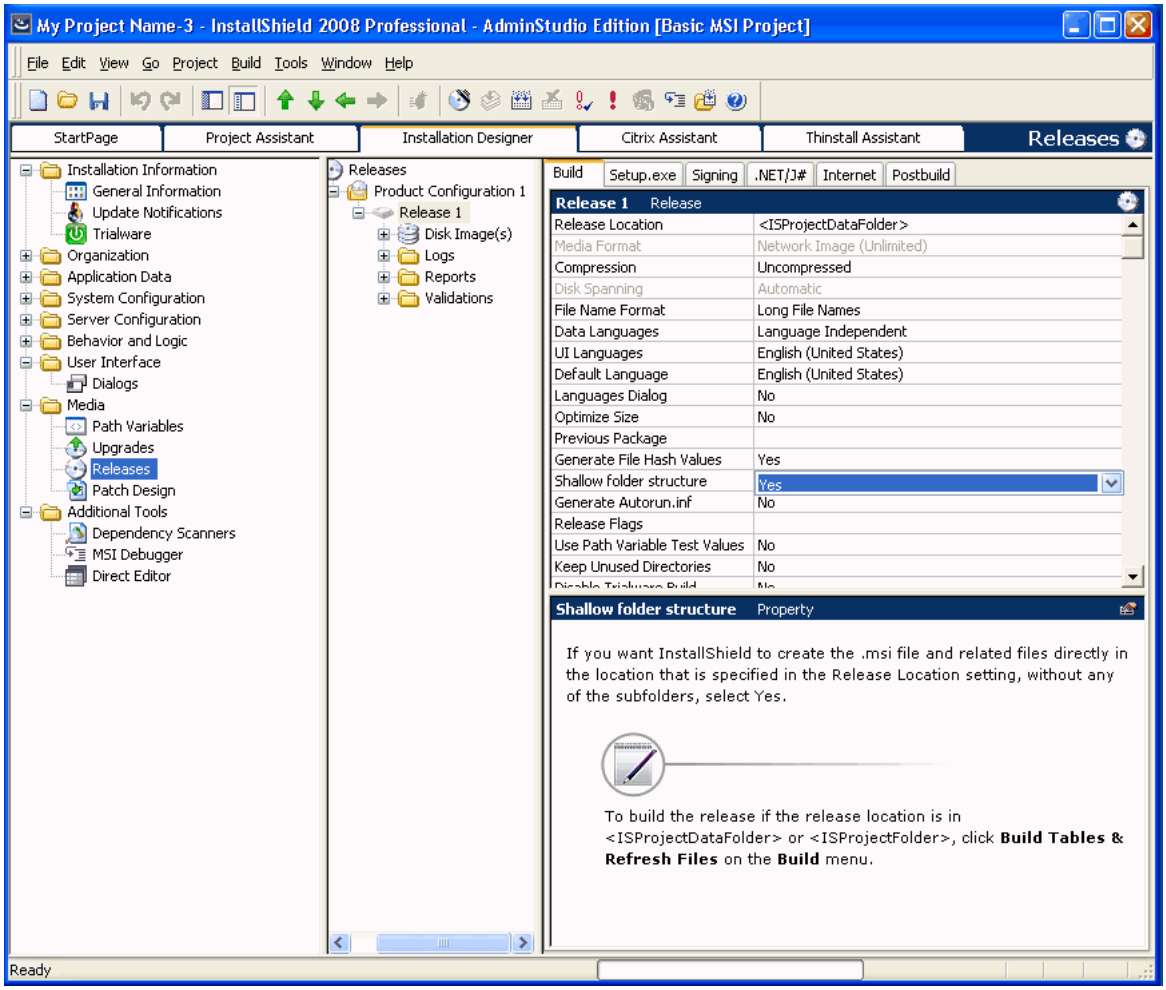
1. Create a new **Basic MSI Project** in the InstallShield Editor.
2. On the **Installation Designer** tab, select the **General Information** node under **Installation Information**, and enter your company-specific information as required.



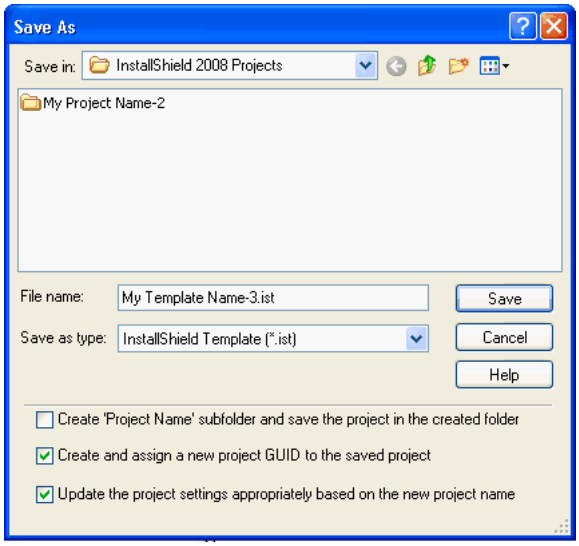
3. Under **Behavior and Logic**, select the **Property Manager** node and add the required properties like ALLUSERS, ISSCRIPTDRIVEN, etc.



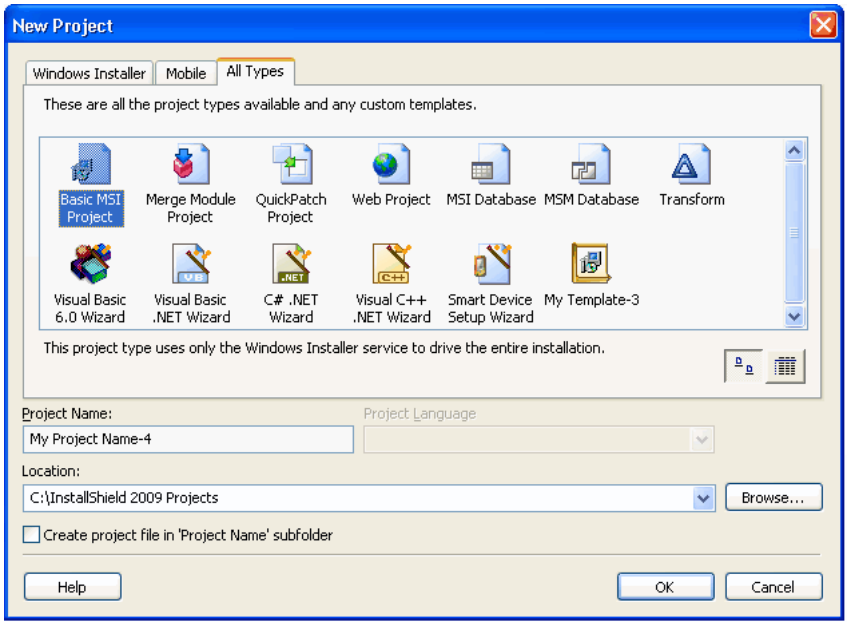
4. You can also optionally set **Shallow Folder Structure** to **Yes** in the **Releases** view under **Media**.



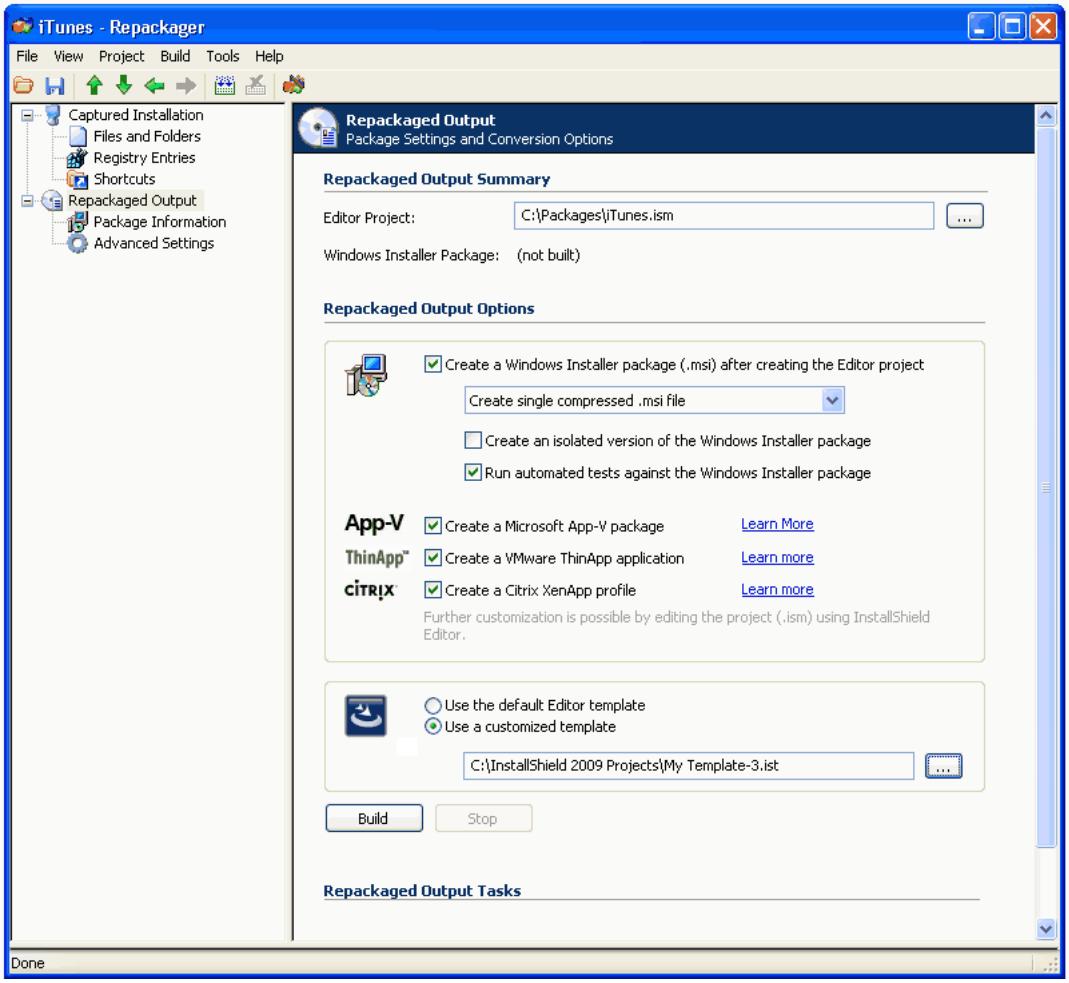
5. After making all required changes, save the project as an InstallShield Editor Template (`.ist`) type.



6. This new template should now be available along with other project types in the InstallShield Editor.



7. From within the Repackager interface, you can start using this customized template by selecting the **Use a customized template** option in the **Repackaged Output** view, and selecting the InstallShield Editor template that you just created.



Repackager Interface Reference

This section describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackager interface. The help topics in this section are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

- Repackager Interface
- Setup Intent Wizard
- VMware Repackaging Wizard
- Exclusions Editor Interface
- Options.ini File
- Files Associated with Repackager
- Using InstallShield to Chain Multiple Windows Installer Packages Together
- Troubleshooting

Repackager Interface

Edition • *The full functionality of the Repackager interface is available in AdminStudio Professional and Enterprise Editions.*

From the Repackager Interface, you can:

- Open the Repackaging Wizard and repackage legacy setups.
- Open the Exclusions Editor and configure exclusions.
- Convert Microfocus ZENworks, Microsoft SMS, and WinINSTALL projects into Repackaging projects.
- Create a package exclusion list.
- Build a Repackager project into an InstallShield Editor project and Windows Installer package.

The Interface consists of several menus, a toolbar, the status bar, the output window, the View List, and several associated views.

- Menus and the toolbar are discussed in the Menus and Toolbar topic.
- Individual views are covered in their respective help topics.
- The status bar, output window, and View List are described in the following table.

Table 9-4 • Repackager Interface Elements

Interface Element	Description
Status Bar	The status bar, which can be toggled from the **View** menu, displays information when you hover over buttons in the toolbar.
View List	The View List allows you to navigate to different views in the Repackager project. The corresponding view is displayed when you select an item in the tree. You can also use the Forward, Back, Navigate Up, and Navigate Down buttons in the View List. The View List includes the following views: • Captured Installation View • Files and Folders View • Registry Entries View • Shortcuts View • INI Files View • Deleted Files View • Deleted Registry Entries View • Repackaged Output View • Package Information View • Software Identification Tag View • Advanced Package Settings View

Table 9-4 • Repackager Interface Elements

Interface Element	Description
Output Window	When you open Repackager 3.x output, Microfocus ZENworks projects, Microsoft SMS projects, WinINSTALL projects, or Wise installation projects in the Repackager Interface, conversion information appears in the Output window. This window can be toggled from the **View** menu.

Repackager Start Page

Edition • *The full functionality of the Repackager interface is available in AdminStudio Professional and Enterprise Editions.*

When you first launch Repackager, the **Repackager Start Page** opens.

This page gives you a brief overview of Repackager functionality and uses, and gives you links to launch the Repackaging Wizard, convert a Windows Installer package to an application virtualization format, open an existing Repackager project, convert a legacy setup to a Repackager project, upgrade a legacy Repackager file, and open a recently accessed package.

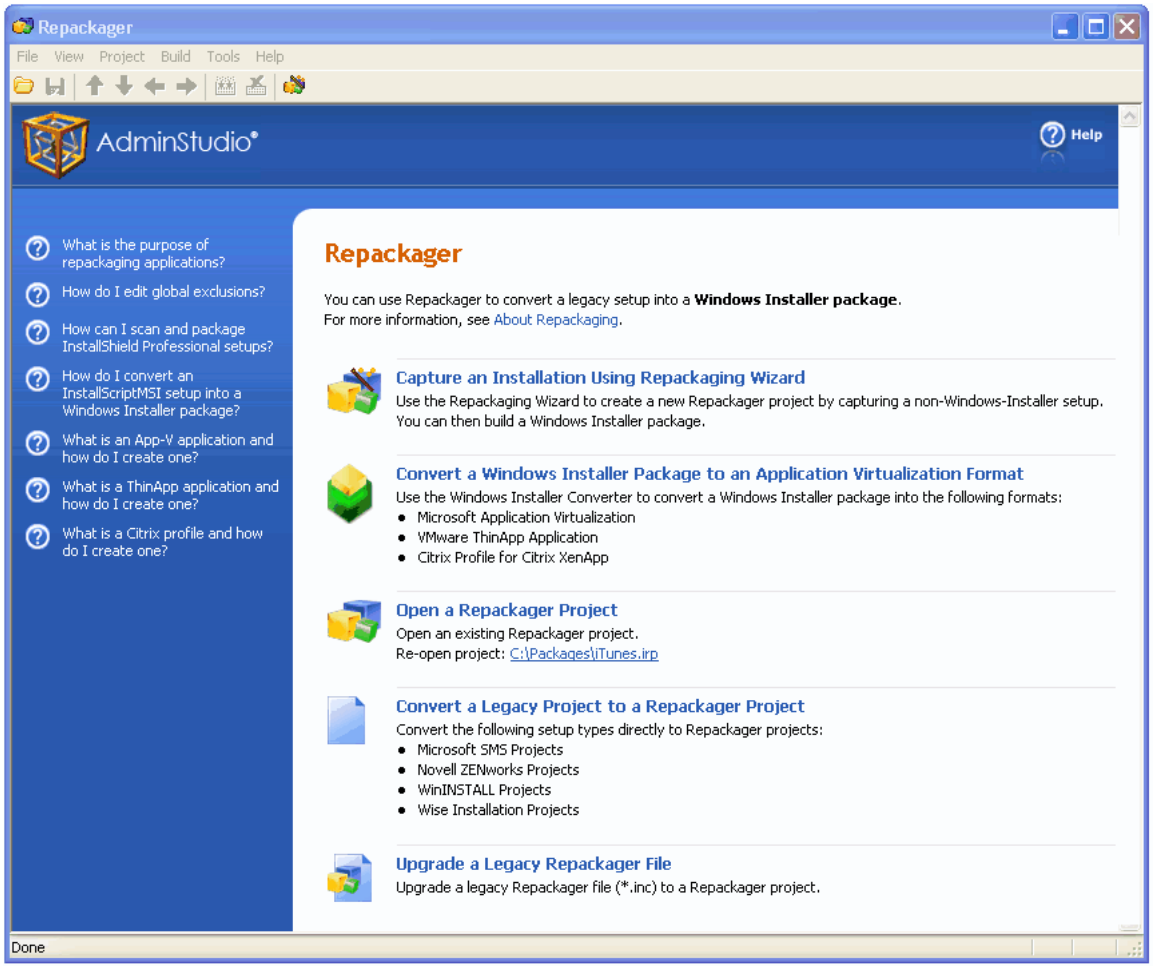


Figure 9-3: Repackager Start Page

Menus and Toolbar

The following table provides a description of each menu command and toolbar button:

Table 9-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
File	**Open**		Ctrl+O	Allows you to open: • An existing Repackager project (`.irp`) • Repackager 3.x output (`.inc`) • Microfocus ZENworks project (`.axt/.aot`) • Microsoft SMS project (`.ipf`) • WinINSTALL converted project (`.txt`) (6.0, 6.5, or 7.x) • Wise Installer project (`.wse`)
File	**Save**		Ctrl+S	Saves the current project.
File	**Save As**			Saves the current project using the name and location you specify.
File	**1,2,3,4**			Allows you to open the four most recently accessed Repackager projects.
File	**Exit**			Exits Repackager.
View	**Toolbar**			Toggles display of the toolbar.
View	**Status Bar**			Toggles display of the status bar.
View	**Output**			Toggles display of the Output window.
View	**Refresh**		F5	Refreshes the current view.
Project	**Edit Windows Installer Package**			Once you build the Repackager project into a Windows Installer package (`.msi`), opens the package in InstallShield Editor (in Direct MSI Edit mode).
Project	**Edit InstallShield Project**			Once you build the Repackager project into an InstallShield Editor project (`.ism`), opens the project in InstallShield Editor.
Project	**Setup Intent Wizard**			Launches the Setup Intent Wizard.

Table 9-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
Project	**Create Report**		Ctrl+R	Allows you to create a report for the project in text or HTML format.
Project	**Properties**			Displays properties for the current project, including exclusion information.
Build	**Build**		F7	Builds the Repackager project into an InstallShield Editor project and a Windows Installer package.
Build	**Stop Build**		Ctrl+Break	Terminates an in-process build.
Tools	**Repackaging Wizard**			Launches the Repackaging Wizard.
Tools	**VMware Repackaging Wizard**			Launches the VMware Repackaging Wizard.
Tools	**Options**			Displays the Options dialog box.
Tools	**Isolation Options**			Displays the Isolation Options dialog box, where you can specify assembly and digital signature isolation options.
Help	**Contents**			Launches the Help Library, displaying the Contents tab.
Help	**Index**			Launches the Help Library, displaying the Index tab.
Help	**Search**			Launches the Help Library, displaying the Search tab.
Help	**Support Central**			Accesses the AdminStudio Support website.
Help	**Web Community**			Accesses the AdminStudio Web Community.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.
Help	**Feedback**			Accesses the feedback form on the AdminStudio website.

Table 9-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
Help	**AdminStudio on the Web**			Accesses the AdminStudio website.
Help	**About Repackager**			Displays the About Repackager dialog box.
	Up			Moves you up one view in the View List.
	Down			Moves you down one view in the View List.
	Back			Displays the previously displayed view in the View List.
	Forward			Returns you to the view from which you selected the Back button.

Dialog Boxes

Repackager includes the following dialog boxes to assist you in your project creation:

- Create Report Dialog Box
- Isolation Options Dialog Box
- Options Dialog Box
- Project Properties Dialog Box
- WinINSTALL Conversion Dialog Box

About Repackager Dialog Box

This dialog box available by selecting **About Repackager** from the **Help** menu, displays version information for Repackager.

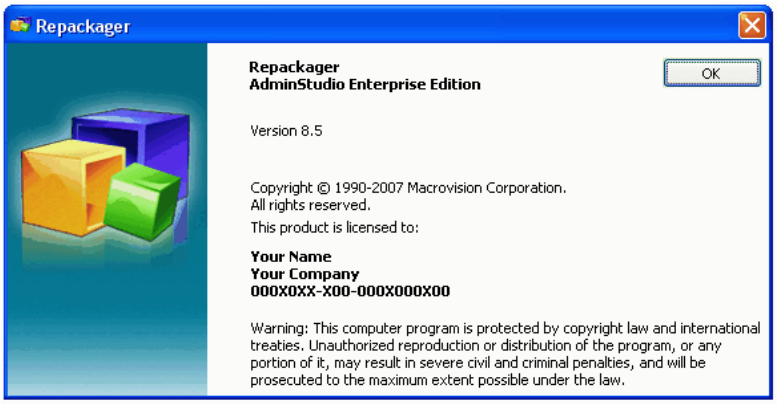


Figure 9-4: About Repackager Dialog Box

Create Report Dialog Box

The Create Report dialog box, available by selecting Create Report from the Project menu, allows you to configure a report for the current Repackager project, or a specific subset of captured data.

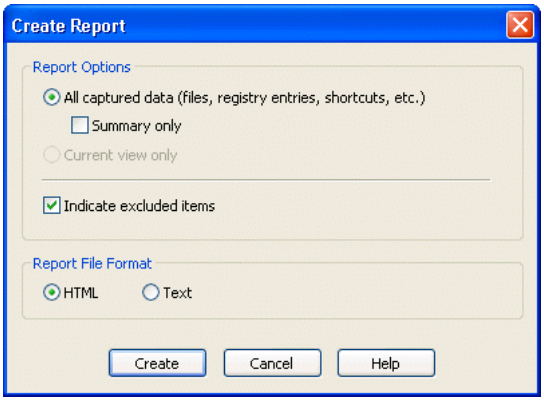


Figure 9-5: Create Report Dialog Box

This dialog box contains the following options:

Table 9-6 • Create Report Dialog Box Options

Option	Description
All captured data (files, registry entries, shortcuts, etc.)	Select to have the report include all captured data.
Summary only	If you select All captured data, you can select this option to only display summary information in the report (the number of items captured and the number of items excluded for files, .ini files, registry data, and shortcuts).
Current view only	Select this option to include only the currently selected view in the report.
Indicate excluded items	Select to display items that have been marked as excluded in Repackager.

Table 9-6 • Create Report Dialog Box Options

Option	Description
Report File Format	Select the file format for Repackager reports: HTML or Text.
Create	When you click Create, you are prompted for a name and location for the outputted report.

Isolation Options Dialog Box

Application isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

On the Isolation Options dialog box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following Repackager isolation options:

- **Assembly Options**—Specify the type of assemblies Repackager will create, and the assembly naming conventions. See Manifest Options Tab
- **Digital Signature Options**—Configure the certificate information required when using shared assemblies. See Digital Signature Tab.

Note • *The modifications you make on the Isolation Options dialog box will be recorded in the* `isolationconfig.ini` `file`*, which is stored in the* `AdminStudio Shared` *directory.*

Manifest Options Tab

The Manifest Options tab allows you to configure several settings associated with manifests. The following settings are included:

Table 9-7 • Isolation Options Dialog Box / Manifest Options Tab

Option	Description
Assembly Type	This option allows you to select the type of assemblies that Repackager will create and use: • Create private side-by-side assemblies in the application folder • Create shared side-by-side assemblies in the WinSxS folder (Default) ***Note*** • *Manifests for shared assemblies must be digitally signed. This can be done in the Digital Signature Tab.* ***Note*** • *A 2048-bit key is required to sign an assembly/manifest being installed to the WinSxS folder.*

Table 9-7 • Isolation Options Dialog Box / Manifest Options Tab (cont.)

Option	Description
Assembly Naming Conventions	Specify your company and division information to define the default naming convention that Repackager will use when creating assemblies during application isolation By default, assembly names are specified in the form of: `Company.Division.Assembly` ***Note*** • *See see About Assemblies and About Manifests for more information.*
Create a new component for each assembly	Select this option if you want to create a new component for each assembly created during isolation. This check box applies to all assemblies created. ***Caution*** • *If you are creating assemblies for applications files within multiple components, this option must be selected for successful application isolation.*

Digital Signature Tab

On the **Digital Signature** tab, you can configure the certificate information required when using shared assemblies.This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

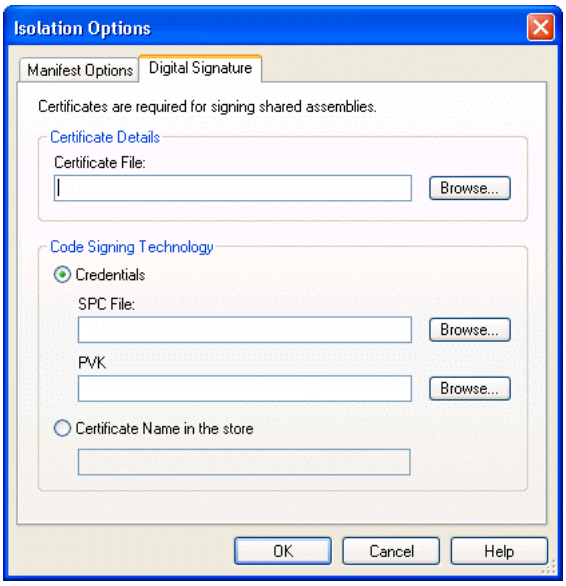


Figure 9-6: Isolation Options Digital Signature Options Tab

Caution • *Repackager uses timestamping when signing global assemblies. Consequently, you must have an Internet connection on the computer when you create a global assembly.*

You must configure the following options when signing these assemblies:

Table 9-8 • Isolation Options Dialog Box / Digital Signatures Tab

Item	Description
Certificate File	Click the Browse (...) button next to the field and navigate to the certificate file you are using to sign assemblies. A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: **SPC File** and **PVK** File. ***Note*** • *In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as VeriSign, with specific information about your company and software.*
SPC File	Specify the name and location of your software publishing credentials file (`.spc`).
PVK	Specify the name and location of your private key file (`.pvk`).
Certificate Name in the Store	Select this option to use the name of an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary.

Note • *For more information, see About Digital Certificates.*

Options Dialog Box

The **Options** dialog box, available from the **Tools** menu, presents options on three tabs: **Colors**, **Merge Modules**, and **Build Options**.

Colors Tab

On the **Colors** tab, you can configure the color of scanned items and deleted items in Repackager's exclusion views (**Files**, **.ini Files**, **Registry Data**, and **Shortcuts**).

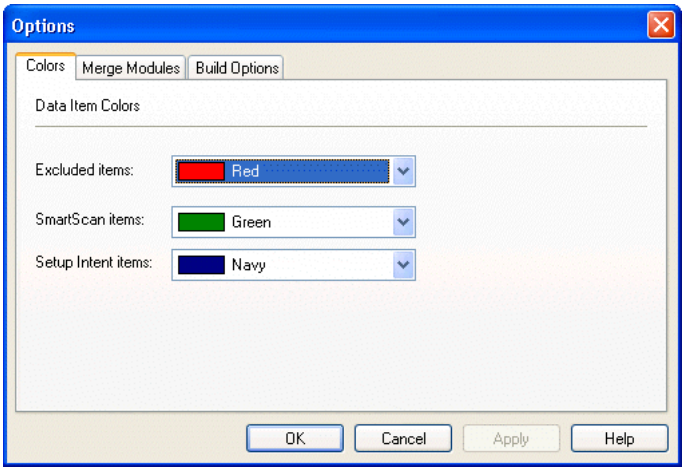


Figure 9-7: Colors Tab of the Options Dialog Box

Merge Modules Tab

On the **Merge Modules** tab, you can specify additional directories containing custom merge modules to use during repackaging.

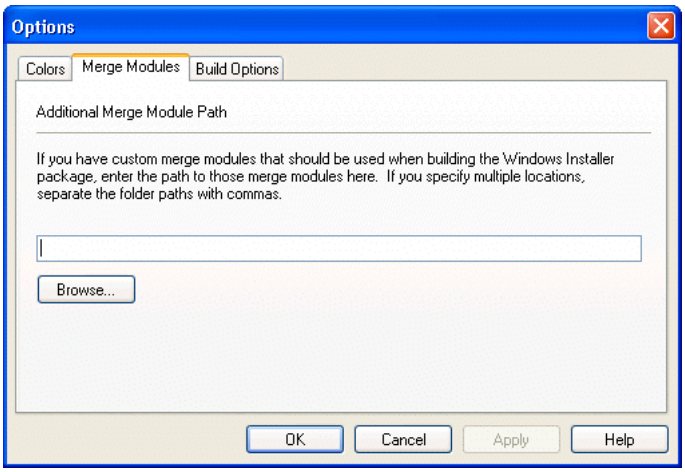


Figure 9-8: Merge Modules Tab of the Options Dialog Box

Build Options Tab

On the **Build Options** tab, you can specify whether or not you want to list ICE validation warnings in the Repackager output window during the Build process and you can set software ID tag file generation options.

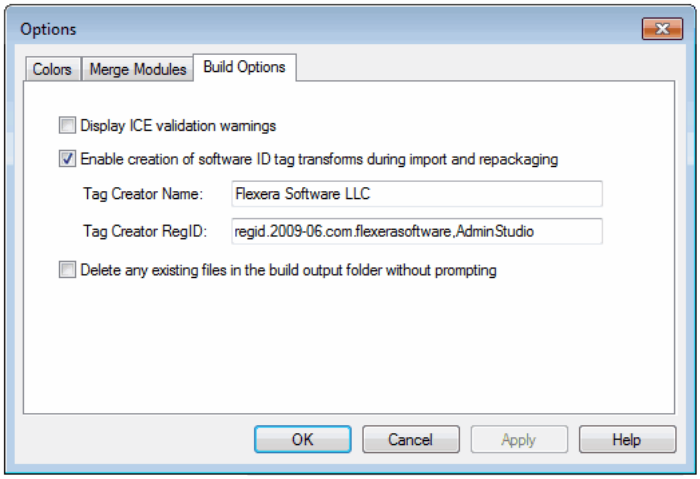


Figure 9-9: Build Options Tab of the Options Dialog Box

The **Build Options** tab includes the following options:

Table 9-9 • Options Dialog Box / Build Options Tab

Option	Description
Display ICE validation Warnings	Select this option to display any ICE validation warnings that occur during the Repackager Build process. By default, this option is not selected.
Enable creation of software ID tag transforms during import and repackaging	Select to instruct AdminStudio to automatically create a transform file containing software tag file(s) for Windows Installer packages that are imported into the Application Catalog or built using Repackager. By default, this option is selected. ***Note*** • *Whenever a Windows Installer package is imported into the Application Catalog or built using Repackager, AdminStudio creates a software ID tag file (which is stored in the Application Catalog), but if the* ***Enable creation of software ID tag transforms during import and repackaging option*** *it not selected, AdminStudio does not create the transform.*
Tag Creator Name	Enter a name to identify the creator of the software ID tag files that will be created by AdminStudio. By default, the value is `Flexera`.

Tag Creator RegID	Enter an ID to uniquely identify the creator of the software ID tag files that will be created by AdminStudio, using the following format:
	
	`regid.YYYY-MM.ReversedDomainName,optional_division`
	
	For example:
	
	`regid.2009-06.com.yourcompany,GlobalProductDivision`
	
	By default, the value is AdminStudio's RegID:
	
	`regid.2009-06.com.flexerasoftware,AdminStudio`

Table 9-9 • Options Dialog Box / Build Options Tab

Option	Description
Delete any existing files in the build output folder without prompting	By default, Repackager will build the Repackager project's associated Windows Installer package in a directory named `MSI_Package`, which is a subdirectory of the directory containing the Repackager project. If you have edited the Repackager project's associated InstallShield Editor project to change this default location, each time you rebuild the Repackager project, Repackager will prompt you to confirm that you want to overwrite the existing files. If you are repeatedly building from the Repackager interface into the same build output folder (which is not the default `MSI_Package` folder) and you do not want to be prompted to confirm that you will be overwriting the existing content, then select this option to suppress the confirmation prompts.

Important • *Any changes that you make to the software tagging options on the* ***Build Options*** *tab of the Repackager* ***Options*** *dialog box will also automatically be made to the options on the* ***General Options > Import Options > Software Tagging*** *tab of the Application Catalog* ***Options*** *dialog box.*

Note • *For more information, see About Software Tagging RegIDs and About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields in the AdminStudio Help Library.*

Project Properties Dialog Box

The Project Properties dialog box, accessed by selecting Properties from the Projects menu, contains two tabs:

Table 9-10 • Project Properties Dialog Box Tabs

Tab	Description
General Tab	Allows you to view properties for the current Repackager project.
Exclusions Tab	Use to configure the location of the default exclusion file.

General Tab

The General tab of the Project Properties dialog box displays information about the current Repackager project (`.irp`).

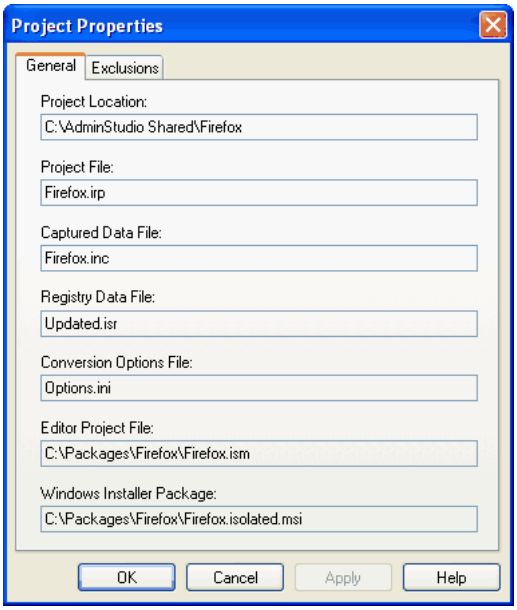


Figure 9-10: Project Properties Dialog Box General Tab

The following options are displayed:

Table 9-11 • General Tab Options

Option	Description
Project Location	The full path of the current Repackager project file (`.irp`).
Project File	The name of the current Repackager project file.
Captured Data File	The name and location of the captured data file (`.inc`), which was either created by the Repackaging Wizard or during conversion of a Microfocus ZENworks project, Microsoft SMS project, or WinINSTALL project. The path is relative to the current Repackager project file.
Registry Data File	The name and location of the file containing captured registry data. The path is relative to the current Repackager project file.
Conversion Options File	The name and location of the Options.ini file, which contains an exhaustive list of all options you can use during conversion of the Repackager project to an InstallShield Editor project and Windows Installer package.
Editor Project File	The name and location of the InstallShield Editor project file as set in the Product view (MSI Package). The path is relative to the current Repackager project file.
Windows Installer Package	The name and location of the Windows Installer package. The path is relative to the current Repackager project file.

Exclusions Tab

The Exclusions tab allows you to select an exclusion file to use as a filter when importing captured data into a Repackager project.

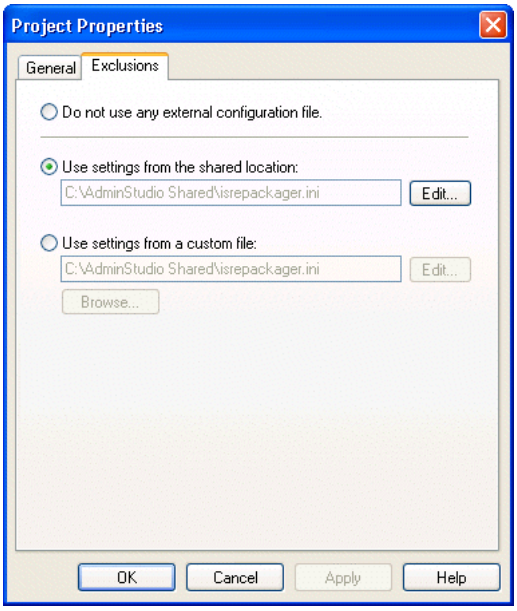


Figure 9-11: Project Properties Dialog Box Exclusions Tab

Select one of the following options for the configuration file:

Table 9-12 • Exclusions Tab Properties

Option	Description
Do not use any external configuration	Repackager will import all captured data into the Repackager project.
Use settings from the shared location	Repackager will use the settings contained in isRepackager.ini in the `AdminStudio Shared` directory (configured during installation). Use this option when you are working in a team environment where the exclusion list needs to be stored in a centralized location.
Use settings from InstallShield defaults	Repackager will use the settings contained in the default.ini file in the Repackager folder. These are the InstallShield Editor-recommended exclusions. It is recommended that you do not modify these exclusions so you can return to them if you need to restart your exclusion list.
Edit	Click to open the Exclusions Editor, which you can use to exclude files, registry entries, .ini files, or shortcuts from the Repackager project. See Configuring Exclusions Using the Exclusions Editor and Exclusions Editor Interface for more information.
Use settings from a custom file	Specify or browse to a file created with the Exclusions Editor that you want to use as your filter during conversion to a Repackager project. You would create a custom exclusion file based upon your company's requirements.

Caution • *Using the custom settings option, it is possible to use the local settings file (`isRepackager.ini`) in the Windows directory. This file is also used for default exclusions for the Repackaging Wizard. By modifying this file, you introduce the possibility of excluding data at repackaging time in subsequent Repackaging Wizard executions, as opposed to marking items as excluded in a Repackager project (which does not affect the captured data). For this reason, it is highly recommended that you do not use the `isRepackager.ini` configuration file in the local Windows folder for your Repackager exclusions.*

WinINSTALL Conversion Dialog Box

When you convert a WinINSTALL project to a Repackager project, this dialog box appears to allow you to set WinINSTALL-specific variables. These variables are:

Table 9-13 • WinINSTALL Variables

Variable	Description
@Server	The machine name of the server where the WinINSTALL directory is located.
@WinstallDir	The location of the directory where the WinINSTALL executables are located.

Repackager Views

Repackager includes several views, from which you can examine the captured data that will be used to create an InstallShield Editor project (`.ism`) and Windows Installer package (`.msi`). Depending on the presence or absence of certain data types, some views may not be displayed. For example, if the setup does not include any `.ini` files, the INI Files view will not be displayed in the View List.

The following views are available in Repackager:

- Captured Installation View
- Files and Folders View
- Registry Entries View
- Shortcuts View
- INI Files View
- Deleted Files View
- Deleted Registry Entries View
- Repackaged Output View
- Package Information View
- Software Identification Tag View
- Advanced Package Settings View

Note • *Information listed in the views (such as files, .ini files, or registry entries) is limited to 267 characters in length. Anything longer than this limit will be truncated in the view. The full value can be viewed in InstallShield Editor.*

Captured Installation View

From the **Captured Installation** view, you can review summary information about the setup you are converting into a Windows Installer package.

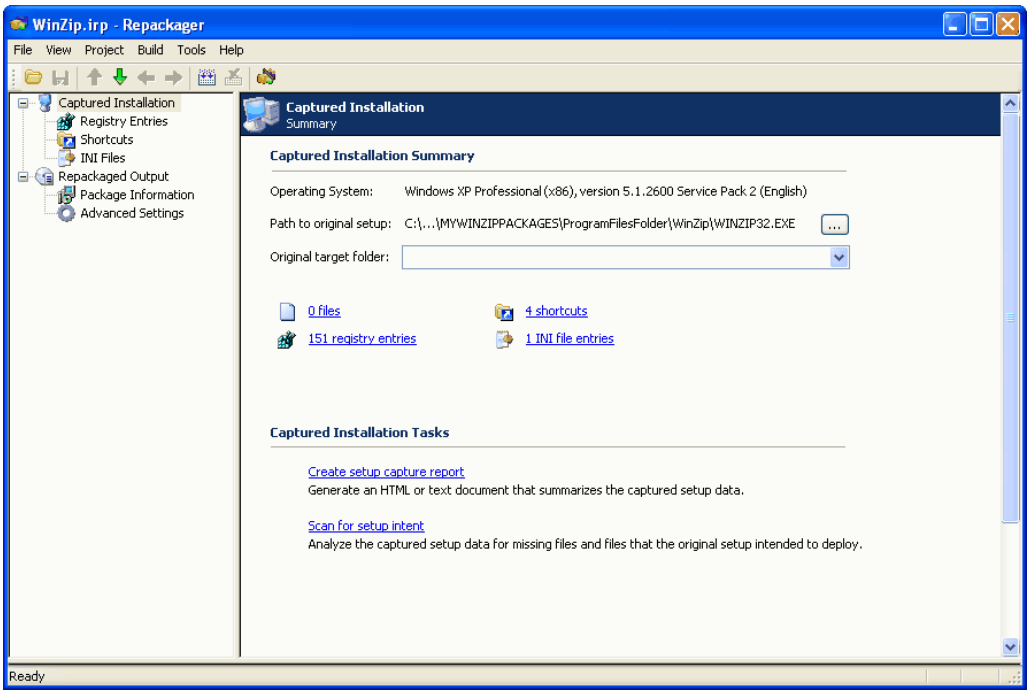


Figure 9-12: Repackager Captured Installation View

The **Captured Installation** view lists the following information:

Table 9-14 • Repackager Captured Installation View

Property	Description
Operating System	Identifies the operating system—including version, service pack, and processor type (32 or 64-bit)—of the machine where the capture was performed.
Path to original setup	Location of setup that was repackaged. Click the browse (...) button to select a different setup.

Table 9-14 • Repackager Captured Installation View

Property	Description
Original target folder	From this list, select the original target folder for the installation. In most cases, this will be a subdirectory of `[ProgramFilesFolder]`. Alternatively, you can enter your own target. This value will be set as the value for `INSTALLDIR`, and is a mandatory property. ***Note*** • *Information about the provided install locations can be found in the SystemFolder Property topic of the Windows Installer Help Library.*
Number of Files, Shortcuts, Registry Entries, and INI File Entries	Links that list the number of files, shortcuts, and registry entries captured, and the number of `.ini` file changes made. Click these links to open the following subviews: • Files and Folders View • Registry Entries View • Shortcuts View • INI Files View Each subview of this view allows you to view the names and associated information of each item captured, and selectively exclude (or reinclude) these items from the ultimate Windows Installer package. If no entries were captured of a particular type, the corresponding view does not appear in the View List. For example, if no `.ini` file changes were captured, the **INI Files** view is not displayed.
Create setup capture report	Click to generate the **Setup Capture Report**, an HTML or text document that summarizes the data that was captured when a setup was repackaged. For more information, see Creating a Setup Capture Report for a Project. ***Edition*** • *The Setup Capture Report feature is included with AdminStudio Professional and Enterprise Editions.*
Scan for setup intent	Click to launch the **Setup Intent Wizard**, which you can use to scan a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files. For more information, see Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project. ***Edition*** • *The **Setup Intent Wizard** is included with AdminStudio Professional and Enterprise Editions.*

Files and Folders View

From the **Files and Folders** view, you can examine information about each captured file, selectively exclude files or directories from the package you are creating, or reinclude files that you previously excluded. You can also add files or directories to the global exclusion list, or remove them from the list.

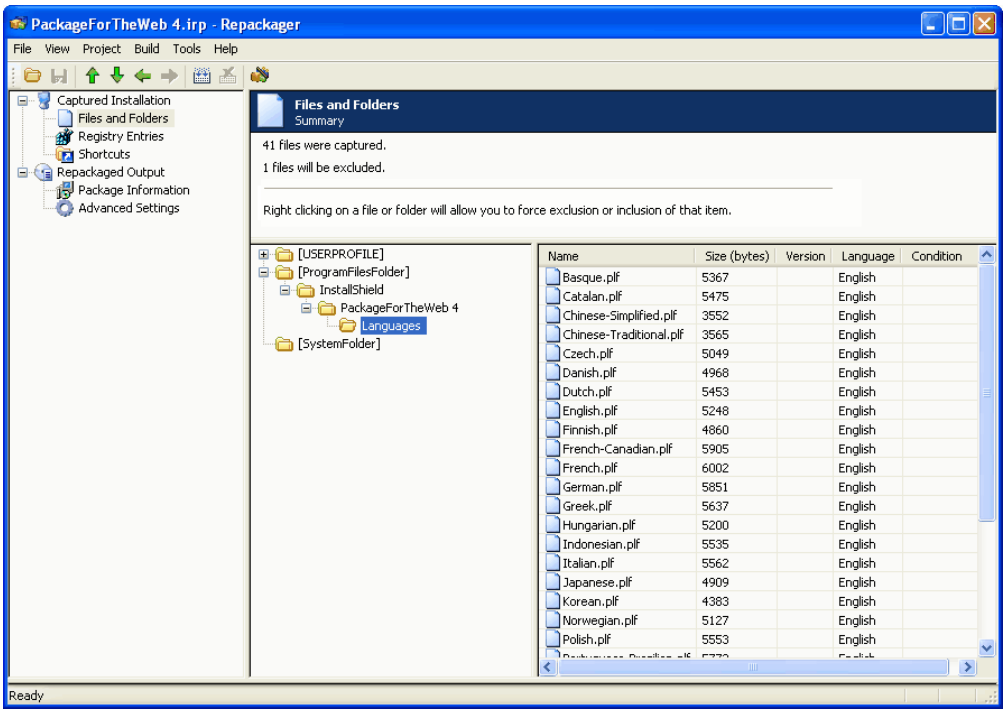


Figure 9-13: Repackager Files and Folders View

The upper pane displays the number of files captured and how many of these files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where files will be installed and the names of the files.

When you select a file from the tree, the lower-right pane displays attributes for that file. These attributes are:

Table 9-15 • File Attributes

Attribute	Description
Name	The file's name.
Size	The file's size in bytes.
Version	The file's version.
Short Name	The short name for the file (if the file's author defined it).
Language	The file's language.

Excluding Files and Subdirectories

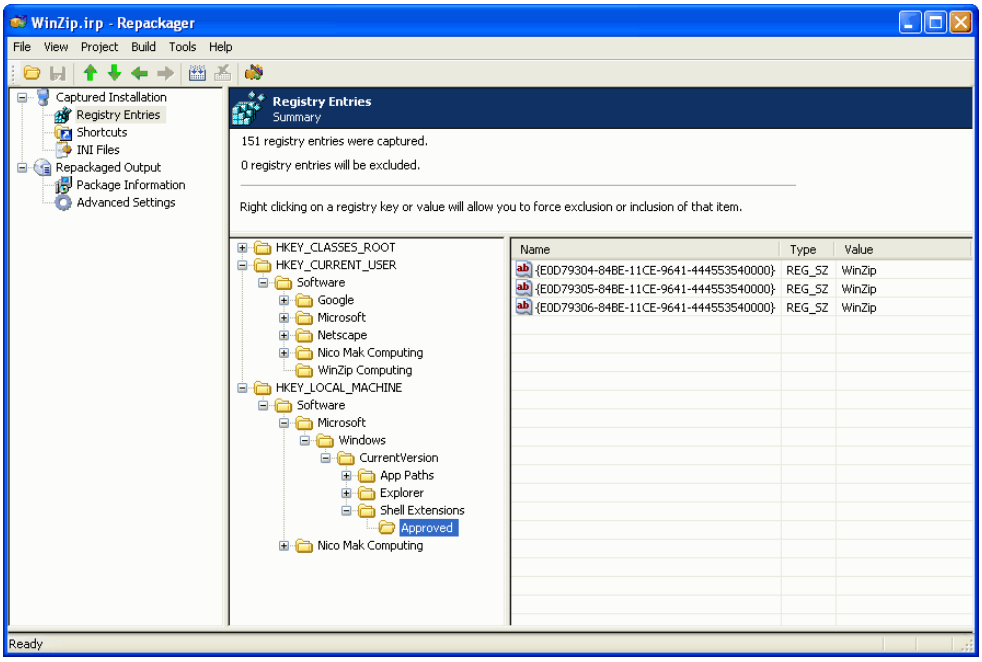
To specify which files and subdirectories you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, **Include All**, **Add to Exclusions**, and **Remove from Exclusions** commands on the shortcut menu:

Table 9-16 • Excluding Files and Subdirectories

Category	To ...	Right-click on ...	Select on shortcut menu ...
Project Exclusions	**Exclude a captured file**	File you want to exclude	**Exclude**
	Exclude captured files within a directory	Directory containing the files you want to exclude	• **Exclude** (to exclude only the files in the selected directory) or • **Exclude All** (to exclude all of the files in the selected directory and all of its subdirectories)
	Include a captured file that had previously been excluded	File you want to include	**Include**
	Include captured files within a directory that had previously been excluded	Directory containing the files you want to include	• **Include** (to include only the files in the selected directory) or • **Include All** (to include all of the files in the selected directory and all of its subdirectories).
Global Exclusion List	**Add a captured file to the global exclusions list**	File you want to add to the global exclusions list	**Add to Exclusions**
	Add captured files within a directory to the global exclusions list	Directory containing the files you want to add to the global exclusion list	**Add to Exclusions** You will be prompted to indicate whether you want to also exclude files in subdirectories of the selected directory.
	Remove a captured file that had previously been added to the global exclusions list	File you want to remove	**Remove from Exclusions**
	Remove captured files within a directory that had previously been added to the global exclusions list	Directory containing the files you want to remove from the global exclusion list	**Remove from Exclusions**

Registry Entries View

From the **Registry Entries** view, you can examine information about each captured registry entry, selectively exclude registry values or registry keys from the package you are creating, or reinclude registry values that you previously excluded.



The upper pane displays the number of registry entries captured and how many of these entries will be excluded from the Windows Installer package when built. The lower-left pane provides a tree displaying the registry keys and subkeys captured. When you select a key from the tree, the lower-right pane displays any registry values for that key. Displayed information includes:

Table 9-17 • Registry Attributes

Attribute	Description
Name	The registry value name.
Type	The registry value type. This can be either a string value, an expandable string value, a multistring value, a dword value, or a binary value.
Value	The content of the registry value.

Excluding Registry Entries

To specify which registry entries you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the shortcut menu:

- **To exclude a registry entry**, select the registry entry you want to exclude and select **Exclude**.
- **To exclude registry entries within a registry key or registry hive**, select the key or hive from the tree and select either **Exclude** (to exclude the registry entries in the selected hive or key only) or **Exclude All** (to exclude all of the registry entries in the selected hive or key and all of its keys and subkeys).

- **To include a registry entry that had previously been excluded**, select the registry entry and select **Include**.
- **To include registry entries within a registry key or registry hive that had previously been excluded**, select the key or hive from the tree and select either **Include** (to include the registry entries in the selected hive or key only) or **Include All** (to include all of the registry entries in the selected hive or key and all of its keys and subkeys).

Shortcuts View

From the Shortcuts view, you can examine information about each captured shortcut, selectively exclude shortcuts from the package you are creating, or reinclude shortcuts that you previously excluded.

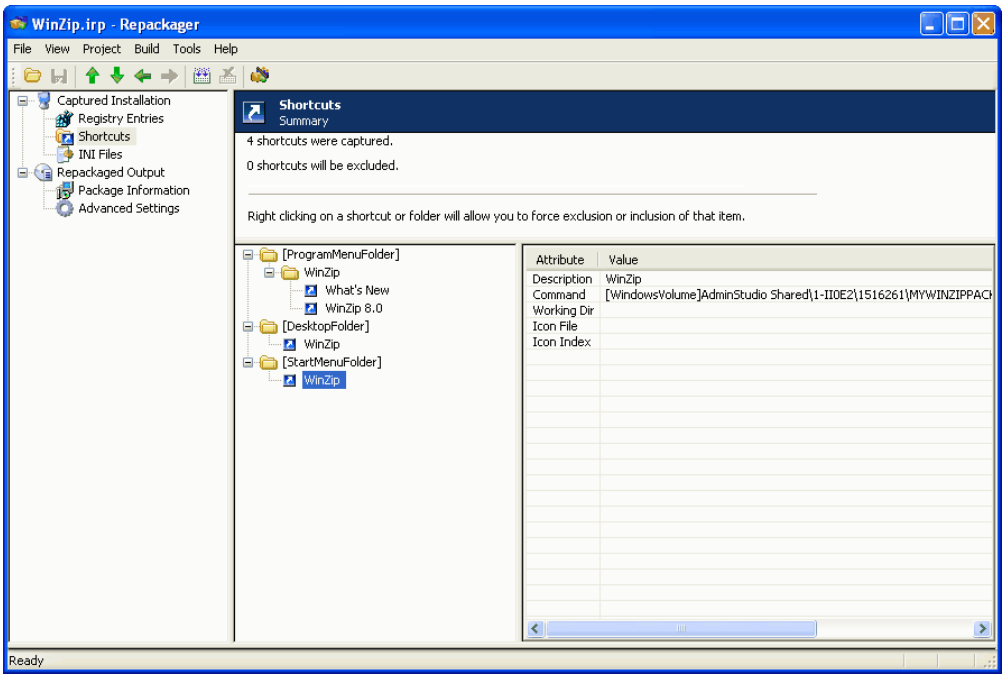


Figure 9-14: Repackager Shortcuts View

The upper pane displays the number of shortcuts captured and how many of these shortcuts will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where shortcuts will be installed and the names of the shortcuts. When you select a shortcut from the tree, the lower-right pane displays attributes for that shortcut. These attributes are:

Table 9-18 • Shortcuts View Attributes

Attribute	Description
Description	The name of the shortcut as it appears on the desktop.
Command	The fully-qualified path and name of the file to which the shortcut points.
Working Dir	The shortcut's working directory, which may need to be specified so required files can load. This is equivalent to the **Start in** value found when right-clicking a shortcut from the desktop and selecting **Properties**.

Table 9-18 • Shortcuts View Attributes

Attribute	Description
Icon File	The name of the file containing the shortcut's icon.
Icon Index	The index number for the icon in the icon file.

Note *• Shortcuts can be excluded from the Windows Installer package you are building on an individual shortcut basis or by directory.*

Excluding Shortcuts

To specify which shortcuts you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the shortcut menu:

- **To exclude a shortcut**, select the shortcut you want to exclude and select **Exclude**.
- **To exclude shortcuts within a directory**, select the directory containing the shortcuts you want to exclude and select either **Exclude** (to exclude only the shortcuts in the selected directory) or **Exclude All** (to exclude all of the shortcuts in the selected directory and all of its subdirectories).
- **To include a shortcut that had previously been excluded**, select the shortcut you want to include and select **Include**.
- **To include shortcuts within a directory that had previously been excluded**, select the directory containing the shortcuts you want to include and select either **Include** (to include only the shortcut in the selected directory) or **Include All** (to include all of the shortcuts in the selected directory and all of its subdirectories).

INI Files View

From the INI Files view, you can examine information about each captured `.ini` file, selectively exclude `.ini` files or `.ini` file sections from the package you are creating, or reinclude `.ini` files or sections that you previously excluded.

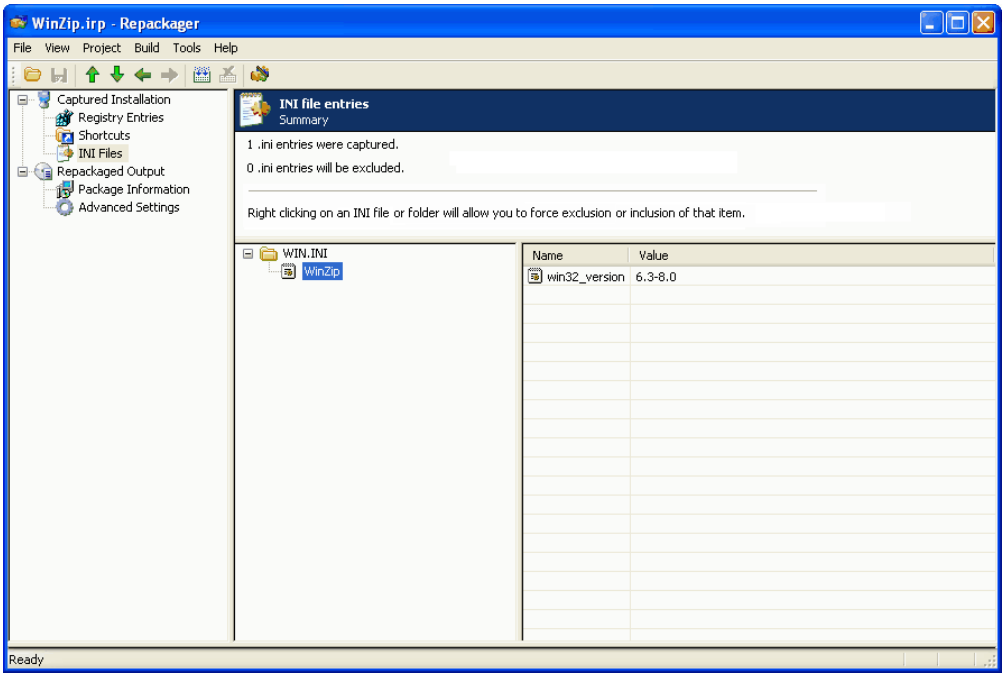


Figure 9-15: Repackager INI Files View

The upper pane displays the number of `.ini` files captured and how many of these `.ini` files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see the full path to captured `.ini` files and sections contained within the `.ini` files. When you select a section from the tree, the lower-right pane displays name/value pairs in that section.

Excluding INI Files

To specify which INI files you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the shortcut menu:

- **To exclude an INI file**, select the INI file you want to exclude and select **Exclude**.
- **To exclude INI files within a directory**, select the directory containing the INI files you want to exclude and select either **Exclude** (to exclude only the INI files in the selected directory) or **Exclude All** (to exclude all of the INI files in the selected directory and all of its subdirectories).
- **To include an INI file that had previously been excluded**, select the INI file you want to include and select **Include**.
- **To include INI files within a directory that had previously been excluded**, select the directory containing the INI files you want to include and select either **Include** (to include only the INI file in the selected directory) or **Include All** (to include all of the INI files in the selected directory and all of its subdirectories).

Deleted Files View

From the **Deleted Files** view, you can examine information about each file deleted during repackaging, selectively exclude files or directories from the package you are creating, or reinclude previously excluded files.

The **Deleted Files** view is populated if you select the **Deleted files** option on the **Analysis Options** dialog box of the Repackaging Wizard

***Note** • The **Analysis Options** dialog box is opened by clicking **Edit** on the **Set Target Project Information and Capture Settings** panel of the Repackaging Wizard.*

The upper pane displays the number of files captured and how many of these files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where files will be installed and the names of the files. When you select a file from the tree, the lower-right pane displays attributes for that file. These attributes are:

Table 9-19 • Deleted Files View Attributes

Attribute	Description
Name	The file's name.
Size	The file's size in bytes.
Version	The file's version.
Short Name	The short name for the file (if the file's author defined it).
Language	The file's language.

Excluding Files and Subdirectories

To specify which files and subdirectories you want to include in the package, use the Exclude, Exclude All, Include, and Include All buttons:

- To exclude a captured file from the package, select the file you want to exclude and click Exclude.
- To exclude all captured files and subdirectories within a directory from the package, select the directory containing the files and subdirectories you want to exclude and click Exclude All.
- To include a captured file in the package that had previously been excluded, select the file you want to include and click Include.
- To include all captured files and subdirectories within a directory, select the directory containing the files and subdirectories you want to include and click Include All.

Deleted Registry Entries View

From the **Deleted Registry Entries** view, you can examine information deleted from the registry repackaging, selectively exclude registry keys from the package you are creating, or reinclude previously excluded data.

The **Deleted Registry Entries** view is populated if you select the **Deleted registry entries** option on the **Analysis Options** dialog box of the Repackaging Wizard

***Note** • The **Analysis Options** dialog box is opened by clicking **Edit** on the **Set Target Project Information and Capture Settings** panel of the Repackaging Wizard.*

The upper pane displays the number of deleted registry entries captured and how many of these entries will be excluded from the Windows Installer package when built. The lower-left pane provides a tree displaying the registry keys and subkeys captured. When you select a key from the tree, the lower-right pane displays any registry values for that key. Displayed information includes:

Table 9-20 • Deleted Registry Entries View Attributes

Attribute	Description
Name	The registry value name.
Type	The registry value type. This can be either a string value, an expandable string value, a multistring value, a dword value, or a binary value.
Value	The content of the registry value.

Excluding Registry Entries

To specify which registry entries you want to include in the package, use the Exclude, Exclude All, Include, and Include All buttons:

- To exclude a registry entry from the package, select the registry entry you want to exclude and click Exclude.
- To exclude all registry entries and subdirectories within a directory from the package, select the directory containing the registry entries you want to exclude and click Exclude All.
- To include a registry entry in the package that had previously been excluded, select the registry entry you want to include and click Include.
- To include all registry entries and subdirectories within a directory, select the directory containing the shortcuts and subdirectories you want to include and click Include All.

Repackaged Output View

From this view, you can configure build options for the project, including whether to build an MSI package automatically following conversion.

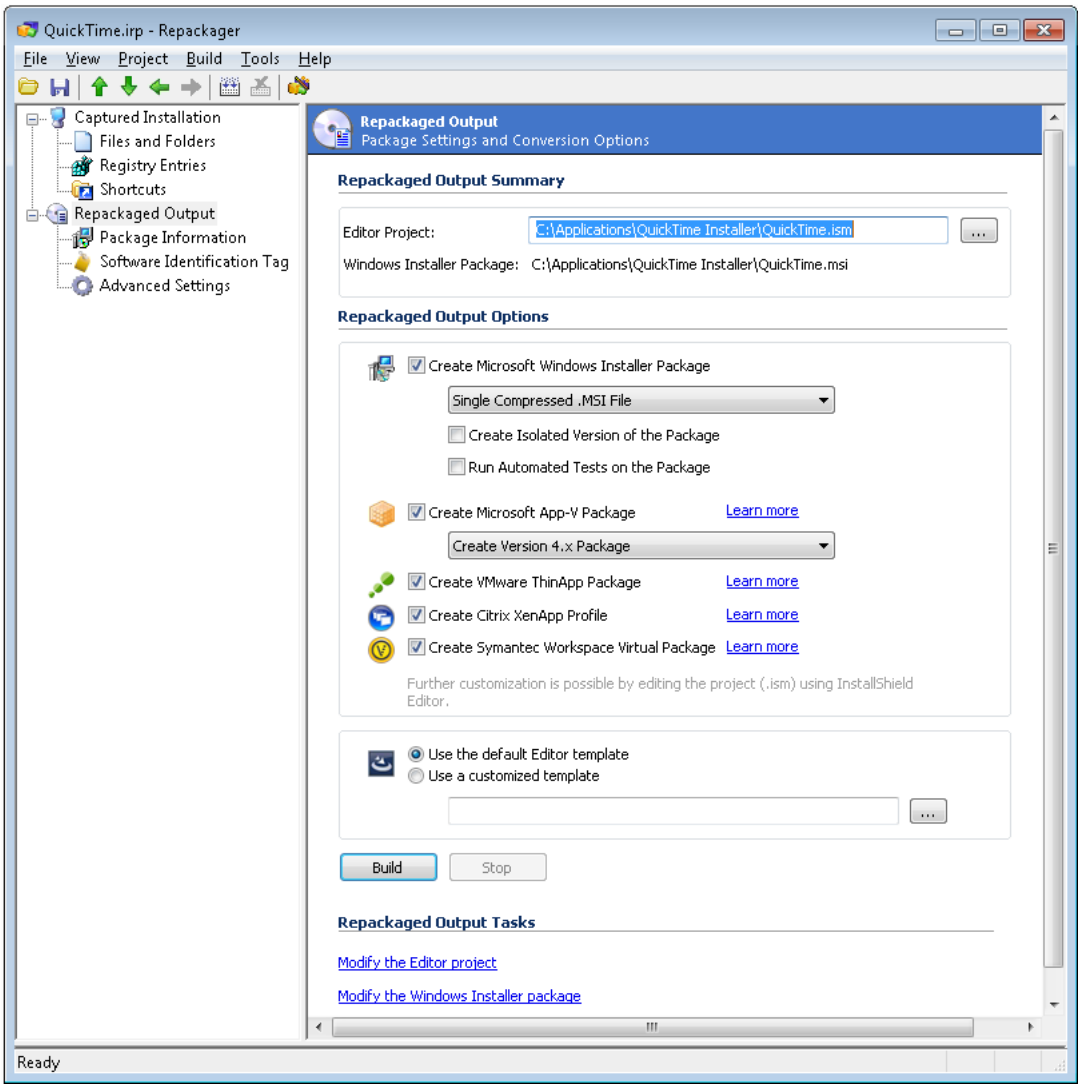


Figure 9-16: Repackager Repackaged Output View

The following properties are available for configuration:

Table 9-21 • Repackaged Output View Options

Option	Description
Editor Project	Provide the name and location of the InstallShield Editor project (`.ism`) file.

Table 9-21 • Repackaged Output View Options

Option	Description
Windows Installer Package	The name and location of the Windows Installer package (`.msi`). By default, Repackager creates the Windows Installer package in a subdirectory, named `MSI_Package`, of the directory containing the Repackager project file. To change this default location, you need to edit this Repackager project's associated InstallShield Editor project file. ***Note*** • *If a Windows Installer package has not yet been built from this Repackager project,* `(not built)` *is listed.*
Create Microsoft Windows Installer Package	If this option is selected, after creating the InstallShield Editor project file (`.ism`), a Windows Installer (`.msi`) file will also be built.
Windows Installer Package Options	If you have selected the **Create Microsoft Windows Installer Package** option, you need to also select one of the following options: • **Single Compressed .MSI File**—Select this option if you want to compress all necessary files inside the .msi package, as opposed to storing them outside of the .msi database. • **Single Compressed Setup.exe File**—Select this option if you want to compress all files inside a setup.exe file, including the .msi file and all other necessary files. • **.MSI File With External .CAB File**—Select this option if you want to create an .msi file and want to compress the rest of the necessary files in an external .cab file. • **.MSI File With External .CAB File and Setup.exe**—Select this option if you want to create an .msi file and a setup.exe file, and want to compress all the rest of the necessary files in an external .cab file. • **Uncompressed .MSI File**—Select this option if you want to create an uncompressed .msi file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi file. • **Uncompressed .MSI File With Setup.exe**—Select this option if you want to create an uncompressed .msi file along with a setup.exe file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi and setup.exe files.

Table 9-21 • Repackaged Output View Options

Option	Description
Create Isolated Version of the Package	Select this option to create a second, isolated version of the Windows Installer package when the Windows Installer package is built. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. If this option is selected, an additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of: *appname*`.isolated.msi` For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages. ***Note*** *• This option is only enabled when the* ***Create Microsoft Windows Installer Package*** *option is selected and one of the following values is chosen:* • *Single Compressed .MSI File* • *.MSI File With External .CAB File* • *Uncompressed .MSI File*
Run Automated Tests on the Package	Select this option to automatically run best practice tests against the newly built Windows Installer package to determine if it is built according to Windows Installer standards, and if it is in compliance with the installation requirements of the Windows operating system. ***Note*** *• This option is only enabled when the* ***Create Microsoft Windows Installer Package*** *option is selected and any of the values except for* ***Single Compressed Setup.exe File*** *is chosen.*
Create Microsoft App-V Package	If this option is selected, after building a Windows Installer (.msi) file, a Microsoft App-V application will also be built. ***Note*** *• This option requires that you build a Windows Installer package.*
Create VMware ThinApp Package	If this option is selected, after building a Windows Installer (.msi) file, a VMware ThinApp application will also be built. ***Note*** *• This option requires that you build a Windows Installer package.*

Table 9-21 • Repackaged Output View Options

Option	Description
Create Citrix XenApp Profile	If this option is selected, after building a Windows Installer (.msi) file, a Citrix profile compatible with Citrix XenApp will also be built. ***Note*** • *This option requires that you build a Windows Installer package.*
Use the default Editor template	When building an InstallShield Editor project, select this option to use the default InstallShield Editor template. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project.
Use a customized template	When building an InstallShield Editor project, select this option to specify a customized InstallShield Editor Project Template to use. For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.
Build	Click to initiate the build process to build a Windows Installer package.
Repackaged Output Tasks	After an InstallShield Editor project and a Windows Installer package has been built, you can use these links to perform the following tasks: • **Modify the Editor project**—Open this Repackager project's associated InstallShield Editor project in InstallShield Editor. • **Modify the Windows Installer package**—Open this Repackager project's associated Windows Installer package in InstallShield Editor.

Once you have built the Windows Installer package and/or InstallShield Editor file, you can launch InstallShield Editor from the **Repackaged Output** area of the view.

Package Information View

The Package Information view allows you to specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.

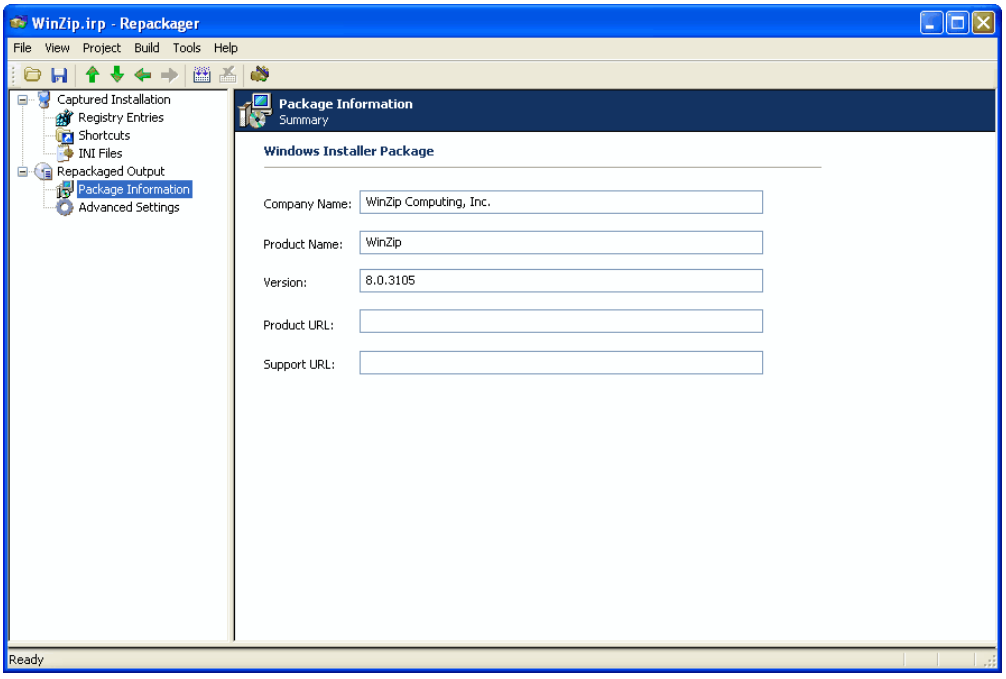


Figure 9-17: Repackager Package Information View

You can configure the following options:

Table 9-22 • Package Information View Options

Option	Description
Company Name	The name of the company that developed the product you are repackaging.
Product Name	The name of the product you are repackaging.
Version	The product's version number.
Product URL	The URL for product information. This appears in Add/Remove Programs in the Control Panel.
Support URL	A URL for support information. This also appears in Add/Remove Programs in the Control Panel, and is often changed during repackaging to provide an internal support URL.

Software Identification Tag View

When you use Repackager to convert a legacy package to a Windows Installer package, by default a tag file is generated for each package when the Windows Installer package is built. You can view and edit tag information in the Repackager interface's **Software Identification Tag** view.

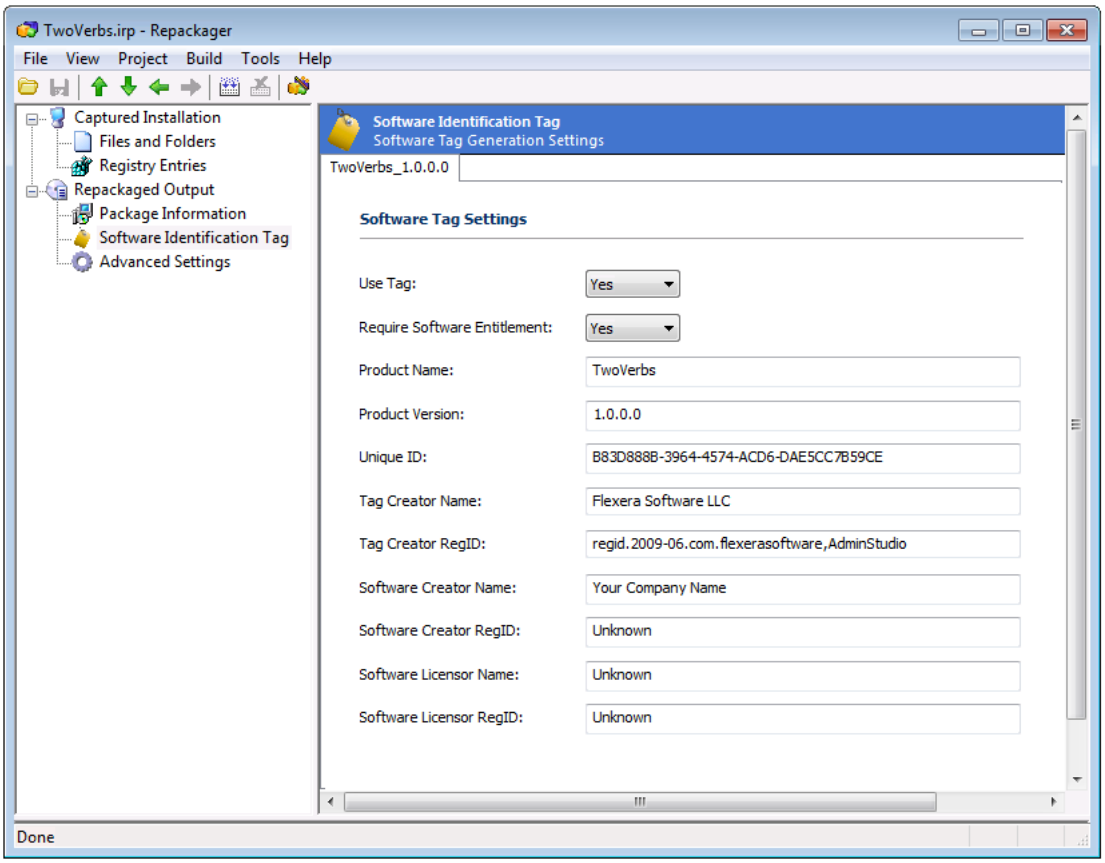


Figure 9-18: Software Identification Tag View

The **Software Identification Tag** tab includes the following properties:

Table 9-23 • Software Identification Tag View Properties

Property	Description
Use Tag	Select **Yes** to enable software ID tag file generation for this package or select **No** to disable it. Using this option enables you to override the setting on the **Build Options** tab of the Repackager **Options** dialog box for this project. The default value is **Yes**.
Require Software Entitlement	To specify that you want to require your product to have a corresponding software entitlement in order for software reconciliation to be considered successful, set this property to **Yes**. In general, if the software must be purchased, this property should be set to **Yes**; if the software is free, this property should be set to **No**.
Product Name	Name of the product, read from the Product Name property of the Windows Installer package.
Product Version	Version of the product, read from the Product Version property of the Windows Installer package.

Table 9-23 • Software Identification Tag View Properties

Property	Description
Unique ID	The product GUID, which is the ProductCode of the MSI package or the unique string used for the Add and Remove Programs uninstall key name, is used to uniquely identify the product in the software identification tag file.
Tag Creator Name	Enter a name to identify the creator of this tag file. The default value is: `Flexera`

Tag Creator RegID	Enter a RegID to identify the creator of this tag file, using the following format:
	`regid.YYYY-MM.ReversedDomainName,optional_division`
	For example:
	`regid.2009-06.com.yourcompany,GlobalProductDivision`

Software Creator Name	(Optional) Enter a name to identify the creator of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.
Software Creator RegID	(Optional) Enter a RegID to identify the creator of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.
Software Licensor Name	(Optional) Enter a name to identify the licensor of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.
Software Licensor RegID	(Optional) Enter a RegID to identify the licensor of this package. By default, the value is **Unknown**. If the value of this field is left as **Unknown**, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.

Note *• For more information, see About Software Tagging RegIDs and About the Tag Creator Name, Software Creator Name, and Software Licensor Name Fields in the AdminStudio Help Library.*

Advanced Package Settings View

From the **Advanced Package Settings** view, you can configure several additional settings that may apply to your repackaged setup.

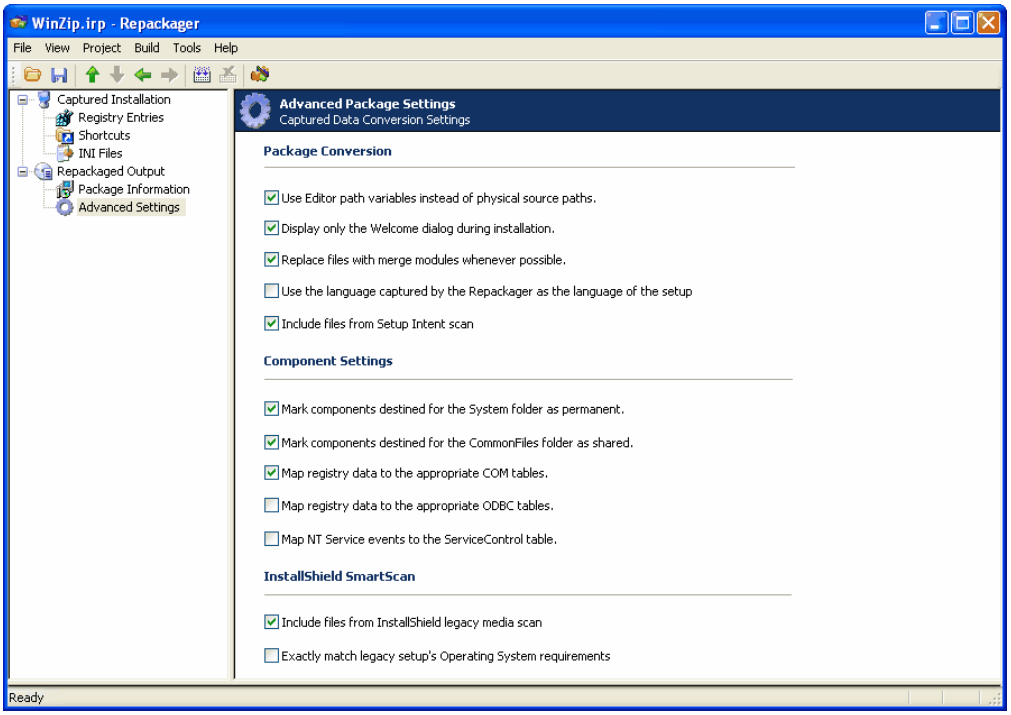


Figure 9-19: Repackager Advanced Package Settings View

Select the appropriate options:

Package Conversion Options

The following package conversion options are available:

Table 9-24 • Package Conversion Options

Option	Description
Use Editor path variables instead of physical source paths	When storing files in the InstallShield Editor project (`.ism`), the Wizard uses path variable locations whenever possible.
Display only the Welcome dialog box during installation	Only the Welcome dialog box is displayed when the Windows Installer package is run on a target machine. If this option is unchecked, the default UI sequence is displayed when the setup is installed.
Replace files with merge modules wherever possible	Following best practice rules, Repackager replaces components with comparable merge modules whenever possible.
Use the language captured by the Repackager as the language of the setup	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view).
Include files from Setup Intent scan	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project).

Component Settings Options

The following component settings options are available:

Table 9-25 • Component Settings Options

Option	Description
Mark components destined for the System folder as permanent	Executable files installed to the system folder (System32Folder) are marked as Permanent files and will not be uninstalled when the package is uninstalled. This eliminates ICE09 validation errors.
Mark components destined for the CommonFiles folder as shared	Executable files installed to the CommonFilesFolder (or a subfolder of CommonFilesFolder) are marked as shared files. This ensures that these components can coexist with DLLs installed by previous setups.
Map registry data to the appropriate COM tables	Setting this option reduces the number of ICE33 warnings that can occur during package validation, resulting from data not being mapped to the appropriate MSI tables.
Map registry data to the appropriate ODBC tables	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is highly recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer.
Map NT Service events to the ServiceControl table	If selected, NT Service–related registry data is mapped to ServiceControl table instead of the Registry table.

Setup Intent Wizard

Although an installation may have intended to install certain files, these files sometimes may not be installed—often because the files already exist on the target machine (either as the same version or a newer version). These files, although not installed or updated, are needed for the product to execute properly when the setup is run on a system that does not already have these files.

The Setup Intent Wizard allows you to scan a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files.

Tip *• Any files found will be displayed in Repackager in a different color (as specified in the* ***Color*** *tab of the Options dialog box).*

The Setup Intent Wizard consists of the following panels:

- Welcome Panel
- Scanning Project Panel
- Results Panel

Welcome Panel

The first panel in the Setup Intent Wizard informs you the purpose of the Wizard, and warns you the source files for your project must be present for successful scanning.

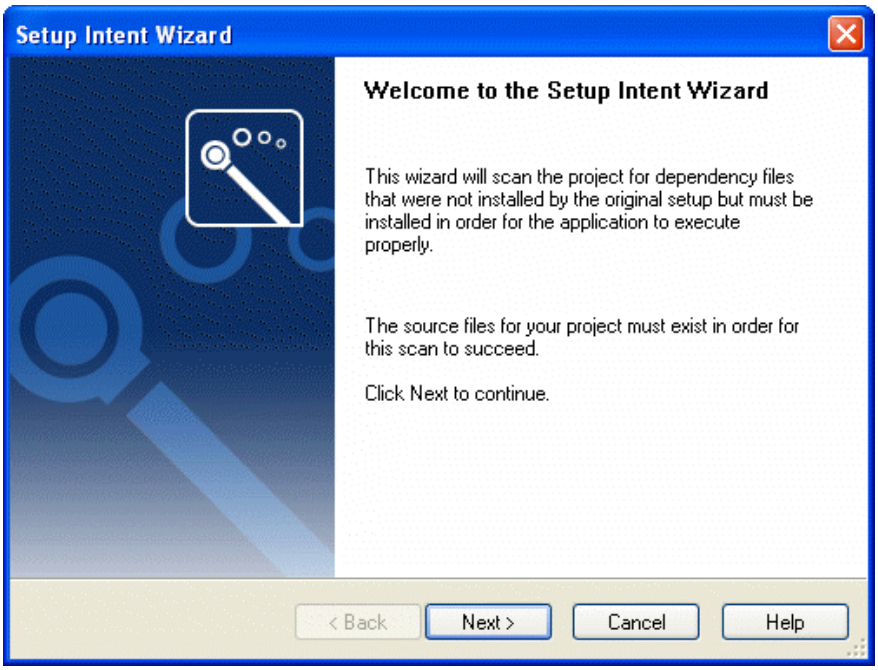


Figure 9-20: Setup Intent Wizard Welcome Panel

Click Next to start the scan and display the **Scanning Project Panel**.

Scanning Project Panel

The Scanning Project Panel is displayed while scanning is in progress. Each file scanned is listed, and a progress bar displays the overall scan progress.

When the scan is complete, the **Results Panel** opens, listing new files that your setup required.

Results Panel

The final panel in the Setup Intent Wizard allows you to view and select new files detected by the Wizard, but not already included in your Repackaging project.

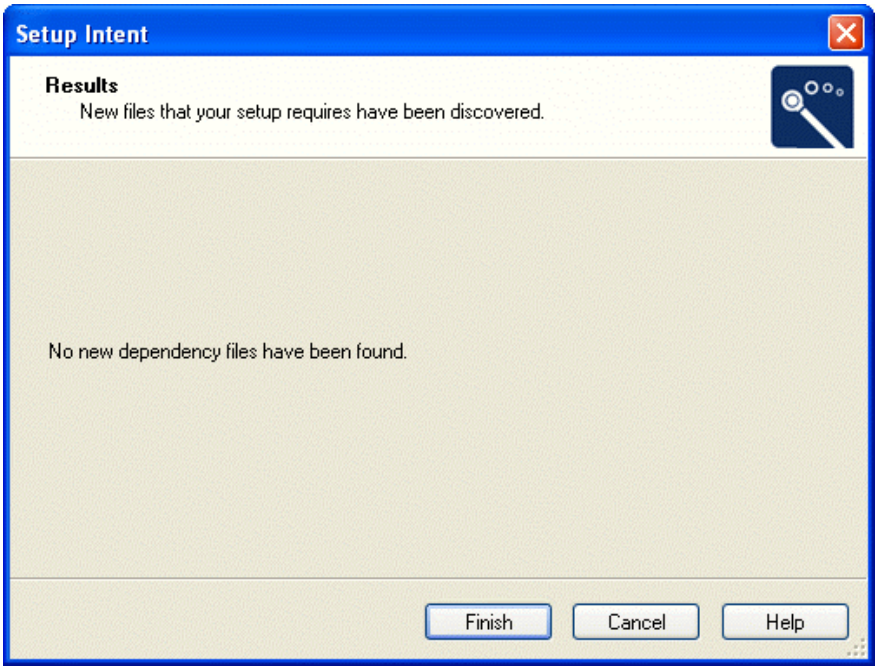


Figure 9-21: Setup Intent Wizard Results Panel

Select the files you want to include in your project which were not identified during repackaging.

Click **Finish** to exit the Setup Intent Wizard and return to the Repackaging project (with selected files automatically added to the project), or click **Back** to return to the **Scanning Project Panel**.

VMware Repackaging Wizard

Repackager includes integration with VMware Workstation's virtual machine technology. This provides you with the ability to launch a VMware session for repackaging purposes, and run different operating systems on the same computer. By using VMware, you are able to forego the traditional "ghosting" for clean images each time a new application is repackaged by simply electing not to save changes to the VMware session. You can then reload the clean state of the operating system, and proceed to the next package.

Note • *AdminStudio supports VMware 3.0 and later.*

Note • *The **VMware Repackaging Wizard** menu item on the **Tools** menu is enabled if Repackager finds VMware 3.0 and later installed on the workstation, and if a VMware image exists on that machine. If no VMware images are found, the VMware Repackaging Wizard menu item will be disabled. Repackager reads the information about VMware images from a* `.vmls` *file that can usually be found in the VMware installation directory or in a subdirectory of the user's* `AppData` *directory. The* `.vmls` *file is a text file that contains information about individual VMware images and where the configuration file for each image is located. This file should contain information for at least one VMware image for the **VMware Repackaging Wizard** menu item to be enabled.*

Using the VMware Repackaging Wizard, you select an available VMware operating system, and then Repackager automatically launches the selected operating system within a VMware session.

The VMware Repackaging Wizard includes two panels:

- Welcome Panel
- VMware Virtual Machines Panel

Welcome Panel

The first panel displayed in the VMware Repackaging Wizard is the Welcome panel. It explains the purpose of this Wizard: to display available VMware images on the current workstation, allowing you to select and launch the one you need.

VMware Virtual Machines Panel

On the VMware Virtual Machines panel, you select a VMware virtual machine available on the current workstation. Repackager automatically launches the selected virtual machine operating system within a VMware session so that you can begin repackaging in that environment.

Note • *AdminStudio supports VMware 3.0 and later.*

Click Back to return to the **Welcome Panel**; click Launch to launch the selected VMware image.

Exclusions Editor Interface

The following topics cover each tab, menu, and dialog box in the Exclusions Editor:

- Menus
- Files Tab
- .ini Files Tab
- Registry Tab
- File Exclusion Information Dialog Box
- INI File Exclusion Information Dialog Box
- Choose Registry Key Dialog Box
- Edit Registry Key Dialog Box
- About Exclusions Editor Dialog Box

Menus

Menus are not available when running the Exclusions Editor from within Repackager. They are only available when you launch the Exclusions Editor by opening the following file:

```
[AdminStudioInstallDirectory]\Repackager\AnalysisOptions.exe
```

Note • *See Launching Exclusions Editor for more information.*

The following table provides a description of each menu command:

Table 9-26 • Exclusions Editor Menu Commands

Menu	Command	Keyboard Shortcut	Description
File	**New**	Ctrl+N	Creates a new, blank settings file.
File	**Open | Shared Exclusions**		Opens the settings file (`isrepackager.ini`) from the `AdminStudio Shared` directory. Open this settings file when working in a team environment where the exclusion list needs to be stored in a centralized location.
File	**Open | Custom Exclusions**		Allows you to browse to an Exclusions Editor settings file and open it. You would create a custom exclusion file based upon your company's requirements.
File	**Save**	Ctrl+S	Saves the current Exclusions Editor settings file.
File	**Save As**		Saves the current Exclusions Editor settings file to the name and location specified.
File	**Exit**		Exits the Exclusions Editor.
Help	**Help Library**		Displays the online Help Library.
Help	**About Exclusions Editor**		Displays the About Exclusions Editor dialog box.

Files Tab

File exclusions for Repackager indicate which files are automatically marked as excluded in the Repackager project. File exclusions in the OS Snapshot Wizard indicate files that will be excluded from the captured OS snapshot.

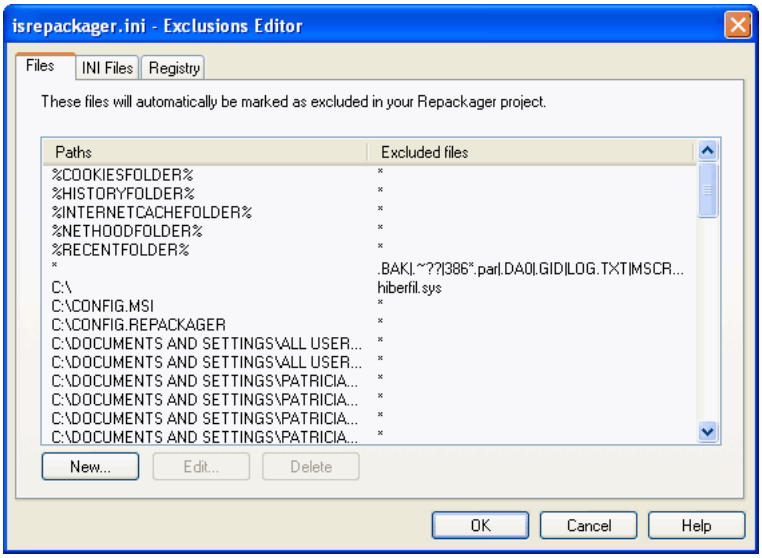


Figure 9-22: Exclusions Editor Files Tab

This **Files** tab contains a list of paths and files currently excluded from the capture process. Specific files, file extensions, and the entire contents of specified directories can be excluded.

The following three buttons allow you to add, edit, and remove files and directories from the exclusion list:

Table 9-27 • Exclusions Editor / Files Tab Buttons

Button	Description
New	Displays the **File Exclusion Information** dialog box from which you can specify additional file exclusions.
Edit	Brings up a dialog box from which you can change settings for the currently selected path in the exclusion list.
Delete	Deletes the currently selected path from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

.ini Files Tab

.ini file exclusions for Repackager indicate which .ini files and sections are automatically marked as excluded in the Repackager project. `.ini` file exclusions in the OS Snapshot Wizard indicate `.ini` files and sections that will be excluded from the captured OS snapshot.

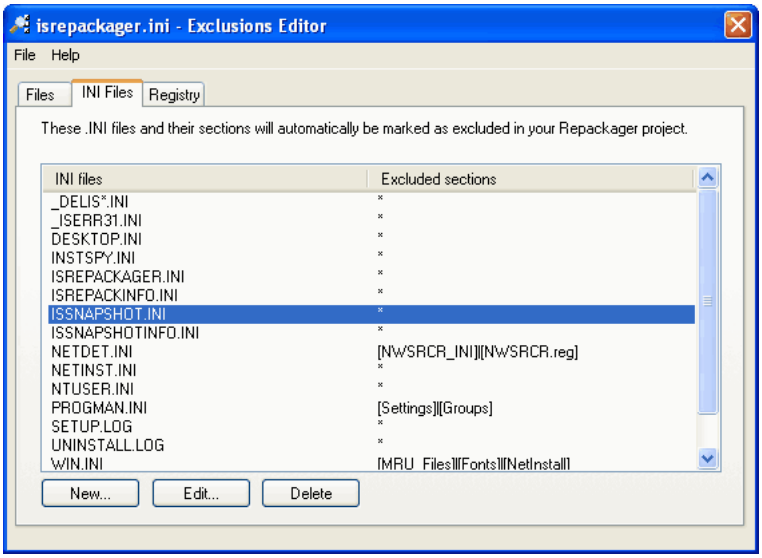


Figure 9-23: Exclusions Editor INI Files Tab

The **INI Files** tab contains a list of the `.ini` files and sections within `.ini` files excluded during analysis. If all sections are excluded, an asterisk (*) is used in the Excluded Sections column.

The following three buttons allow you to add, edit, and remove `.ini` files from the exclusion list:

Table 9-28 • Exclusions Editor / .ini Files Tab Buttons

Button	Description
New	Displays the **INI File Exclusion Information** dialog box from which you can specify additional .ini file exclusions.
Edit	Brings up a dialog box from which you can edit currently excluded .ini files.
Delete	Deletes the selected .ini file from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

Registry Tab

Registry exclusions for Repackager indicate which registry keys are automatically marked as excluded in the Repackager project. Registry exclusions in the OS Snapshot Wizard indicate registry keys that will be excluded from the captured OS snapshot.

The **Registry** tab contains a list of keys and values to be excluded during registry analysis. For keys that have specific values excluded, the value name appears in the Value column. For keys that have all values excluded, an asterisk (*) represents the entire key in the Value column.

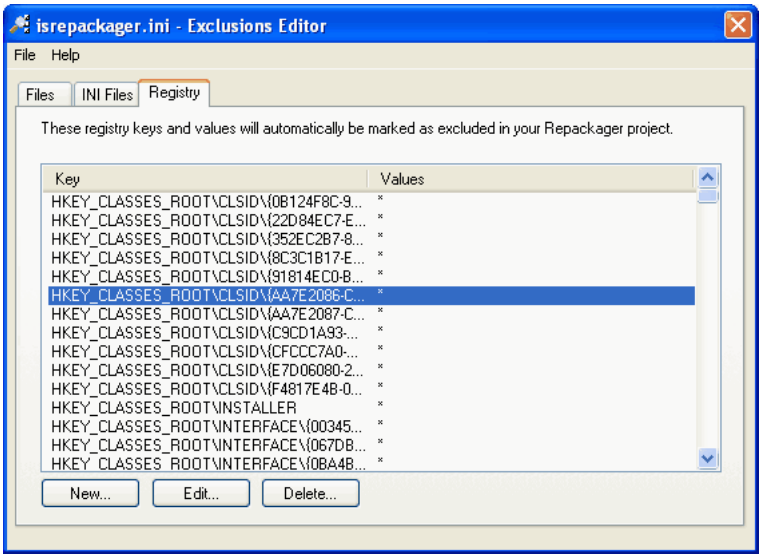


Figure 9-24: Exclusions Editor Registry Tab

There are three buttons available from this dialog box that are used to add, edit, or remove keys from the exclusion list:

Table 9-29 • Exclusions Editor / Registry Tab Buttons

Button	Description
New	Displays the **Choose Registry Key** dialog box, from which you can select registry keys and values for exclusion during analysis.
Edit	Brings up a dialog box from which you can modify the selected key's exclusion settings.
Delete	Removes the selected key from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

File Exclusion Information Dialog Box

The **File Exclusion Information** dialog box, which is accessed by clicking **New** or **Edit** on the Files Tab, allows you to specify files to be excluded from analysis by the capture tool.

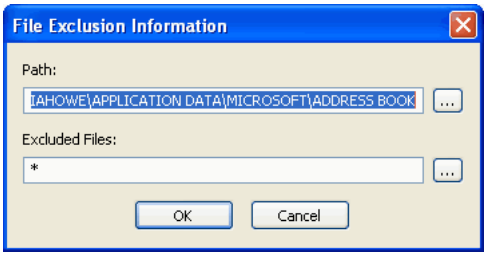


Figure 9-25: File Exclusion Information Dialog Box

Click the Browse button next to the **Path** field and select the directory that contains the file or files you want to exclude. Then, click the Browse button next to the **Excluded Files** field and select the file or files you want to exclude. In the **Excluded Files** field, you can specify files to exclude in the following ways:

- To exclude multiple files from the same directory, separate the file names with pipes (|), such as:

  `file.dll|myfile.exe|anotherfile.exe`

- To exclude all files with a certain extension in the selected directory, enter an asterisk (*) plus the extension, such as:

  `*.txt`

- To exclude all files in the selected directory, enter an asterisk (*) .

Click **OK** to return to the Files Tab.

INI File Exclusion Information Dialog Box

The INI File Exclusion Information dialog box, which is accessed by clicking **New** or **Edit** on the .ini Files Tab, allows you to specify `.ini` files to be excluded from analysis by the capture tool.

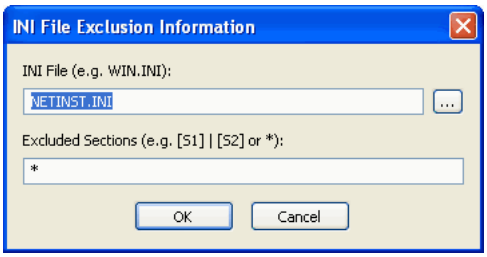


Figure 9-26: INI File Exclusion Information Dialog Box

Enter or browse to the `.ini` file you want to exclude, and provide the section(s) to be excluded. Sections must be enclosed in square brackets ([]), and separated by vertical bars (|) if more than one section in an `.ini` file is to be excluded (for example, [Groups],[Settings]). You can also exclude all `.ini` file sections by only entering an asterisk in the Excluded Sections field.

Click **OK** to return to the .ini Files Tab.

Choose Registry Key Dialog Box

The Choose Registry Key dialog box, which is accessed by clicking **New** on the Registry Tab, provides a way for you to select registry keys that you want excluded from analysis.

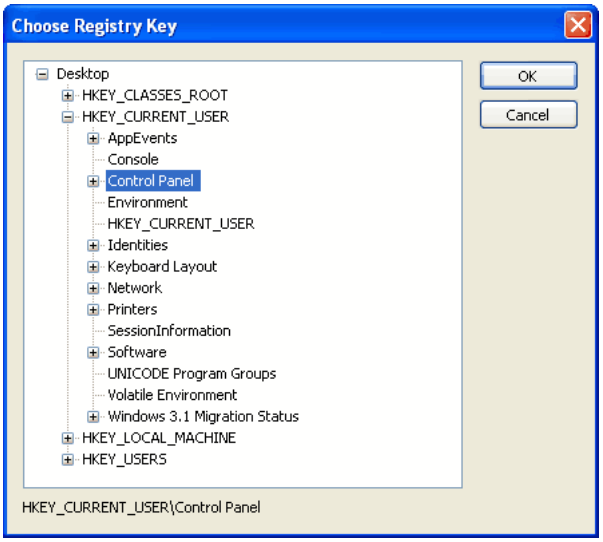


Figure 9-27: Choose Registry Key Dialog Box

Navigate through the tree until you find the key you want to exclude and click **OK** to return to the Registry Tab.

By default, all values in that key are excluded. To modify this, select the key from the Registry Tab and click **Edit** to display the **Edit Registry Key** dialog box.

Tip • *You can also select a registry hive to exclude. As with individual registry keys, all values (and keys) contained in the hive are excluded by default.*

Edit Registry Key Dialog Box

When you select a registry key on the Registry Tab and click **Edit**, the **Edit Registry Key** dialog box opens.

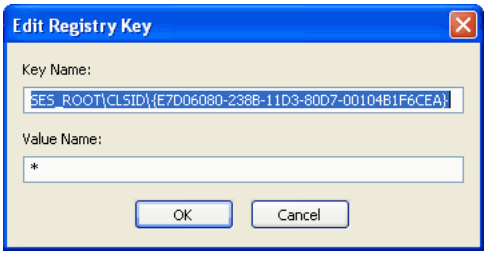


Figure 9-28: Edit Registry Key Dialog Box

You can modify the **Key Name** and/or **Value Name** excluded during analysis.

Click OK to return to the Registry Tab.

About Exclusions Editor Dialog Box

The About Exclusions Editor dialog box displays version and copyright information for the Exclusions Editor. This may be useful if you need to report a problem encountered when using the Exclusions Editor.

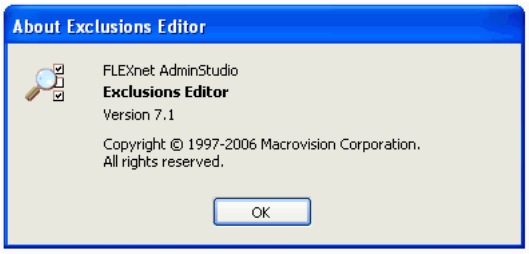


Figure 9-29: About Exclusions Editor Dialog Box

Options.ini File

The `Options.ini` file is created by Repackager and is used during the conversion of Repackager output into an InstallShield Editor project (`.ism`). It includes basic project settings which are required by Repackager. Information about this file is presented in the following sections:

- [MMExclusions] Section
- [General] Section
- [IgnoreShortcuts] Section
- Options.ini File Defaults

Note • *Although many of these settings have a one-to-one correspondence with settings available in the Repackager interface, some can only be accessed by editing this .ini file directly.*

[MMExclusions] Section

This section lists the merge module GUIDs that should not be included in your package. This section only applies if you have selected to replace files with merge modules during conversion.

[General] Section

Following are descriptions of properties that can be set in the [General] section of the Options.ini file.

Table 9-30 • Options.ini File/General Section Properties

Properties	Description
AddIMMSearchPath	Use to specify additional directories containing custom merge modules to use during repackaging.
ALLUSERS	If this option is set to **Y** and if the template file (specified using the `ProjectTemplate` option) does not contain ALLUSERS in its Property table, then a property named ALLUSERS with a value of 2 will be added to the Property table. This will cause silent installs to behave as non-silent installs do (non-silent installs rely on a custom action to set this property). This option is set to **Y** by default.

Table 9-30 • Options.ini File/General Section Properties (cont.)

Properties	Description
ARPPublisher	This populates the **Publisher** field in **Add/Remove Programs** in the Control Panel.
ARPPublisherURL	This populates the **Publisher URL** field in **Add/Remove Programs** in the Control Panel.
ARPSupportURL	This populates the **Support URL** field in **Add/Remove Programs** in the Control Panel.
AutoUpgrade	Upgrades the InstallShield Editor template project file (if used) if needed. This option is set to **Y** by default.
BuildCompressed	This option, set to **Y** by default, compresses all necessary files inside the MSI package, as opposed to storing them outside of the MSI database.
BuildFeatures	No longer used.
BuildMSI	Specifies whether or not to build the MSI package after building ISM. This option is set to **Y** by default.
BuildProduct	Identifies the InstallShield Editor Product configuration to build.
BuildProScannedFiles	Files identified in the Media Scan Wizard will be included in the package (unless you have manually excluded them from the project). This option is set to **Y** by default.
BuildRelease	Identifies which InstallShield Editor Release configuration to build.
BuildStaticScannedFiles	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project). This option is set to **Y** by default.
COMMapping	When this option is set to **Y**, registry data pertaining to COM information will be mapped to the appropriate MSI tables whenever possible. This reduces the number of ICE33 warnings that can occur during package validation. This option is set to **Y** by default.
CreateSetupExe	This option, which is set to **N** by default, allows you to automatically create a `Setup.exe` file to begin the installation.

Table 9-30 • Options.ini File/General Section Properties (cont.)

Properties	Description
EnablePathVariables	Set this option to **Y** to use path variables. If enabled, the repackaged setup is significantly more portable between computers (with dependencies to the system where the setup was repackaged removed). This option is set to **Y** by default.
ExtraHKCRPermanent	When this option is set to **Y**, any changes made to existing registry data during repackaging which cannot be identified as belonging to a file installed by the setup are placed in permanent components, which are not removed by default when the repackaged setup is uninstalled. This prevents inadvertently removing registry entries required by other applications that were not originally made by the repackaged setup. By default, this option is set to **Y**, and it is strongly recommended that you retain this setting to prevent unexpected results when the package is uninstalled.
INSTALLDIR	This value will be used for `INSTALLDIR` (the installation directory) and can use a Windows Installer property such as `[ProgramFilesFolder]\MyProgram`
ISProSetup	If one of the original setups that was repackaged was created by InstallShield Professional 5.5 or later, this option will be set to **Y**. This option is set to **N** by default.
LimitedUI	Set this option to **Y** display only the InstallWelcome dialog box when the MSI package is run. This option is set to **Y** by default.
MultiUserShortcuts	When this option is set to **Y**, non-advertised shortcuts will work for all users on the target system. This will generate ICE43 warnings when validation is run. If you know the installation is for a single-user environment, change this option to **N** to avoid these warnings. This option is set to **Y** by default.
MMPathVersion	When including merge modules, if this option is set to **Y**, compare path and version information. This option is set to **Y** by default.
NewInstallDir	Value for INSTALLDIR variable.

Table 9-30 • Options.ini File/General Section Properties (cont.)

Properties	Description
ODBCMapping	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer. This option is set to **N** by default.
OSGranular	For ProLogged Projects: ● **If this is set to Y**, component conditions will store specific operating system information. For example, if the filter is NT4, the condition will be (VersionNT=4). ● **If this is set to N**, component conditions will store a grouping of the operating system. For example, if the filter is NT4, the condition will be (VersionNT). This is set to **N** by default.

OtherComponentFileExtensions	Specify additional extensions to use when defining components. MSI has rules governing component creation for file types. For example, portable executable (PE) files must have separate components. Therefore, certain extensions have been defined (EXE, DLL, etc.). Additional extensions can be defined in the options.ini file in the format of:
	
	`Type1:Extension1|Type2:Extension2`
	
	where Type is one of the following numbers:
	
	`0 = other`
	`1 = PE`
	`2 = help`
	`3 = font`
	`4 = INI`
	
	This option is set to 1:QTX|1:AX by default.

OtherFilesNewComponents	When this option is set to Y, one component will be created for every file in your setup. Otherwise, new components will only be created for each portable executable file. This option is set to N by default.
PermanentSystemFiles	Set this option to Y to mark portable executable files installed to a system folder (System32Folder) as Permanent files (will not be uninstalled). This option is set to Y by default.

Table 9-30 • Options.ini File/General Section Properties (cont.)

Properties	Description
PermanentSystemFilesSubfolders	Set this option to Y to mark files installed to a subfolder of a system folder as Permanent files (will not be uninstalled). This option is set to N by default.
ProductName	The name of the product. You must provide a value for this option either in this file or in Repackager.
ProductVersion	The version of the product. You must provide a value for this option either in this file or in Repackager.
Project	Name of InstallShield Editor project file.
ProjectTemplate	The name and location of the default InstallShield Editor project template (`.ism`) used in the conversion process.
ServiceControlEvents	When this option is set to Y, the ServiceControl table will be populated for NT Services. This option is set to N by default.
SharedCommonFiles	Set this option to Y to mark portable executable files installed to the CommonFilesFolder (or subfolder) as Shared files. This option is set to Y by default.
SISAuthor	This option populates the Author field of the Summary Information Stream (accessible from the package's properties). This option is set to Repackager by default.
SISSubject	This option populates the Subject field of the Summary Information Stream (accessible from the package's properties).
SkipMMIfShortcut	Merge Modules that have files pointed to by shortcuts should be skipped even if they are not in the exclusion list. This option is set to Y by default.
UseAdvertisedShortcuts	Create advertised shortcuts where applicable. This option is set to Y by default.
UseHKCUProxy	Set this option to Y to copy all registry entries in HKEY_CURRENT_USER to HKEY_USERS\.default. This option is set to N by default.

Table 9-30 • Options.ini File/General Section Properties (cont.)

Properties	Description
UseLanguage	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view). This option is set to N by default.
UseMergeModules	Set this option to Y to replace files with merge modules whenever possible during conversion. Exceptions are listed under the [MMExclusions] section. This option is set to Y by default.
UseSrcFolder	Set this option to Y to make the InstallShield Editor project (`.ism`) folder default to the Repackager output project (`.inc`) folder. This option is set to Y by default.

[IgnoreShortcuts] Section

Shortcuts that refer to executables listed in this section will be ignored during conversion.

Options.ini File Defaults

This section lists the default settings in the `Options.ini` file that is shipped with Repackager:

```
[MMExclusions]

[General]
UseSrcFolder=Y
EnablePathVariables=Y
UseHKCUProxy=N
LimitedUI=Y
SISAuthor=InstallShield Repackager
OtherFilesNewComponents=N
UseMergeModules=Y
SharedCommonFiles=Y
PermanentSystemFiles=Y
PermanentSystemFilesSubfolders=N
ExtraHKCRPermanent=Y
COMMapping=Y
ODBCMapping=N
ServiceControlEvents=N
ALLUSERS=Y
ProjectTemplate=
BuildCompressed=Y
CreateSetupExe=N
MultiUserShortcuts=Y
ISProSetup=N
BuildFeatures=Y
OtherComponentFileExtensions=1:QTX|1:AX
OSGranular=N
MMPathVersion=N

[IgnoreShortcuts]
```

```
TargetExe1=isuninst.exe
TargetExe2=uninst.exe
TargetExe3=setup.exe
TargetExe4=uninst.dll
TargetExe5=rnuninst.exe
```

Files Associated with Repackager

Several files are associated with Repackager. Some are output files, and some contain default information for Repackager to function. These files are described in the tables below.

Files Used By the Repackaging Wizard

The following files are used by the Repackaging Wizard.

Table 9-31 • Files Used by the Repackaging Wizard

File	Location	Description
`Repack.ini`	**Windows** If the file is not found in the Windows directory, then the Repackaging Wizard extracts a default file from the resource and stores it in the Windows directory.	This is an input file for the Repackaging Wizard. It contains a list of the exclusions for the files, folders, `.ini` files and registry entries for the last used configuration of Repackager. During the Snapshot and Install Monitoring modes of repackaging, the entries in this file are filtered out from the repackaged output. See Repack.ini File for more information.
`Options.ini`	Repackager output directory (specified in the Set Target Project Information and Capture Settings Panel). The Repackaging Wizard makes a copy of the default `options.ini` that is present in the following directory: `[AdminStudioInstallDirectory]\Repackager` and saves this file in the same location as the current repackaged output file (`.inc`). Additionally, the `UseSrcFolder` flag can be used to store the created InstallShield Editor file in the same directory as the `.inc` file.	This is an output file from the Repackaging Wizard. It contains configuration information about the repackaged setup, including whether to use path variables, whether to display a limited user interface during installation of the repackaged setup and whether every file will go into its own component.

Table 9-31 • Files Used by the Repackaging Wizard (cont.)

File	Location	Description
`productname.inc`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard. It contains the locations of files, `.ini` files, and shortcuts detected by Repackager as having been created, modified, or removed during repackaging. Also, it contains a link to the `standard.nir` and `deleted.isr` files for registry information.
`updated.isr`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard when the Install Monitoring method is used. It contains registry additions and modifications detected during repackaging using installation monitoring only.
`deleted.isr`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard. It contains registry deletions detected during repackaging using Installation Monitoring and Snapshot.
`standard.nir`	Created in the Repackager output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard when the Snapshot method is used. It contains registry additions and modifications detected during repackaging using the Snapshot method.
`*.spy`	Created in the following folder: `WindowsDrive\InstallHook`	This is an output file from the Repackaging Wizard when the Install Monitoring method is used. It contains AdminStudio PowerShell Cmdlet call logs for installation monitoring.
`Default.ini`	`[AdminStudioInstallDirectory]\Repackager`	Contains the default configuration for Repackager, including default exclusion information.
`Repack.log`	`WindowsFolder`	Log file created by the Repackaging Wizard.

Files Used By the Repackager Interface

The following files are used by the Repackager interface.

Table 9-32 • Files Used by the Repackager Interface

File	Location	Description
`*.irp`	Saved in the same location as the `.inc` file.	This is a Repackager project file. It is the main file for each repackaged or converted setup. It contains information about the .inc files referred to and also stores the file, folder, `.ini` files and registry exclusions made in the Repackager Interface.
`<Exclusion List>.ini`	varies	This is an input file for the conversion of the .inc file to an MSI package. It contains the list of files, folder, `.ini` files and registry entries exclusions. Users can choose a different exclusion file from the Repackager Interface and the exclusions will be reflected in the Interface.
`Options.ini`	Saved in the same location as the `.inc` file.	This is an input file for the conversion of the `.inc` file to an MSI package. It contains configuration information about the repackaged setup, including whether to use path variables, whether to display a limited user interface during installation of the repackaged setup, and whether every file will go into its own component. Additionally, the `UseSrcFolder` flag can be used to store the created InstallShield Editor file in the same directory as the `.inc` file.

Repack.ini File

The `Repack.ini` file is the default capture exclusion file for the Repackaging Wizard. It contains exclusions to be applied during repackaging, and mainly focuses on specific items that should not be included in applications, such as InstallShield Professional-specific COM settings, OS settings, and Internet Explorer settings. Any item excluded during capture will not be available for exclusion/inclusion in the Repackager project file.

The file is located in the Windows folder, and can be edited using the Exclusions Editor, or using a text editor.

Note • *It is strongly recommended that you not modify this file, as it increases the likelihood of either inadvertently omitting necessary pieces of applications you are repackaging, or including registry entries or files that should not be part of the repackaged application. In the first scenario, you may need to recapture your application; in the second, you may need to exclude more from the Repackager project.*

Instead, capture your application using the default exclusions in the Repackaging Wizard, and then selectively exclude captured data using the Repackager Interface. This way, if you inadvertently exclude a necessary piece, you need only reinclude it in Repackager—not recapture the application entirely.

Using InstallShield to Chain Multiple Windows Installer Packages Together

If your application includes more than one Windows Installer (`*.msi`) package, you can use InstallShield Editor to chain them together using a nested MSI Custom Action. This enables you to run multiple MSI files within a single setup process.

To do this, you open the InstallShield Editor **Custom Actions** view and use the **Custom Action Wizard**.

Task ***To add a Nested MSI Custom Action:***

1. Launch **InstallShield Editor**.
2. Open your Windows Installer package in Direct Edit Mode.
3. In the Installation Designer, expand the **Behavior and Logic** tree and select the **Custom Actions** node. The Custom Actions view opens.
4. In the middle pane, right-click **Custom Actions** and then click **Custom Action Wizard**.
5. Follow the **Nested Installations** procedure in the InstallShield Editor user documentation to create a nested MSI Custom Action.

Troubleshooting

Repackager Troubleshooting information is presented in the following topics:

- Troubleshooting Guidelines for WinINSTALL Conversion
- Troubleshooting Guidelines for SMS Conversion
- Resolving an “Error Building Table File” Error

Troubleshooting Guidelines for WinINSTALL Conversion

Use the following troubleshooting guidelines to identify and fix WinINSTALL conversion problems.

- **Repackager tool supports 6.0, 6.5, and 7.x project formats only.** For all other formats, please use the WinINSTALL LE tool available as a free download in Windows 2000 to convert to 7.x files.
- **Repackager tool cannot convert WinINSTALL .NAI files**—It can only convert WinINSTALL projects that have been converted to text (`.txt`).
- **All files must be available**—All the files that were available to the original WinINSTALL installation project must be available to the converted installation at the exact same locations.
- **Not all elements of a WinINSTALL installation are converted**—Because WinINSTALL installations are based on a different technology than Windows Installer, not all elements of a WinINSTALL installation are converted. Only the installation of files, registry changes, and other system changes are converted.
- **Custom logic is not converted**—Custom logic written in WinINSTALL’s custom scripting language is not converted.

- **WinINSTALL environment variable assignments are not converted**—To re-add environment variable assignments in a Windows Installer installation, open the converted project in InstallShield Editor and use the Environment Variable view.
- **WinINSTALL variables are converted to a Windows Installer variable**—If the target path of a file contains a WinINSTALL variable, then the WinINSTALL variable is converted to a Windows Installer variable.
- **Specify @ variables at conversion time**—If the source path of a file in WinINSTALL contains either the @Server or @Wininstall variable, you can specify the values of these two variables at conversion time in the Repackager.
- **The WinINSTALL Preinstall and Postinstall scripts are not converted.**

Troubleshooting Guidelines for SMS Conversion

Use the following troubleshooting guidelines to identify and fix SMS conversion problems.

- **All files must be available**—All the files that were available to the original SMS installation project must be available to the converted installation at the exact same locations.
- **Not all elements of an SMS installation are converted**—Because SMS installations are based on a different technology than Windows Installer, not all elements of a SMS installation are converted. Only the installation of files, registry changes, `.ini` Files, ODBC, NT Services, Fonts, Shortcuts, Variables, and other system changes are converted.
- **Custom logic is not converted**—Custom logic written in SMS's custom scripting language is not converted.
- **SMS environment variable assignments are not converted**—To re-add environment variable assignments in a Windows Installer installation, open the converted project in InstallShield Editor and use the Environment Variable view.

Resolving an "Error Building Table File" Error

When building with Repackager, if you have received the following error message during the build:

```
ISDEV: fatal Error 5023: Error building table File
```

your first step is to go to the Repackager Interface and check whether the number of files installed by this setup is greater than 32,767. If it is, this error occurs because Windows Installer supports 32,767 files in the File table but the package being built exceeds this limit. See Authoring a Large Package in Windows Installer Help for more information.

If you want to fix this error using Repackager, perform the steps listed below.

Task ***To fix this error using Repackager:***

1. Browse to the appropriate directory:
   - If you are using the standalone Repackager, browse to the Repackager folder.
   - If you are using the Repackager on a machine where AdminStudio is fully installed, browse to the following directory:

     `<AdminStudio INSTALLDIR>\Editor\Support\0409`
2. Locate the `IsMsiPKg.itp` and `IsMsiPKgLarge.itp` files in this directory.
3. Rename `IsMsiPKg.itp` to `IsMsiPKg.itp.bak`.

4. Make a copy of `IsMsiPKgLarge.itp` and rename the copy `IsMsiPKg.itp`.
5. Perform the conversion and create the MSI.
6. Delete `IsMsiPKg.itp`.
7. Rename `IsMsiPKg.itp.bak` back to `IsMsiPKg.itp`, thereby restoring the original file.

Note • *Transforms and patches cannot be created between two packages with different column types.*

Note • *For more information, see the Authoring a Large Package and File Table topics in the Windows Installer Help.*

10

Performing Virtualization and Repackaging Using the Automated Application Converter

Edition • *The Automated Application Converter is included with AdminStudio Professional and Enterprise Editions, and you can use it to perform automated repackaging on a virtual machine.*

You can use the Automated Application Converter to convert a single package or a group of packages into Microsoft App-V, VMware ThinApp, Citrix XenApp.

When converting a Windows Installer package to a virtual package, you often need to repackage it prior to being able to successfully convert it. The reason for this is that it is not possible to determine the run-time behavior of certain Windows Installer package elements—such as custom actions, conditional components, and launch conditions—without actually installing the package.

You can use the Automated Application Converter to:

- Examine a group of selected setups.
- Perform automated virtualization of setups that can be cleanly virtualized.
- Perform automated repackaging of those setups that cannot be cleanly virtualized (due to custom actions, etc.), and then perform automated virtualization of those repackaged MSIs.

Note • *You can also use the Application Catalog* ***Conversion Wizard*** *to quickly convert one or multiple Windows Installer packages or legacy installers to virtual packages using default Automated Application Converter settings. For more information, see Using the Conversion Wizard to Perform Express Conversion to Virtual Packages or Automated Repackaging.*

Information about using the Automated Application Converter is presented in the following sections:

Table 10-1 • AdminStudio Automated Application Converter Help Library

Section	Description
About the Automated Application Converter	Describes the benefits of using the Automated Application Converter to perform automated repackaging and virtualization, and provides an overview of its workflow.
Getting Started With the Automated Application Converter	Explains how to use the Application Conversion Project Wizard to get started using the Automated Application Converter to perform automated repackaging and virtualization.
Managing Virtual Machines	Explains how to use the Virtual Machine Import wizard to add new virtual images to the Automated Application Converter and how to manage virtual machines on the **Machines** tab.
Managing Packages to Convert	Explains how to use the Package Import Wizard to add packages to the Automated Application Converter, and how to manage packages on the **Packages** tab.
Using the Application Conversion Wizard to Perform Automated Package Conversion	Explains how to use the Application Conversion Wizard to perform a conversion run using the selected packages and virtual machines, and how to view conversion run log report information.
Testing Packages	Explains how to quickly launch a package for testing on a virtual machine directly from Automated Application Converter.
Importing Converted Packages into the Application Catalog	Explains how to import converted virtual packages or repackaged Windows Installer packages into the AdminStudio Application Catalog.
Publishing Converted Packages to a Distribution System	Explains how to publish an application containing converted virtual packages or repackaged Windows Installer packages to a distribution system.
Setting Default Project Properties	Explains how to set project-wide default options on the **Project Options** dialog box.
Capturing Virtualization Context	Explains the purpose of the `packagename.context.msi` file that is created during repackaging.
Reference	Describes each of the user interface elements and Wizard panels that you might encounter when using the Automated Application Converter. The help topics in this Reference section are the same detailed documentation that is displayed when you press the F1 key or click the **Help** button while working in a dialog box.
Troubleshooting	Includes information to help you resolve typical problems that you might encounter when using the Automated Application Converter.

Note • *Automated Application Converter is available in a Single Application Version (one-at-a-time conversion) and a Multiple Application Version (batch conversion). For more information, see AdminStudio Editions and Components.*

Getting Started With Application Virtualization

AdminStudio provides support for the conversion of Windows Installer packages to the following virtual application formats:

- Microsoft App-V virtual packages (versions 4.x and 5.0)
- VMware ThinApp virtual packages
- Citrix XenApp virtual packages

You have several options when deciding how you want to create a virtual application, depending upon your source files, whether you are an enterprise user or an independent software vendor, and the degree of customization you want to perform:

Table 10-2 • Application Virtualization Support in AdminStudio

If you have ...	And want to ...	Use ...	Description
Windows Installer package(s) Legacy application(s) 	**Convert it to a virtual package with ...** • Customized App-V options • Default ThinApp, XenApp, Symantec options • Default isolation options	**Automated Application Converter**	Use Automated Application Converter to convert a single or group of Windows Installer (`.msi`) and legacy (`.exe`) packages to virtual applications. For detailed information, see Using the Application Conversion Project Wizard to Perform an End-to-End Conversion. ***Note*** • *For more information, see Using Automated Application Converter vs. the InstallShield Virtual Assistants.*

Table 10-2 • Application Virtualization Support in AdminStudio (cont.)

If you have ...	And want to ...	Use ...	Description
Windows Installer package or InstallShield project	**Convert it to a virtual package with ...** • Modified package contents, registry settings, and shortcuts • Custom isolation options on folders and registry entries • Operating system and/or language requirements	**InstallShield Editor Microsoft App-V Assistant** **or** **InstallShield Editor Citrix Assistant** **or** **InstallShield Editor ThinApp Assistant**	Use the InstallShield Editor Microsoft App-V Assistant, ThinApp Assistant or Citrix Assistant to create a virtual application from an InstallShield project or a Windows Installer package. Customization options include: • Modifying package contents, registry settings, and shortcuts • Setting custom isolation options on folders and registry entries • Setting operating system and/or language requirements • Specifying deployment server ***Note*** • *For more information, see Using Automated Application Converter vs. the InstallShield Virtual Assistants.* For detailed information on how to use the Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant, see *Creating Customized Virtual Applications* in the InstallShield Help Library.

Table 10-2 • Application Virtualization Support in AdminStudio (cont.)

If you have ...	And want to ...	Use ...	Description
Repackager Project	**Convert it to a virtual package with ...** • No modifications • Default isolation options	**Repackager**	By selecting an option on the **Repackaged Output** view, you can simultaneously build an InstallShield Editor project, a Windows Installer package, and any of the following virtual package types: • Microsoft App-V 4.x or 5.0 • VMware ThinApp • Citrix XenApp For information on this feature, see Automatically Generating a Virtual Application During Repackager Project Build.

Using Automated Application Converter vs. the InstallShield Virtual Assistants

Whether you should choose to use Automated Application Converter or an InstallShield Virtual Assistant to perform virtualization could depend upon whether you are a system administrator for an enterprise or an independent software vendor (ISV):

- **Enterprises: Automated Application Converter**—Automated Application Converter is the tool of choice when doing a mass conversion of a variety of setups to virtual packages because it can operate on multiple packages in one project and handle repackaging when it is necessary. This scenario most often applies to enterprises.
- **ISVs: InstallShield Virtual Assistants**—InstallShield Virtual Assistants could be used when focusing on one particular Windows Installer package that does not need to be repackaged. This scenario most often applies to ISVs.

Important • *InstallShield Editor does not support conversion to Symantec Workspace virtual package format. Also, the Microsoft App-V Assistant does not offer the option to perform conversion to App-V 5.0 format using the Microsoft App-V 5.0 Sequencer (which is offered by the Automated Application Converter). The App-V Assistant uses AdminStudio's native technology to perform the conversion.*

The InstallShield Virtual Assistants allow for customizing the various virtualization-related options for converting a Windows Installer package to a virtual package. In addition, it is possible to make modifications to the source Windows Installer `.msi` file.

Most of the package-level virtualization options—such as whether to compress the package or not—are also available in Automated Application Converter, but file and registry-specific isolation options are only available in the InstallShield Virtual Assistants.

Also, while Automated Application Converter enables you to customize the majority of App-V virtualization options, it does not enable you to set VMware ThinApp or Citrix XenApp conversion options. If the user needs to set some ThinApp or XenApp conversion options, then it would be necessary to use the InstallShield Virtual Assistant for VMware ThinApp or Citrix XenApp.

About Application Virtualization

Note • *This section provides a description of virtualization in general for those that are not familiar with it. It does not represent the architecture of any specific vendor.*

A typical Windows application has dependencies on components that are shared by multiple applications, such as registry entries or COM controls. When an installation author recognizes that their application requires a shared component—such as MDAC (Microsoft Data Access Components)—they include a merge module to install that component.

When one of these shared components is installed during an application's installation, it is possible that a previously-installed version of the same component could be overwritten, causing the existing application to break. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment. The following diagram provides an example of two conflicting installed applications.

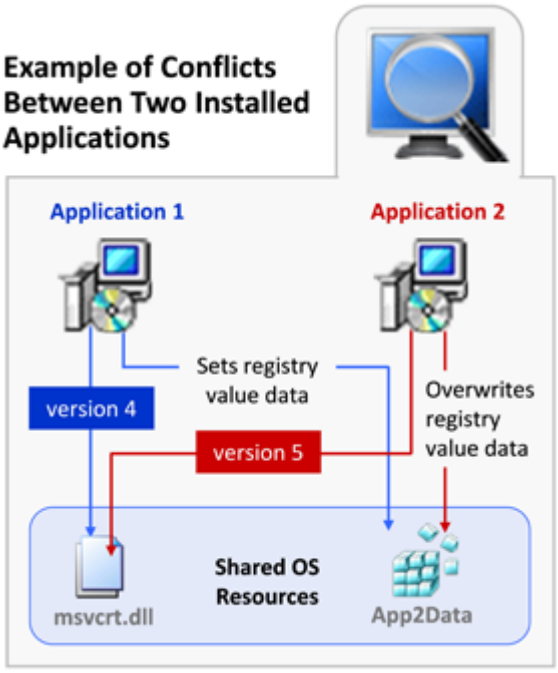


Figure 10-1: Example of Conflicts Between Two Installed Applications

Virtualization simplifies the situation by keeping the application layer and the operating system layer separate, so that the virtual application has no impact on the other applications. In application virtualization, a container or isolation environment is created around the application: a controlled virtual space for application execution that separates the interaction between an application and the underlying operating system's resources in order to protect applications from conflicting with each other.

The following diagram provides an example of how application virtualization would solve the conflicts shown in the previous example.

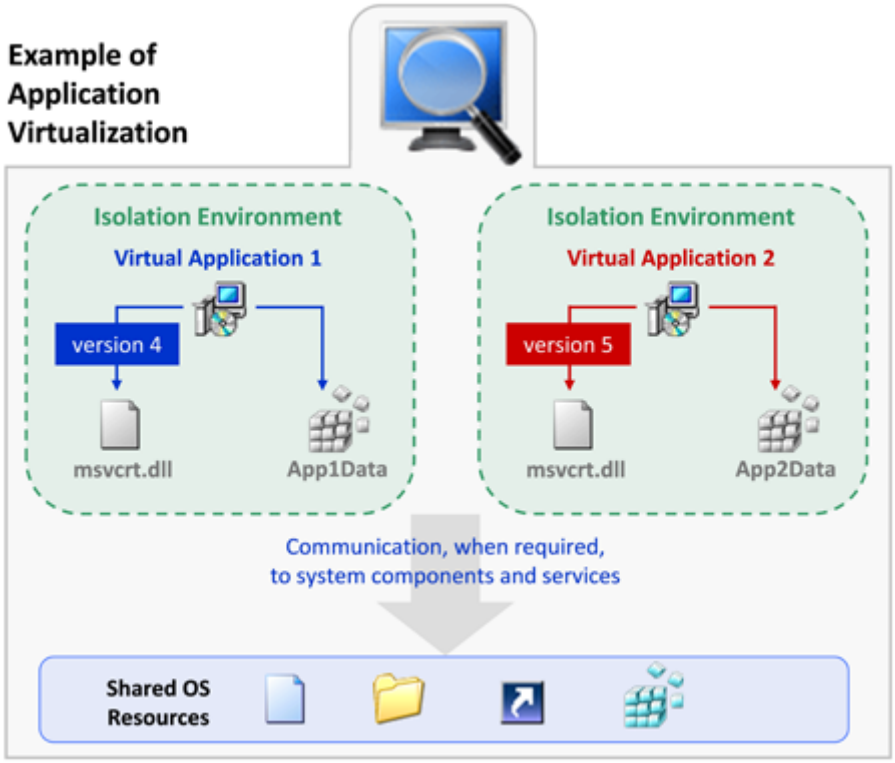


Figure 10-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies compatibility testing.

About Microsoft Application Virtualization

Microsoft Application Virtualization (App-V) provides the capability to make applications available to end user computers without having to install the applications directly on those computers. Information on Microsoft Application Virtualization is presented in the following topics:

- About Microsoft Application Virtualization (App-V)
- Components of an App-V Package
- Comparison of the App-V 5.0 Conversion Methods
- Support for App-V 5.0 SP2 Shell Extension and Runtime Features
- Creating 64-Bit App-V Packages
- Editing an OSD File to Make Advanced Changes to an App-V 4.x Package
- How Windows Services Are Integrated into an App-V Package

About Microsoft Application Virtualization (App-V)

Microsoft Application Virtualization (App-V) enables applications to run as network services, removing the need for local installation of the applications. An App-V package runs in a self-contained, virtual environment. The virtual environment contains the information necessary to run the application on the client without installing the application locally. Only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middle ware, plug-ins, and other applications.

Because App-V packages are not installed on the client, there is minimal impact on the host operating system or other applications. As a result, application conflicts and the need for regression testing are dramatically reduced.

Using Microsoft Application Virtualization enables you to centralize the installation and management of deployed applications, and control access to applications. The App-V client presents to the end user a list of applications to which that user has access.

The Microsoft Application Virtualization (App-V) infrastructure includes:

- **App-V Sequencer**—The App-V Sequencer converts application data into a format which is compatible with the App-V server and client, producing an App-V package.
- **App-V Server**—An App-V package can be placed on one or more App-V servers so that it can be streamed down to the clients on demand and cached locally.
- **App-V Client**—The App-V Client is the system component that enables the end user to interact with the App-V packages that are available on the App-V server.

Components of an App-V Package

Version • *Automated Application Converter has support for both App-V 4.x and 5.0 packages.*

The files that comprise an App-V package depend on the version of the App-V package.

Components of an App-V 5.0 Package (.appv)

The following table describes the main components of an App-V 5.0 package (`.appv`):

Table 10-3 • Components of an App-V 5.0 Package

File	Description
.appv	The `.appv` file is the compressed package file that contains all of the other parts of the package.
AppxBlockMap.xml	This file contains a list of files with details such as header size and file size.
AppxManifest.xml	This file contains information about the package name and version, OS requirements, and all of the integration points with the system such as shortcuts, file type associations, environment variables, services, URL Protocols, Default Programs, Software Clients, and COM registration.

Table 10-3 • Components of an App-V 5.0 Package

File	Description
FilesystemMetadata.xml	This file contains information such as short file names, the directory-file hierarchy, and the mapping between the root folder and `INSTALLDIR`.
Registry.dat	This file contains registry data for the package.
StreamMap.xml	This file contains publishing and primary feature block information.
<package name> _DeploymentConfig.xml	This file provides a way for the user to customize the integration points (such as disabling, modifying, or adding them) and can be applied when publishing the package for all users.
<package name> _UserConfig.xml	This file provides similar functionality as the `DeploymentConfig.xml` file when publishing for individual users. User created scripts can be added to these files as well.

Components of an App-V 4.x Package (.sft)

The following table describes the main components of an App-V 4.x package (`.sft`):

Table 10-4 • Components of an App-V 4.x Package

File	Description
.sft	The `.sft` file contains all of the files, registry information, and other configuration details of the package.
Manifest file	This file is an XML file that lists all of the `.osd` files in an App-V package.
.osd	The `.osd` files are XML-based files that describe the package's shortcuts, file extensions, dependencies, and other data that can influence the environment.
.ico	The `.ico` files are icons files that are used for published shortcuts and file type associations.
.sprj	This file is the Microsoft App-V Sequencer project file. It contains references to the `.sft` and `.osd` files, and to a large number of settings related to the sequencing process.

Comparison of the App-V 5.0 Conversion Methods

When you use AdminStudio to convert a Windows Installer package to App-V 5.0 format, you can choose to use AdminStudio's native virtual conversion functionality to perform the conversion or you can choose to use the Microsoft App-V 5.0 Sequencer. When preparing to convert a package to App-V 5.0 format, you are required to select one of the following **Package Creation** methods (as described in Selecting the App-V Conversion Method):

Table 10-5 • App-V 5.0 Conversion Methods

Method	Description
App-V 5.x with AdminStudio	The conversion workflow for this method depends upon whether the package requires repackaging: • **Package requires repackaging**—The Windows Installer or legacy executable (.exe) package is copied to a VM snapshot, along with the Repackager files. The package is then repackaged, copied back to the AdminStudio machine, and then converted to App-V 5.0 format using the AdminStudio virtual converter. • **Package is a Windows Installer package and does not require repackaging**—Package is directly converted to App-V 5.0 using the AdminStudio virtual converter.
App-V 5.x with Sequencer	When using this method, the Windows Installer or legacy executable (.exe) package is copied to a virtual machine snapshot that has the Microsoft App-V 5.0 Sequencer installed on it. The App-V 5.0 Sequencer converts the package to App-V 5.0 format. The virtual package is then copied back to the AdminStudio machine.

Advantages / Disadvantages of Each Method

There are advantages of using each App-V 5.x **Package Creation** methods:

Table 10-6 • Advantages / Disadvantages of AdminStudio's App-V 5.x Package Creation Methods

Method	Advantages	Disadvantages
App-V 5.x with AdminStudio	• **Provides post-installation configuration option**—Enables you to perform both pre-installation and post-installation configuration tasks. (The **App-V 5.x with Sequencer** option only offers pre-installation configuration.) See Enabling Pre-Installation and Post-Installation Configuration.	

Table 10-6 • Advantages / Disadvantages of AdminStudio's App-V 5.x Package Creation Methods

Method	Advantages	Disadvantages
App-V 5.x with Sequencer	• **App-V 5.x with Sequencer method could be quicker**—When using this method, you are not required to copy the Repackager files to the virtual machine. Also, no intermediate Windows Installer package is built. Therefore, this method could perform conversion slightly quicker than the **App-V 5.x with AdminStudio** method. • **Uses Microsoft technology**—You may prefer this method if your organization prefers to use Microsoft technology to perform App-V 5.x conversion.	• **Installation directory must be known before sequencing can begin**—In order to sequence the application effectively, you must have detailed knowledge of the how the installation is supposed to work. Prior to beginning the sequencing process, you are required to specify the installation directory for the application being sequenced. This information is often not readily available, and may require you to open the installation in an editing tool, such as InstallShield, in order to find it, or to run the installation one time prior to sequencing.
Both Methods	• **Can perform App-V package testing prior to deployment**—Automated Application Converter includes a launch utility that allows you to launch and test the App-V package locally immediately after conversion before distributing them to the App-V Server. • **Bulk conversions**—You can use either App-V 5.x conversion method to perform bulk conversions of multiple applications in a directory hierarchy. The Automated Application Converter has both a user interface and a command line interface. ***Note*** *• To perform bulk conversion, you need to have purchased Automated Application Converter (Multiple Application Version).*	

App-V Conversion Methods Available in Repackager and InstallShield Editor

In addition to using Automated Application Converter, you can also use Repackager or the InstallShield Editor App-V Assistant perform conversion to App-V 4.x or 5.0 format. However, these tools offer only one App-V 5.x conversion method: **App-V 5.x with AdminStudio**.

Support for App-V 5.0 SP2 Shell Extension and Runtime Features

App-V 5.0 packages created by AdminStudio fully support the following new features of App-V 5.0 SP2:

- **Shell extensions**—App-V 5.0 packages created by AdminStudio support shell extensions, including:
  - Context menu handler
  - Drag-and-drop handler
  - Drop target handler
  - Data object handler
  - Property sheet handler
  - Infotip handler
  - Column handler
- **ActiveX controls**—App-V 5.0 packages created by AdminStudio support ActiveX controls, which are now registered and supported via the `AppxManifest.xml` file.
- **Browser helper objects**—App-V 5.0 packages created by AdminStudio support browser plug-ins, which aids scenarios where applications need to integrate with Internet Explorer.
- **Side-by-side (SxS) runtime dependencies**—App-V 5.0 packages created by AdminStudio support side-by-side runtime dependencies. App-V 5.0 SP2 automatically detects side-by-side assemblies and deployment on the computer running the App-V 5.0 SP2 client.
- **Full VFS write mode**—App-V 5.0 packages created by AdminStudio support the **Full VFS Write Mode** option, which gives a virtual application full write permissions to its virtual file system files and folders.

Note • *The `Full VFS Write Mode` feature was introduced in App-V 5.0 SP2 HotFix 4.*

Creating 64-Bit App-V Packages

AdminStudio supports converting 64-bit applications into Microsoft App-V 4.6 and 5.x package formats, which can be deployed on Windows 64-bit systems with Microsoft App-V 64-bit clients installed. This process can be a direct conversion from a 64-bit Windows Installer package or one involving repackaging on a 64-bit machine.

Note • *Automated Application Converter converts 64-bit packages to App-V 4.6 and converts non-64-bit packages to App-V 4.5 to increase the backwards compatibility of the package. 64-bit application support was the main new feature of App-V 4.6.*

Note • *App-V 4.5 packages created using earlier versions of AdminStudio can be made compatible with App-V 4.6. In some cases, it may be necessary to manually edit the `.osd` files to specify support for 64-bit operating systems.*

Important • *It is highly recommended that you perform the conversion of 64-bit Windows Installer packages to App-V packages on a Windows 64-bit machine. If you attempt conversion on a 32-bit Windows machine, it could result in a failure to*

extract COM information for 64-bit binaries. Also, in some cases, Windows Installer packages contain shortcuts that target executables not found in the package itself. If these shortcuts target executables found in 64-bit Windows folder locations, then these shortcuts will not be handled correctly on 32-bit machines.

Editing an OSD File to Make Advanced Changes to an App-V 4.x Package

Version • *This information applies to App-V 4.x packages.*

An `.osd` file is an XML-based file that describes an App-V 4.x package's shortcuts, file extensions, dependencies, and other data that can influence the environment of the application.

For advanced control over the information that is stored in the `.osd` files, you can edit the `.osd` file in an XML or text editor. For example, you may want to edit an `.osd` file directly in a text editor to specify the location of an `.sft` file, instead of configuring the location in the Automated Application Converter or on the Package Information page of the InstallShield Microsoft App-V Assistant. The following instructions explain how to do this. These instructions are for advanced users only.

To use a text or XML editor to edit an `.osd` file for making advanced changes, such as specifying the App-V server location for an `.sft` file, perform the following steps:

Task ***To use a text or XML editor to edit an .osd file:***

1. Open the OSD file using any XML or ASCII text editor—for example, Microsoft Notepad.

   Note • *Before modifying the `.osd` file, read the schema prescribed by the `.xsd` file in the install directory. Failing to follow this schema might introduce errors that prevent a sequenced application from starting successfully.*

2. Locate the `CODEBASE` element. Below is a sample `CODEBASE` element:

   ```
   <CODEBASE HREF="HTTP://%SFT_SOFTGRIDSERVER%:80/orca.sft" GUID="A895355A-5883-41C6-A144-
       1BDA12242AAA" PARAMETERS="" FILENAME="{A895355A-5883-41C6-A144-1BDA12242AAA}\Orca.exe"
       SYSGUARDFILE="{A895355A-5883-41C6-A144-1BDA12242AAA}\osguard.cp" SIZE="2555268"/>
   ```

3. Locate the `HREF` attribute of the `CODEBASE` element and enter a valid URL to the published location of that App-V package's `.sft` file.

Guidelines for Editing an .OSD File

When editing an `.osd` file, adhere to the prescribed schema and the following guidelines:

- Ensure that named elements are nested within the `<SOFTPKG>` root element.
- Ensure that element names are in all uppercase letters.
- Be aware that attribute values are case sensitive.
- Type carefully, and observe the XML specifications.

How Windows Services Are Integrated into an App-V Package

When you use the Automated Application Converter to convert a Windows Installer package to an App-V package, references to Windows services that are encountered are integrated into the App-V package. In a Windows Installer package, a Windows service may be indicated by either an entry in its `ServiceInstall` table or by a Registry entry for Windows services.

- **ServiceInstall table**—If a Windows Installer package's use of a Windows service is indicated by an entry in the `ServiceInstall` table, Automated Application Converter will convert that entry to a standard Registry entry for Windows services.
- **Registry entry**—If a Windows Installer package's use of a Windows service is indicated by a Registry entry for Windows services (perhaps as the result of being repackaged), Automated Application Converter does not need to make any changes to support the application's use of the Windows service within the virtual environment.

Start Up and Shut Down Sequences

If an App-V package has an associated Windows service, App-V will start up the Windows service first, in the virtual environment, and then start up the App-V package. You will see the Windows service start up in the Task Manager as a separate process, but App-V will be running the service within the virtual environment.

Upon shut down, App-V will first shut down the App-V package and then shut down the Windows service.

About VMware ThinApp Virtual Packages

You can use the Automated Application Converter to convert a Windows Installer package to a VMware ThinApp virtual application. Information about ThinApp applications is presented in the following topics:

- About ThinApp Applications
- Prerequisites for Building a ThinApp Application

Note • *You can also convert a Windows Installer package to a ThinApp application using InstallShield Editor's ThinApp Assistant. Using the ThinApp Assistant, you can configure a ThinApp application's Active Directory settings, files, folders, shortcuts, registry settings, isolation options, and build options. See Getting Started With Application Virtualization and the InstallShield Help Library for information on the ThinApp Assistant.*

Note • *For information on how to simultaneously build an InstallShield Editor project, a Windows Installer package, and a ThinApp application from your Repackager project, see Automatically Generating a Virtual Application During Repackager Project Build.*

About ThinApp Applications

VMware ThinApp is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine.

ThinApp applications can be deployed on a machine without modifying the local operating system or file system. They run in a "sandbox" (or virtual environment) which protects the local operating system from installation modifications that could affect stability or security. Also, ThinApp applications can be run safely from restricted user accounts without local installation.

Information about ThinApp applications is presented in the following sections:

- ThinApp Virtual Operating System
- Benefits of Deploying ThinApp Applications
- Components of a ThinApp Application

ThinApp Virtual Operating System

A ThinApp application runs in a virtual operating system—a small light-weight component which is embedded with each ThinApp application—that consists of a virtual file system and a virtual registry. When the ThinApp application is run, the virtual operating system environment is merged with the real system environment.

The virtual operating system technology enables entire applications to be packaged into a single `.exe` file that can be run without an installation process, and without modifying the resident operating system.

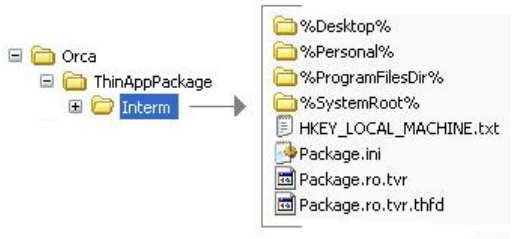
A ThinApp application can be run from a network or offline on the local machine.

Components of a ThinApp Application

When you use Automated Application Converter, Repackager, or InstallShield Editor to build a ThinApp virtual package, the resources you generate are called **ThinApp applications**.

When package conversion is complete, a `Conversion completed` message appears in the **Output** window and the path to the generated ThinApp application is listed, such as:

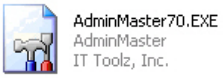
```
C:\AdminStudio Shared\My Application\ThinAppPackage
```



The ThinApp application is created in a folder named `ThinAppPackage` that is created in the same directory as the Windows Installer package you converted.

The number of files included in a ThinApp application depends upon how many shortcuts are defined:

Table 10-7 • Components of a ThinApp Application

Number of Shortcuts	ThinApp Application Components	Description
1 shortcut	`ProductName.exe` 	The ThinApp application consists of a single executable (`.exe`) file: • **Launching the application**—This executable file is used to launch the ThinApp application. • **Location of application data**—This executable file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

More than 1 shortcut	`ProductName.exe`
	`FeatureName.exe`
	`Package.DAT`
	
		The ThinApp application consists of two or more executable files and a `Package.DAT` file:
		• **Launching the application**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

Metadata File	`metadata.ami`	A file created during AdminStudio 9.0+ package conversion that contains metadata identifying the original Windows Installer package that was used to create the virtual package. ***Note*** • *Because of this file, you are able to import this virtual package into the Application Catalog and associate it with its source Windows Installer package.*

Caution • *Modifying these files directly is not recommended. To make any modifications, use the InstallShield ThinApp Assistant.*

Intermediate Data Files: Interm Directory

When a ThinApp application is built, files that support the ThinApp application build process are extracted out of the Windows Installer package and saved in a subdirectory of the `ThinAppPackage` directory named the `Interm` directory.

Figure 10-3: Interm Subdirectory of the ThinAppPackage Directory

The data in this directory is then compiled into ThinApp application as part of the build process. The data in the `Interm` directory *does not* need to be distributed with the ThinApp application.

Benefits of Deploying ThinApp Applications

Deploying ThinApp applications provides the following benefits:

- **Reduces time to deployment and costs associated with testing**—Applications can be deployed and run in independent sandboxes, eliminating the need for expensive and time-consuming multi-application regression testing. This reduces the time to deployment and the costs associated with testing.
- **Fast, lightweight virtualization** —ThinApp does not use emulation, so all processes are executed natively at full speed.
- **Reduces the cost of maintaining secure locked-down desktops**—ThinApp applications can run in restricted user accounts without requiring any host modifications.
- **Enhances work-force mobility, business continuity and disaster recovery**—ThinApp applications can be run off-line, directly from any external media including USB Flash, CD-ROM, and off-line laptops.
- **No infrastructure changes needed**—ThinApp applications can be deployed using any existing software deployment systems including Active Directory and SMS. ThinApp has no client or server components to manage or maintain and ThinApp can transparently stream large applications from any network attached storage devices without server software.
- **Sandboxing prevents modifications**—ThinApp redirects all changes intended for the host computer's file system and registry to a private per-user sandbox. Sandboxes can be located on a network share, allowing application settings to follow users as they move from machine to machine. For mobile users, sandboxes can be stored on local USB flash drives, thus preventing damage to the host computer or accidental host storage of sensitive data.

Prerequisites for Building a ThinApp Application

AdminStudio will convert the package installation into a format compatible with VMware ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools.

As a prerequisite to building a ThinApp application from AdminStudio, you must have installed VMware ThinApp **and accepted any and all license agreements**.

Caution • *If you install ThinApp but you have not yet accepted the license agreement, the build process will fail. For more information, see the VMware website.*

About Citrix XenApp Virtual Packages

You can use the Automated Application Converter to convert a Windows Installer package to a Citrix profile for deployment on Citrix XenApp.

Information about using the Automated Application Converter is presented in the following topics:

- About Citrix XenApp and Citrix Profiles
- Benefits of Deploying Citrix XenApp Profiles

Note • *You can also convert a Windows Installer package to a Citrix profile using InstallShield Editor's Citrix Assistant. Using the Citrix Assistant, you can configure a Citrix profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options. See Getting Started With Application Virtualization and the InstallShield Help Library for information on the Citrix Assistant.*

Note • *For information on how to simultaneously build an InstallShield Editor project, a Windows Installer package, and a Citrix profile from your Repackager project, see Automatically Generating a Virtual Application During Repackager Project Build.*

About Citrix XenApp and Citrix Profiles

Citrix XenApp is an application delivery system for Windows applications. When you use Repackager or InstallShield Editor to prepare a Windows Installer package for deployment on Citrix XenApp, the resources you generate are called **profiles**.

Overview information about Citrix XenApp and Citrix profiles is presented in the following topics:

Table 10-8 • Overview of Citrix XenApp

Topic	Description
About Citrix XenApp	Provides an overview of how Citrix XenApp works and provides a diagram illustrating application delivery.
About Citrix Profiles (.profile)	Lists the files and directories that comprise a Citrix profile.

About Citrix XenApp

Citrix XenApp is an application delivery system for Windows applications that offers both application virtualization and application streaming. Applications are centralized on Citrix XenApp and then those applications are deployed to users throughout the enterprise. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine.

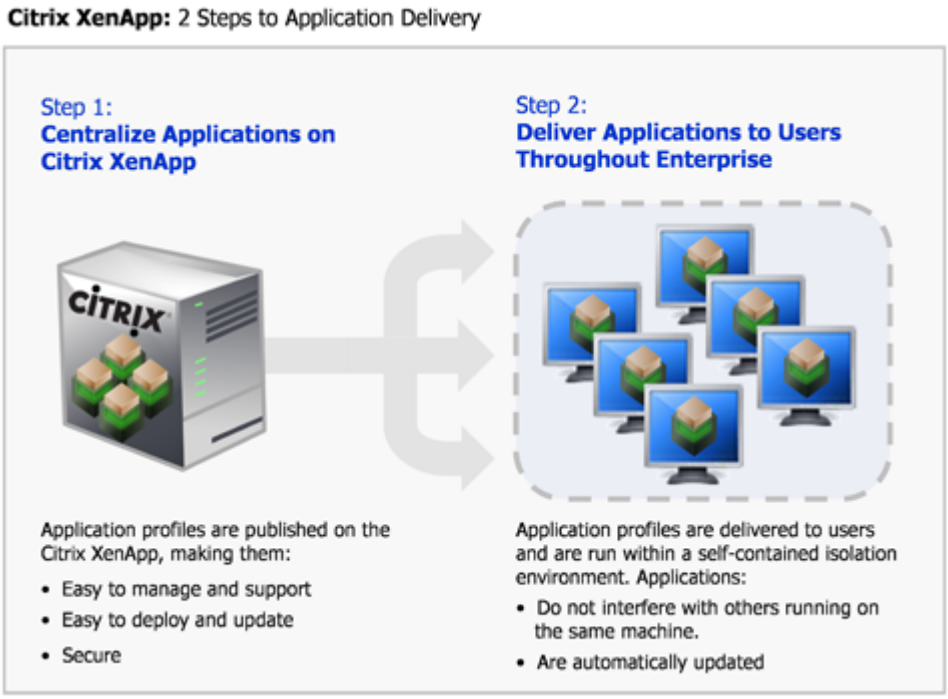


Figure 10-4: Citrix XenApp: Two Steps to Application Delivery

When applications are deployed on a Citrix XenApp, users can run those applications in an isolation environment, without installing, while connected or offline. Applications behave just like they were installed locally, but without any of the problems of installation, such as interfering with other applications on the same device. Files are saved locally and individual settings are preserved. Every time the application is run, it checks for errors or updates and they are delivered automatically.

Note • *For more information, see Benefits of Deploying Citrix XenApp Profiles.*

About Citrix Profiles (.profile)

When you use Automated Application Converter, Repackager, or InstallShield Editor to prepare a Windows Installer package for deployment on Citrix XenApp, the resources you generate are called **profiles**.

When package conversion is complete, the Automated Application Converter, Repackager, or InstallShield Editor displays the path to the generated virtual package, such as:

```
C:\AdminStudio Shared\MyPackage\CitrixProfile\MyPackage.profile
```

These files are saved in a subfolder of a folder named `CitrixProfile` that is created in the same directory as the Windows Installer package you converted. The profile, which is published on Citrix XenApp, consists of the following:

Figure 10-5: Profile Files and Directories

A profile contains the following files and directories:

Table 10-9 • Components of an Application Profile

Component	Name	Description
Profile Manifest File	`myapp.profile`	An XML file that defines the profile.
CAB File	`[alphanumeric_string].cab`	Compressed cabinet file that provides the isolation environment contents for the application.
Hashes File	`Hashes.txt`	Hash key file for digital signatures and signing profiles.
Icons File	`Icons.bin`	Icons repository.
Scripts Folder	`Scripts`	Folder containing any pre- launch or post-exit scripts that you have chosen to include.
Metadata File	`metadata.ami`	A file created during AdminStudio 9.0+ package conversion that contains metadata identifying the original Windows Installer package that was used to create the virtual package. ***Note*** *• Because of this file, you are able to import this virtual package into the Application Catalog and associate it with its source Windows Installer package.*

Caution *• Modifying these files directly is not recommended. To make any modifications, use the InstallShield Citrix Assistant.*

A profile can contain a single application or suite of applications.

Benefits of Deploying Citrix XenApp Profiles

Converting a Windows Installer package to a Citrix profile and deploying it on a Citrix XenApp offers the following benefits:

- Reduces Application Conflicts

- Enables Rapid, Low Cost Application Deployment
- Enables Automatic Software Updates
- Centralized Application Management Provides Controlled Access and Security
- Enables User-Based Application Access Rather Than Machine-Based Access

Reduces Application Conflicts

Traditionally to deploy an application throughout an enterprise, the application was installed on each user's desktop. Therefore, prior to installation, each application had to be tested for conflicts against each target desktop image (operating system with existing applications). After resolving conflicts that were found during testing, each application then had to be installed on each desktop. This process was very time consuming not only during initial installation, but also when applying patches or upgrading.

Citrix profiles run within isolation environments, which separate the interaction between an application and the underlying operating system's resources in order to prevent the applications from interfering with others running on the same machine. Because applications do not interact, the need to perform any conflict analysis and regression testing prior to deployment is eliminated. This not only results in rapid application deployment, but it also reduces the total cost of application delivery, due to decreased labor by IT.

Also, because users running applications in an isolation environment encounter no conflicts with other applications, user calls to the help desk are decreased.

Enables Rapid, Low Cost Application Deployment

Deploying Citrix profiles on Citrix XenApp simplifies the deployment of new applications, updates and patch deployment, regardless of the diversity of the access devices, software languages, computing architectures, and networks that are involved.

- **Only a single instance of the application is installed**—Instead of deploying, managing, updating and securing a vast array of heterogeneous client software on each individual user's access device, a single instance of the application is installed on Citrix XenApp. The IT department only has to test for one environment, and deploy and update in one place. This reduces the cost of application installation and support. Also, you can deploy a Citrix profile once on a Citrix XenApp and replicate it to other Citrix XenApps within the existing enterprise infrastructure.
- **Prevents application-specific server silos**—Deploying applications on Citrix XenApp prevents the build-up of application-specific server silos because you can safely install and reliably run multiple application versions and incompatible applications on the same server.
- **Enables you to quickly install and update software throughout your enterprise**—Because you can manage the delivery of all of your Windows-based applications from one centralized location, there is no need to go from desktop to desktop, travel from office to office, or wait for laptops to return to headquarters in order to install or update software. With Citrix XenApp, you can deliver applications and updates instantly anywhere, any time—to offshore employees, outsourcers, new branch offices, new mergers and acquisitions, and mobile workforces.

Enables Automatic Software Updates

When an upgrade or patch needs to be deployed, you would only need to update the Citrix profile on Citrix XenApp, which will then automatically update all of the instances of that Citrix profile throughout the enterprise. This means that users always have the latest application updates and patches, automatically.

Centralized Application Management Provides Controlled Access and Security

With Citrix XenApp, you can centralize applications and data in secure data centers, which increases data security and ensures fast, reliable performance. Centralized application management using Citrix XenApp provides the following benefits:

- **Enhances security**—Enables you to control, protect, and retain intellectual property centrally to reduce the chance for data loss and theft. Citrix XenApp helps you prevent data from leaving the data center without your explicit permission, which supports regulatory compliance and security objectives. You can provide authorized access to appropriate users—such as employees, customers, and partners—while verifying the ongoing security of the environment.
- **Can provide managed access to applications to users outside of your organization**—You can standardize the use of applications, without having to standardize the machines that the applications use. This enables you to provide managed access to applications from computers that are not your own corporate assets, such as from contractor or consultant computers.
- **Monitors application usage and performance**—Citrix XenApp gives you end-to-end visibility into application usage and performance. It gives IT administrators the power to understand who is using what, how often, and to what extent. They can observe, monitor, measure, audit, report and archive all the dimensions of information flow throughout the computing environment. This enables informed decisions regarding application consolidation and retirement, capacity planning, service level agreements and departmental charge-back.
- **Enables identity-driven access**—Citrix XenApp enables you to provide identity-driven access tailored to any user environment. It automatically analyzes the user's permissions and then delivers the appropriate level of access to applications without compromising security. Depending on who and where users are and what device and network they're using, they may be granted different levels of access. You can also easily "decommission" applications by simply turning off a user's permission to it.

Enables User-Based Application Access Rather Than Machine-Based Access

Users can access their applications anywhere on the network, regardless of where they are or what device they are using.

About the Automated Application Converter

The AdminStudio Automated Application Converter combines the functionality of the Windows Installer Batch Converter with the additional capability to automatically repackage and convert Windows Installer packages, as well as setups in other formats, into virtual applications. You can also choose to automatically repackage setups into Windows Installer packages.

Information about the Automated Application Converter is presented in the following sections:

- Benefits of Using the Automated Application Converter
- Automated Application Converter Workflow Diagram
- Supported Operating Systems
- Supported Virtual Machines

Benefits of Using the Automated Application Converter

Previously, when converting a Windows Installer package to a virtual application, there were cases when you needed to capture its installation prior to being able to perform a successful conversion. Repackaging is sometimes required because it is not possible to determine the run-time behavior of certain Windows Installer package elements—such as conditional components and custom actions—without actually running the install. While converting a Windows Installer package to a virtual application is automated and is a batch process, repackaging setups is a manual process requiring a packager to individually repackage each setup on clean machines and then to convert them into virtual applications—which is a time consuming task requiring several hours of a packagers time.

The Automated Application Converter examines a group of setups to automatically determine which need to be repackaged and which can be virtualized without repackaging. It converts the setups that can be directly converted and then automatically repackages the others by launching virtual machines, running the setups, and capturing them prior to converting them into the target virtual formats.

The Automated Application Converter provides the following benefits:

- **Automated repackaging on virtual machines**—The Automated Application Converter provides an interface to provision and manage virtual machines, silently repackage installs on them, and create virtual packages for the resulting MSIs.
- **Ability to repackage non-MSI setups**—You can use the Automated Application Converter to repackage legacy (non-MSI) setups to create an MSI package that can be converted to a virtual package.

  Note • *These setups must support silent install mode.*
- **Conversion of packages from multiple sources**—Using the Automated Application Converter, you can convert setups from multiple sources:
  - AdminStudio Application Catalog
  - Specified directory or file
- **Efficiently manages repackaging queue on multiple virtual machines**—The Automated Application Converter efficiently manages the virtual machine queue, allowing setups to be packaged simultaneously on multiple machines.
- **Easy-to-understand reports**—The Automated Application Converter generates easy-to-understand HTML reports for each conversion run, providing detailed information on each package.
- **Easy-to-read progress indicators with one-click access to virtual machines**—The Automated Application Converter provides dashboard-type progress indicators with one-click access to open a virtual machine in a Remote Desktop session, enabling you to view the progress of a repackaging session and to quickly perform troubleshooting.
- **Provides option to use App-V 5.0 Sequencer when converting to App-V 5.0 format**—When converting a package to Microsoft App-V 5.0 format, you can choose to either use AdminStudio's virtualization technology to perform the conversion or to use the App-V 5.0 Sequencer, Microsoft's native technology. For more information, see Comparison of the App-V 5.0 Conversion Methods.

Automated Application Converter Workflow Diagram

You can use the Automated Application Converter to examine a group of setups to automatically determine which need to be repackaged and which can be virtualized without repackaging. It converts the setups that can be directly converted and then automatically repackages the others by launching virtual machines, running the setups, and capturing them prior to converting them into the target virtual formats.

The following diagram provides an overview of the AdminStudio Automated Application Converter workflow including:

- Input sources
- Virtualization readiness check
- Automated repackaging on virtual machines
- Conversion to virtual packages
- Output types

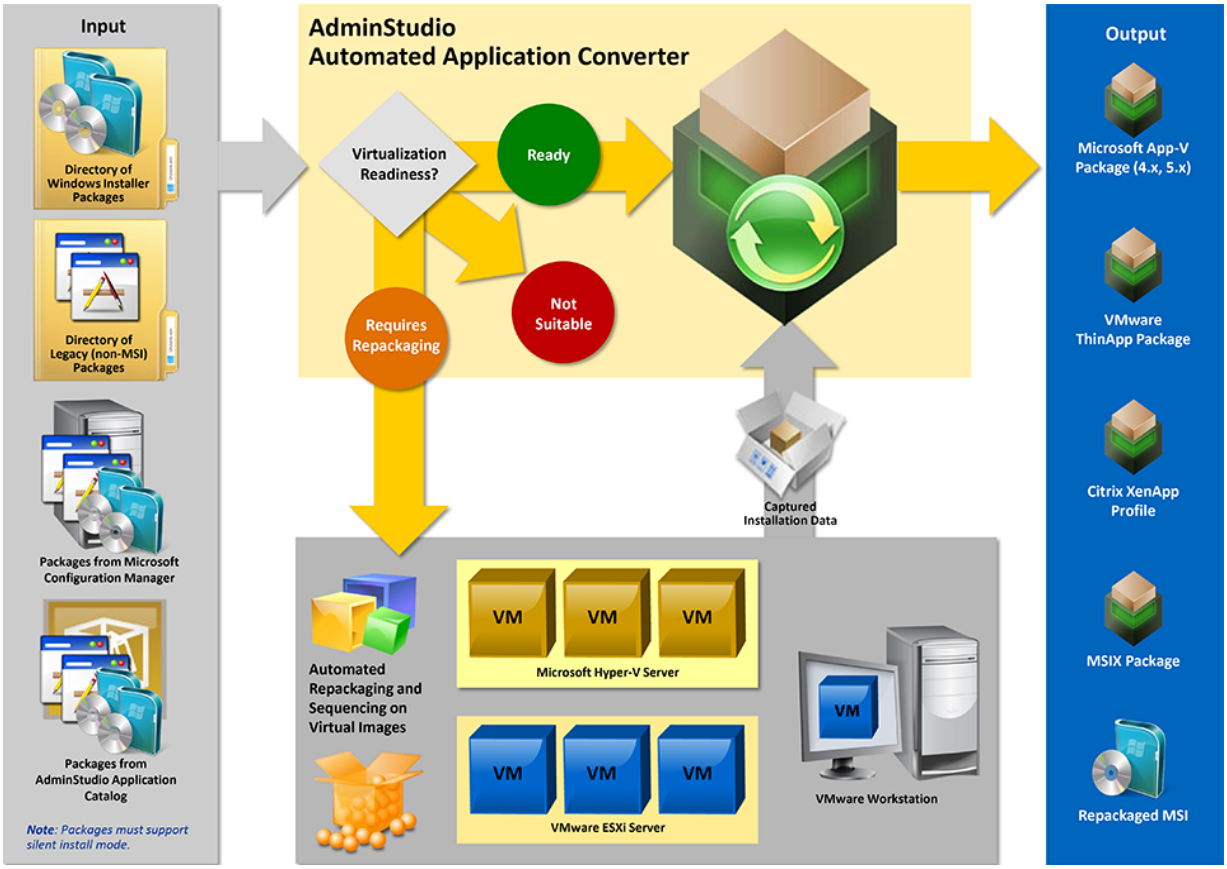


Figure 10-6: AdminStudio Automated Application Converter Workflow Diagram

Supported Operating Systems

The Automated Application Converter supports the following operating systems:

- Windows Vista (32-bit and 64-bit)

- Windows 8.1 (32-bit and 64-bit)
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Windows 11 21H2 (64 bit)
- Windows 10 21H2 (32 bit and 64 bit)
- Windows 10 21H1 (32 bit and 64 bit)
- Windows 10 20H2 (32 bit and 64 bit)
- Windows 10 2004 (32 bit and 64 bit)
- Windows 10 1909 (32 bit and 64 bit)
- Windows 10 1809 (and 2019 LTSC) (32 bit and 64 bit)

Supported Virtual Machines

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- Microsoft Hyper-V Server
- VMware ESX or ESXi Server
- VMware Workstation 6.5 or later

Launching the Automated Application Converter

The Automated Application Converter can be launched by doing either of the following:

- On the Windows Start Menu, point to **All Programs**, **AdminStudio**, **AdminStudio Tools**, and click **Automated Application Converter**.
- Launch **Repackager**, and then click the **Automatically Repackage Installations on Your Virtual Machines** link on the Repackager Home Page.

The **Open Project** panel of the Application Conversion Project Wizard opens.

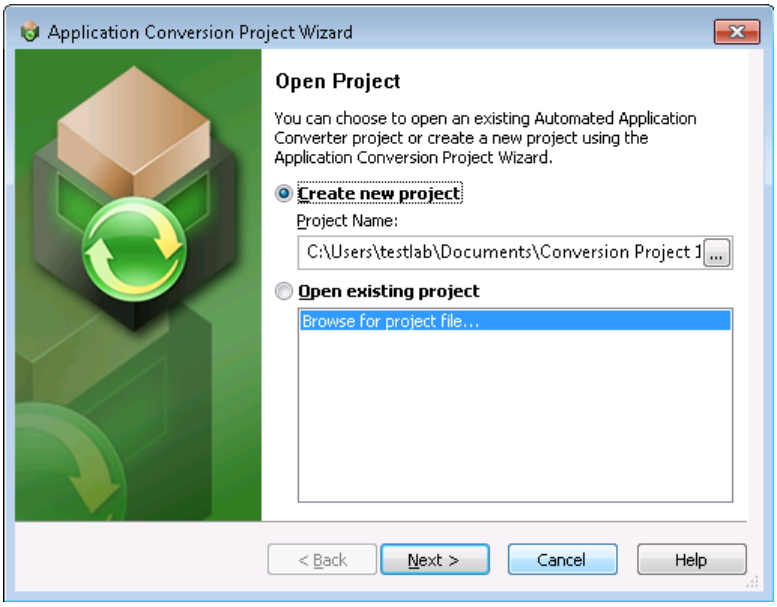


Figure 10-7: Open Project Panel of Application Conversion Project Wizard

On the **Open Project** dialog box, you can choose to create a new project or open an existing project. For more information, see Opening a Project.

Getting Started With the Automated Application Converter

The quickest way to get started using the Automated Application Converter is to use the end-to-end **Application Conversion Project Wizard**, which takes you through the three main steps in automated batch virtualization: selecting the packages to convert, selecting the virtual machines to use for repackaging, and converting the selected packages. See Using the Application Conversion Project Wizard to Perform an End-to-End Conversion for instructions.

You can also choose to perform each of these tasks separately by using one of the other three wizards that are provided:

Table 10-10 • Automated Application Converter Wizards

If you want to ...	Use this wizard ...	Description and Purpose
Add virtual machines	**Virtual Machine Import Wizard**	Add virtual machines to use to perform automated repackaging to Windows Installer packages. See Adding Virtual Machines Using the Virtual Machine Import Wizard.
Add packages	**Package Import Wizard**	Add packages from an AdminStudio Application Catalog or from a local or network file system. See Adding Packages from an AdminStudio Application Catalog and Adding Packages from a Local Machine or Network.

Table 10-10 • Automated Application Converter Wizards

If you want to ...	Use this wizard ...	Description and Purpose
Virtualize or repackage packages	**Application Conversion Wizard**	Virtualize packages to the virtual formats you specify. You can also perform repackaging. See Using the Application Conversion Wizard to Perform Automated Package Conversion.

Opening a Project

When you launch the Automated Application Converter, the **Open Project** dialog box opens, prompting you to either create a new project or open an existing project.

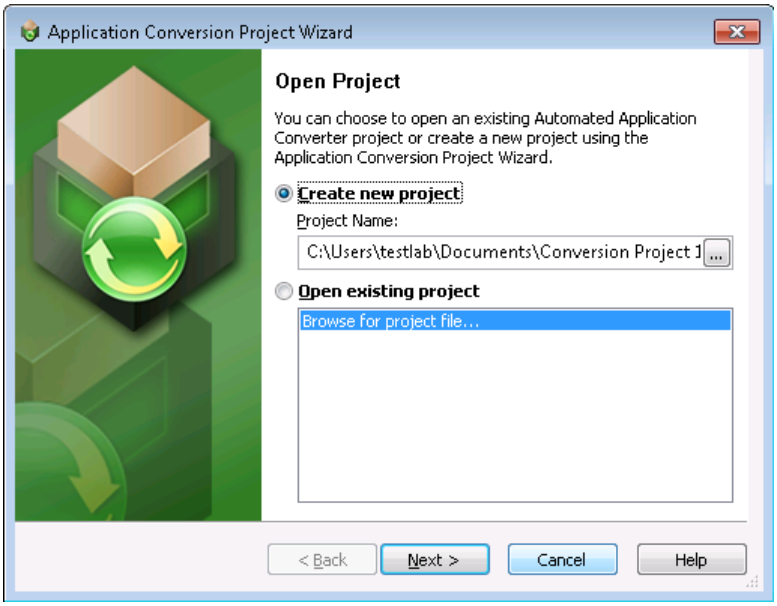


Figure 10-8: Open Project Dialog Box

The following procedures explain how to create a new project or open an existing project.

- Creating a New Project
- Opening an Existing Project

Creating a New Project

To create a new project, perform the following steps.

Task ***To create a new project:***

1. Do one of the following depending upon whether the Automated Application Converter is open:
   - **Not open**—Launch the Automated Application Converter.

- **Open**—On the **File** menu, click **New Project**.

The **Open Project** dialog box opens.

2. Select **Create new project**.
3. Click the Browse button next to the **Project Name** field. The **Save As** dialog box opens.
4. Enter a name (with an `.aacx` extension) and location for the new project file and click **Save**. The new project name is now listed in the **Project Name** box.
5. Click **Next**. The **Select Package Source** panel of the Application Conversion Project Wizard opens.
6. Continue with the steps in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion.

Opening an Existing Project

To open an existing project, perform the following steps.

Task ***To open an existing project:***

1. If the Automated Application Converter is not open, perform the following steps:
   a. Launch the Automated Application Converter. The **Open Project** dialog box opens.
   b. Select **Open existing project**.
   c. From the list, either select a project name or select **Browse for project file**.
   d. Click **Finish**. One of the following occurs:
      - **If you selected an existing project** from the list, the project opens in the Automated Application Converter interface.
      - **If you selected Browse for project file**, the **Open** dialog box opens. Select a project file and click **Open**. The project opens in the Automated Application Converter interface.
2. If the Automated Application Converter is open, perform the following steps:
   a. On the **File** menu, click **Open**. The **Open** dialog box opens.
   b. Browse to the project file you want to open and click **Open**. The project opens in the Automated Application Converter interface.

Note • *If you had unsaved changes in the project file that was already open, you will be prompted to save those changes prior to opening the new project file.*

Using the Application Conversion Project Wizard to Perform an End-to-End Conversion

When using Automated Application Converter to perform conversion, you need to perform the following tasks:

- **Task 1**: Add virtual machines to use during conversion

- **Task 2**: Add packages to convert
- **Task 3**: Perform conversion

You can perform these tasks all at once using one wizard (**Application Conversion Project Wizard**) or perform these tasks separately using three different wizards:

- **Virtual Machine Import Wizard**
- **Package Import Wizard**
- **Application Conversion Wizard**

The following instructions explain how to use the **Application Conversion Project Wizard** to perform these three tasks using the same wizard.

Note • *For instructions on how to perform these conversion tasks separately, see:*

- *Adding Virtual Machines Using the Virtual Machine Import Wizard*
- *Adding Packages from an AdminStudio Application Catalog* or *Adding Packages from a Local Machine or Network*
- *Performing a Conversion Using the Application Conversion Wizard*

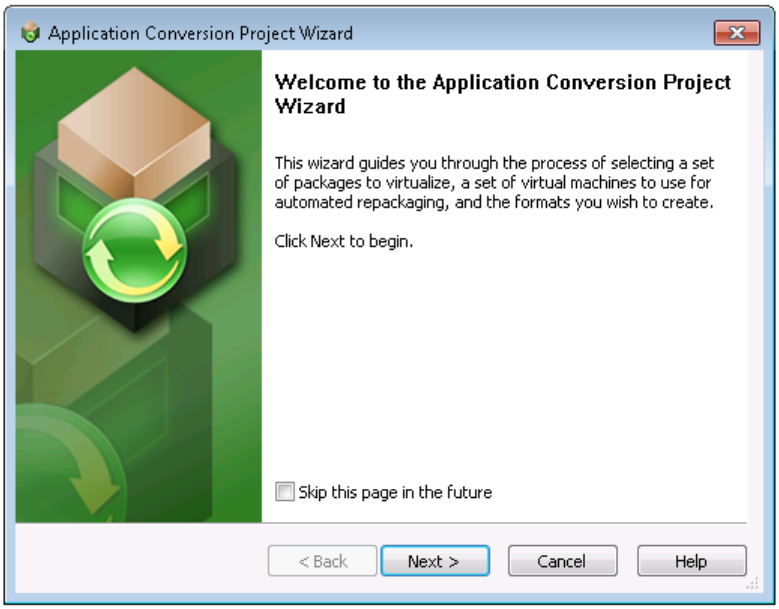
To get started using the **Application Conversion Project Wizard**, perform the following steps:

Task ***To get started using the Application Conversion Project Wizard:***

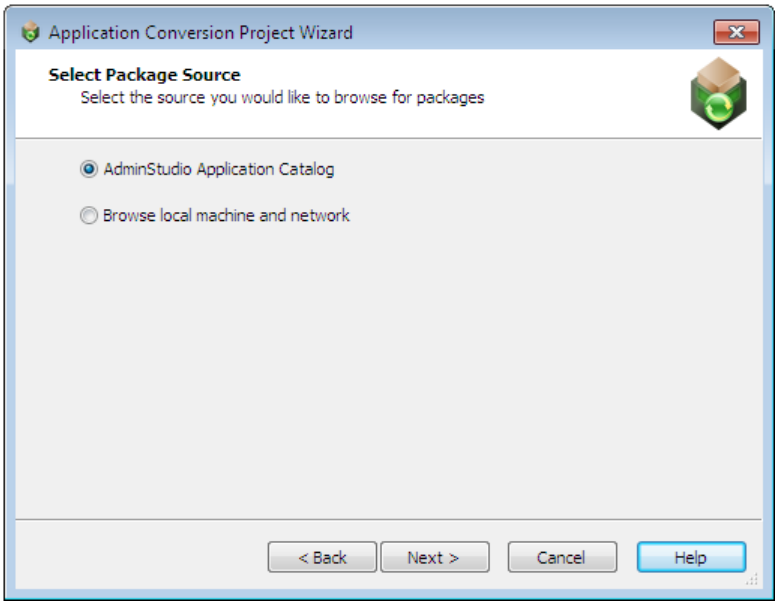
1. Perform the steps in Preparing Your Virtual Machines for Use With the Automated Application Converter to prepare your virtual machines to use for automated repackaging.
2. Launch the Automated Application Converter by doing one of the following:
   - On the Windows Start Menu, point to **All Programs**, **AdminStudio**, **AdminStudio Tools**, and click **Automated Application Converter**.
   - Launch **Repackager**, and then click the **Automatically Repackage Installations on Your Virtual Machines** link on the Repackager Home Page.

   The Automated Application Converter opens.

   Note • *See Automated Application Converter User Interface for more information.*
3. Create a new project or open an existing project, as described in Opening a Project.
4. On the **Tools** menu, click **Project Wizard** (or click the icon in the toolbar). The **Welcome to the Application Conversion Project Wizard** panel opens.



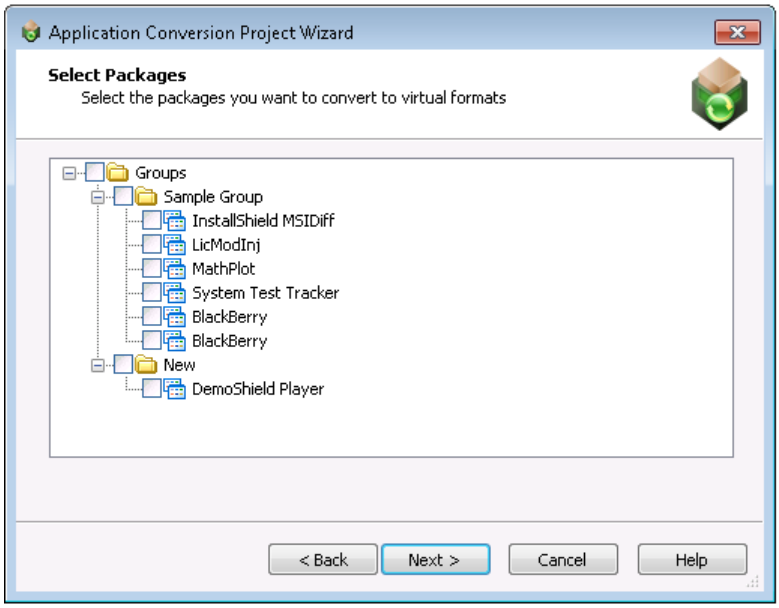
5. Click **Next**. The **Select Package Source** panel opens, prompting you to select the source that contains the packages you want to convert.



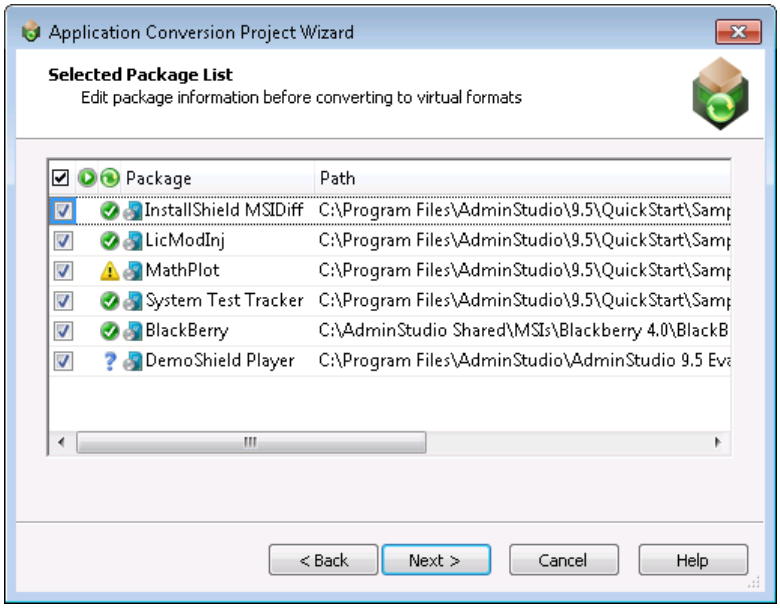
6. Select one of the following options:

Option	Description
AdminStudio Application Catalog	Select this option to connect to an AdminStudio Application Catalog and add all of the installations in that catalog to the list of packages to convert. If you select this option, the Connect to an AdminStudio Application Catalog panel opens, prompting you to login to an Application Catalog. ***Tip*** • *To select packages from Microsoft Configuration Manager to convert, first import those packages into the Application Catalog, as described in Importing From ConfigMgr (Formerly called as System Center Configuration Manager).*
Browse local machine and network	Select this option to browse a local or network machine to add installations to the list of packages to convert. If you select this option, the **Select Packages** panel opens, where you are prompted to select an installation file or a directory of installation files to add to the list of packages to convert. ***Note*** • *For information on the rules that the Automated Application Converter uses to determine which packages in the selected directory's subdirectories would be added to the list on the `Selected Package List` panel, see Automated Application Converter's Selection Rules When Adding Packages from a Directory.*

If you connected to an Application Catalog, the **Select Packages** panel opens.



When you have finished this step, packages will be listed and selected on the **Selected Package List** panel, and an icon indicates each package's virtualization readiness status.

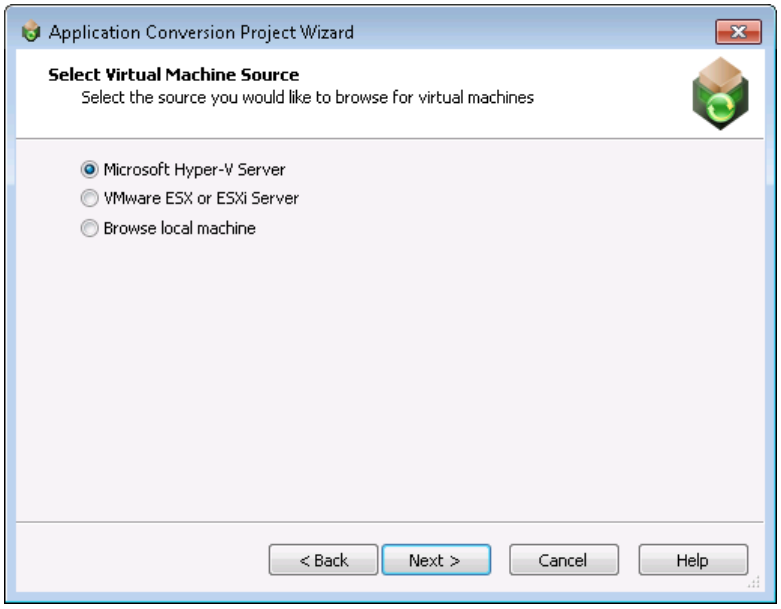


One of the following icons is listed in each package's Virtualization Readiness ( ) column:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.

Icon	Meaning	Description
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

7. Click **Next**. The **Select Virtual Machine Source** panel opens, prompting you to select the type of virtual machine that you are going to use for automated repackaging.

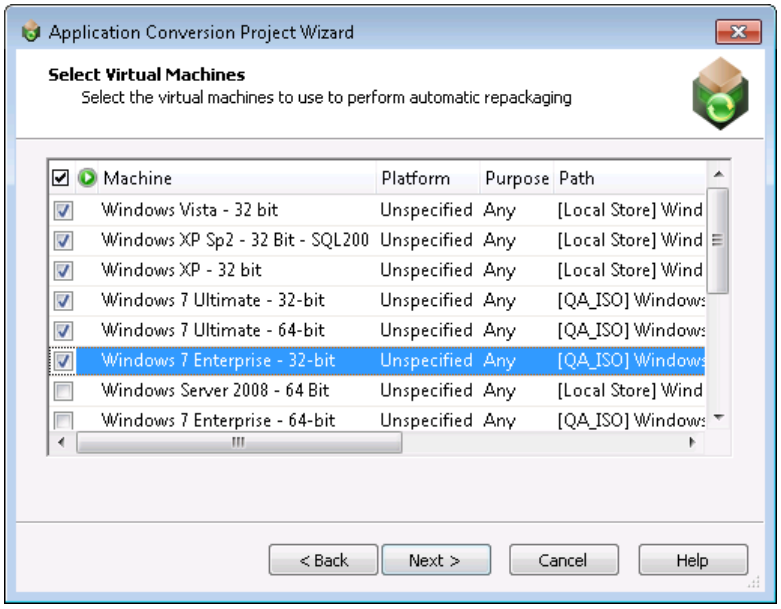


Note • *If none of the packages selected on the* ***Selected Package List*** *panel require repackaging in order to be converted into a virtual package, the* ***Select Virtual Machine Source*** *panel will not be displayed. Instead, the* ***Initial Configuration Complete*** *panel will open.*

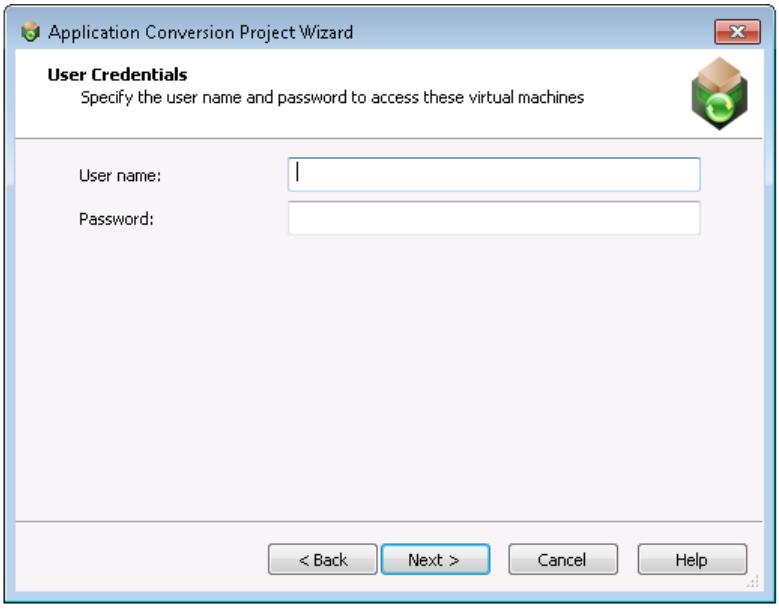
Select one of the following options and click **Next**:

Option	Description
Microsoft Hyper-V Server	Select this option to connect to a Microsoft Hyper-V Server. You will then be prompted for login information on the Select Virtual Machines from a Microsoft Hyper-V Server panel.
VMware ESX or ESXi Server	Select this option to connect to a VMware ESX or ESXi Server. You will then be prompted for login information on the Select Virtual Machines from a VMware ESX or ESXi Server panel.
Browse local machine	Select this option to connect to a VMware Workstation virtual image installed locally. The Select Virtual Machines opens, where will be prompted to select either a VMware Workstation image or directory of images.

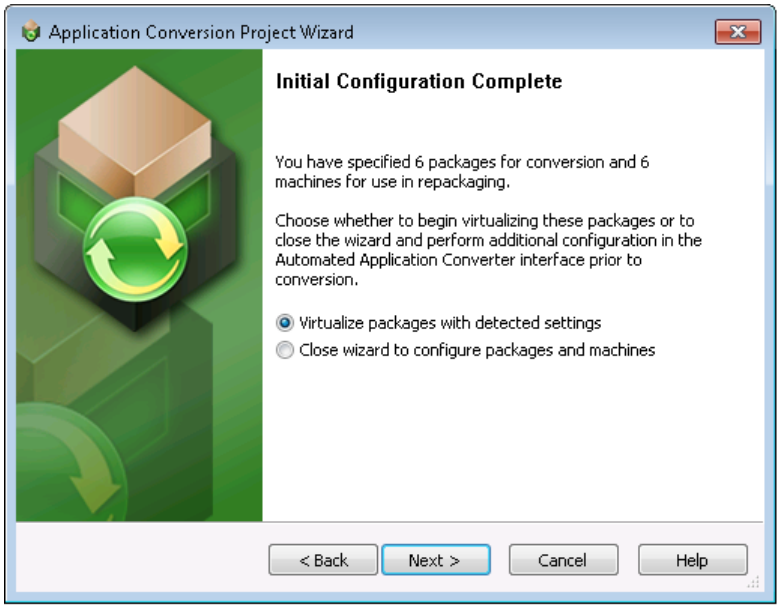
When you have finished this step, the virtual machines will be listed (but not selected) on the **Select Virtual Machines** panel.



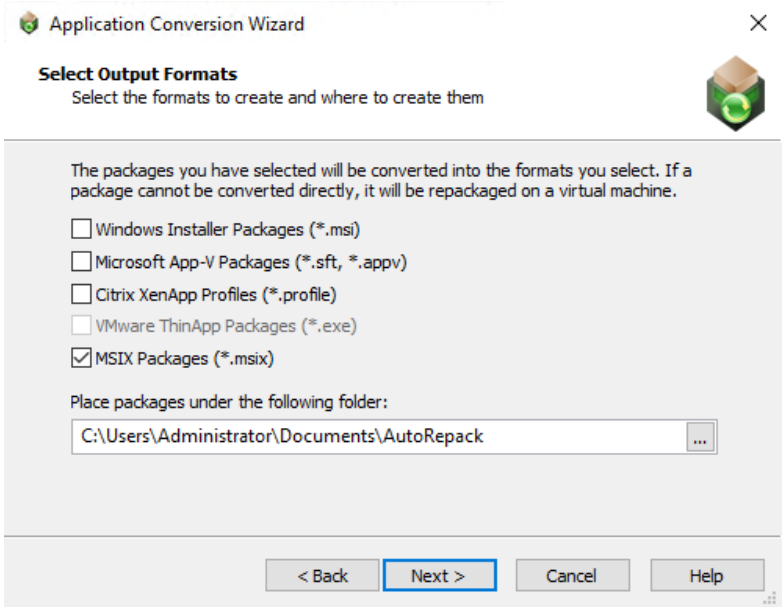
8. On the **Select Virtual Machines** panel, select the virtual machine images that you want to use to perform automated repackaging.
9. For each selected image, click in the **Platform** column and identify its platform.
10. Optionally, if you want to limit the use of a virtual machine to either repackaging only or testing only, click in the **Purpose** column and select **Repackaging** or **Testing** from the list. The default value is **Any**.
11. Click **Next**. The **User Credentials** panel opens, prompting you to specify the login credentials to use to access the selected virtual machines.



12. Enter login credentials and click **Next**. The **Initial Configuration Complete** panel opens, listing a summary of your selections, and prompting you to select whether you want to begin to **Virtualize packages with detected settings** or to **Close wizard to configure packages and machines**.



13. Select **Virtualize packages with detected settings** and click **Next**. The **Select Output Formats** panel opens, prompting you to select one or more output formats:

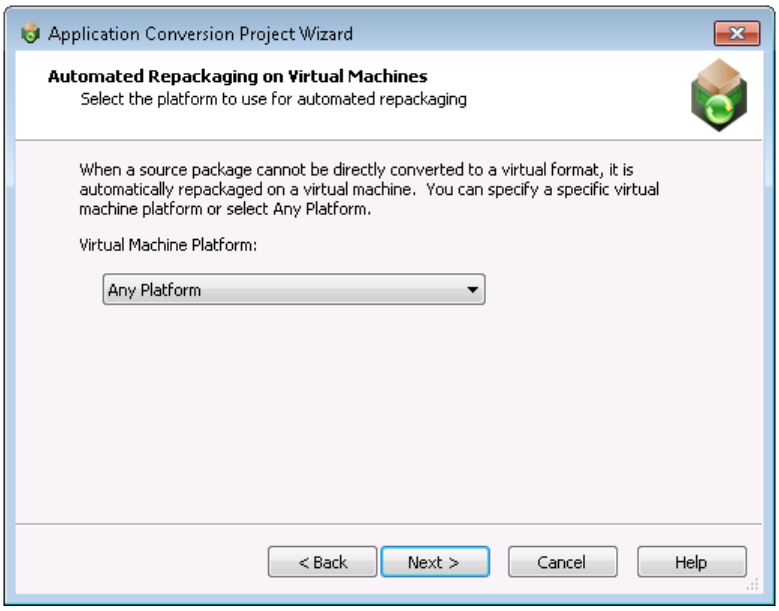


Note • *If you have selected Windows Installer packages on the* ***Selected Package List*** *panel, but those packages do not require repackaging prior to virtualization, the* ***Windows Installer Package (*.msi)*** *option on the* ***Select Output Formats*** *panel will be disabled. If you want to force the Automated Application Converter to repackage that package, return to the* ***Selected Package List****, click in that package's Virtualization Readiness column and select* ***Requires repackaging*** *from the list.*

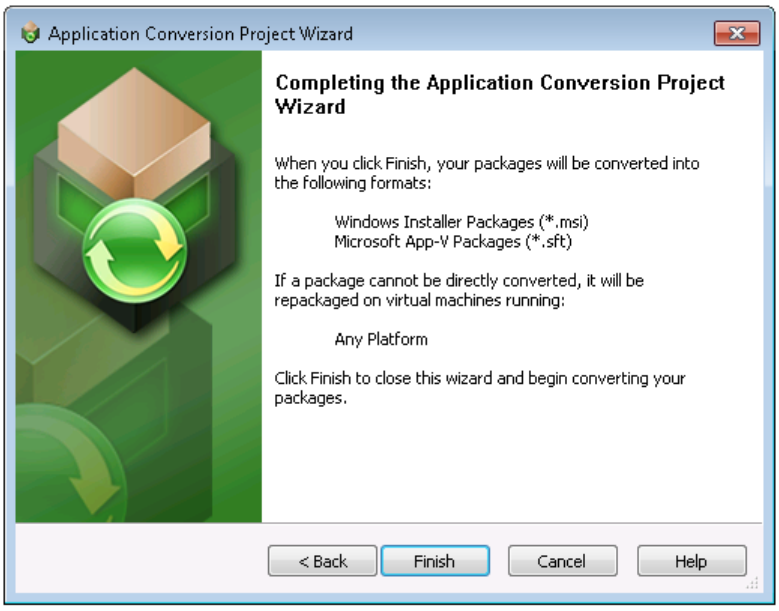
14. On the **Select Output Formats** panel, select one or more of the following formats:

Option	Description
Windows Installer Packages (*.msi)	Select this option to repackage the selected packages into Windows Installer packages (.msi).
Microsoft App-V Packages (*.sft)	Select this option to convert the selected packages to Microsoft App-V virtual applications. ***Note*** *• If you select this option, your packages will be converted using the* ***Package Creation*** *method (described in Comparison of the App-V 5.0 Conversion Methods) that is selected on the* ***Project Options*** *dialog box:* • *App-V 4.6 with AdminStudio* • *App-V 5.x with AdminStudio* • *App-V 5.x with Sequencer* *However, if a method is selected in the* ***Package Creation*** *field of a package's* ***Properties*** *window, the selected method will be used.*
Citrix XenApp Profiles (*.profile)	Select this option to convert the selected packages to Citrix XenApp profiles.
VMware ThinApp Packages (*.exe)	Select this option to convert the selected packages to VMware ThinApp virtual applications.
MSIX Packages (*.msix)	Select this option to convert the selected packages to **MSIX** packages. ***Note*** *• If you select this option, either one of the following must be selected.* • *Certificate File (.pfx)* • *Certificate Store* as shown in Editing Package Properties on the Packages Tab **>>** Setting MSIX Signing Options.

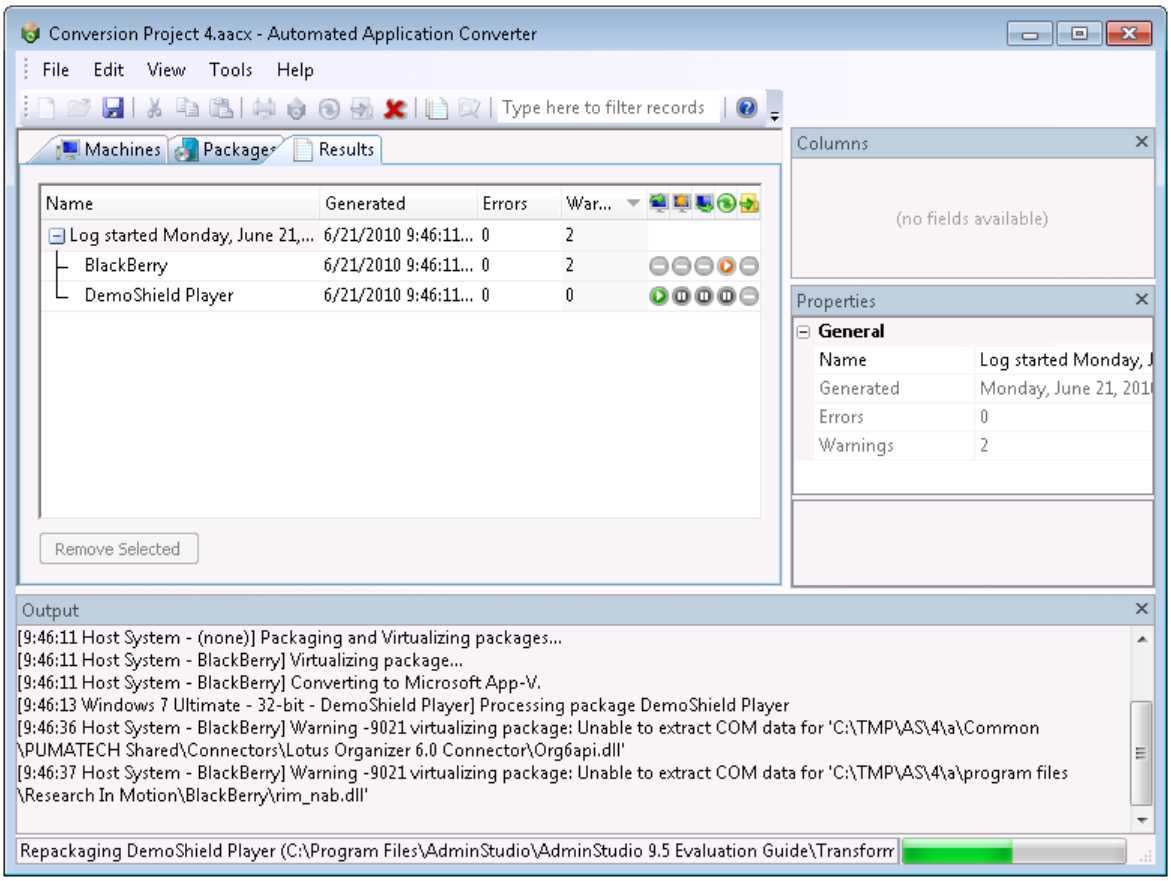
15. Under **Place packages under the following folder**, select the output location where you want to store the packages.
16. Click **Next**. The **Automated Repackaging on Virtual Machines** panel opens, prompting you to select the platform of the virtual machines that you want to use to perform automated repackaging during this conversion process.



17. From the **Virtual Machine Platform** list, select a platform, or leave **Any Platform** selected, and click **Next**. The **Application Conversion Project Wizard Complete** panel opens.



18. Click **Finish** to close the wizard and begin converting the selected packages.The conversion process begins. The **Results** tab opens and messages are displayed in the **Output** window.

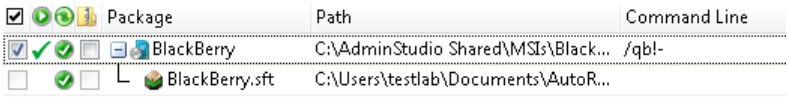

Conversion Project 4.aacx - Automated Application Converter
File Edit View Tools Help
Type here to filter records
Machines Packages Results
Name Generated Errors War...
Log started Monday, June 21,... 6/21/2010 9:46:11... 0 2
BlackBerry 6/21/2010 9:46:11... 0 2
DemoShield Player 6/21/2010 9:46:11... 0 0
Remove Selected
Columns
(no fields available)
Properties
General
Name Log started Monday, J
Generated Monday, June 21, 201
Errors 0
Warnings 2
Output
[9:46:11 Host System - (none)] Packaging and Virtualizing packages...
[9:46:11 Host System - BlackBerry] Virtualizing package...
[9:46:11 Host System - BlackBerry] Converting to Microsoft App-V.
[9:46:13 Windows 7 Ultimate - 32-bit - DemoShield Player] Processing package DemoShield Player
[9:46:36 Host System - BlackBerry] Warning -9021 virtualizing package: Unable to extract COM data for 'C:\TMP\AS\4\a\Common \PUMATECH Shared\Connectors\Lotus Organizer 6.0 Connector\Org6api.dll'
[9:46:37 Host System - BlackBerry] Warning -9021 virtualizing package: Unable to extract COM data for 'C:\TMP\AS\4\a\program files \Research In Motion\BlackBerry\rim_nab.dll'
Repackaging DemoShield Player (C:\Program Files\AdminStudio\AdminStudio 9.5 Evaluation Guide\Transform

Icons displayed on the **Results** tab indicate each package's progress:

Column	Icon	Description
Copy In **Repackage** **Copy Out**		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed, but warnings were encountered. View the results AdminStudio Automated Application Converter Log Report for detailed information on these warnings.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) failed. • **Copy In**—Error could have been caused by not being able to connect to the virtual machine. • **Repackage**—Error means that repackaging has failed. • **Copy Out**—Error could mean that you ran out of hard drive space at the package source location or that there is a permission problem preventing you from writing to the selected directory. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was skipped. Possible reasons that the operation was skipped could be: • **Repackaging not required**—Because repackaging was not required, these three operations were not required. • **Could not connect to virtual machine**—The Automated Application Converter could not successfully connect to the virtual machine, so therefore the **Repackage** and **Copy Out** operations were skipped.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is currently being performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is still being performed even though a warning was generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation was cancelled

Column	Icon	Description
Conversion Column		Package was converted to a virtual application successfully.
		Package was converted to a virtual application, but warnings were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Package was converted to a virtual application, but errors were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		The Automated Application Converter was unable to convert this package to a virtual application.
		Conversion is in progress.
		Conversion is in progress, but a warning has been generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		An error was generated when converting one of the virtual formats which caused it to fail. However, the conversion to another one of the selected virtual formats continues.
		Conversion was cancelled

19. When conversion is complete, the virtual packages will be listed in a tree structure under the original package on the **Packages** tab.



20. To view the **AdminStudio Automated Application Converter Log** report, select the top level node of the conversion run log (such as `Log started Monday, June 21, 2010...`) on the **Results** tab and do one of the following:

    - Click the **Results** button on the toolbar.
    - Select **View Report** from the shortcut menu.
    - Select **View Report** on the **Tools** menu.
    - Press Ctrl+R.

    See AdminStudio Automated Application Converter Log Report for more information.

21. Continue with the steps in Testing Packages and Importing Converted Packages into the Application Catalog.

About Automated Application Converter Project Files

All of the selections that you make on wizard panels or in the Automated Application Converter interface are saved in an XML-based project file: `ProjectName.aacx`. You can also choose to modify project settings by editing this XML file.

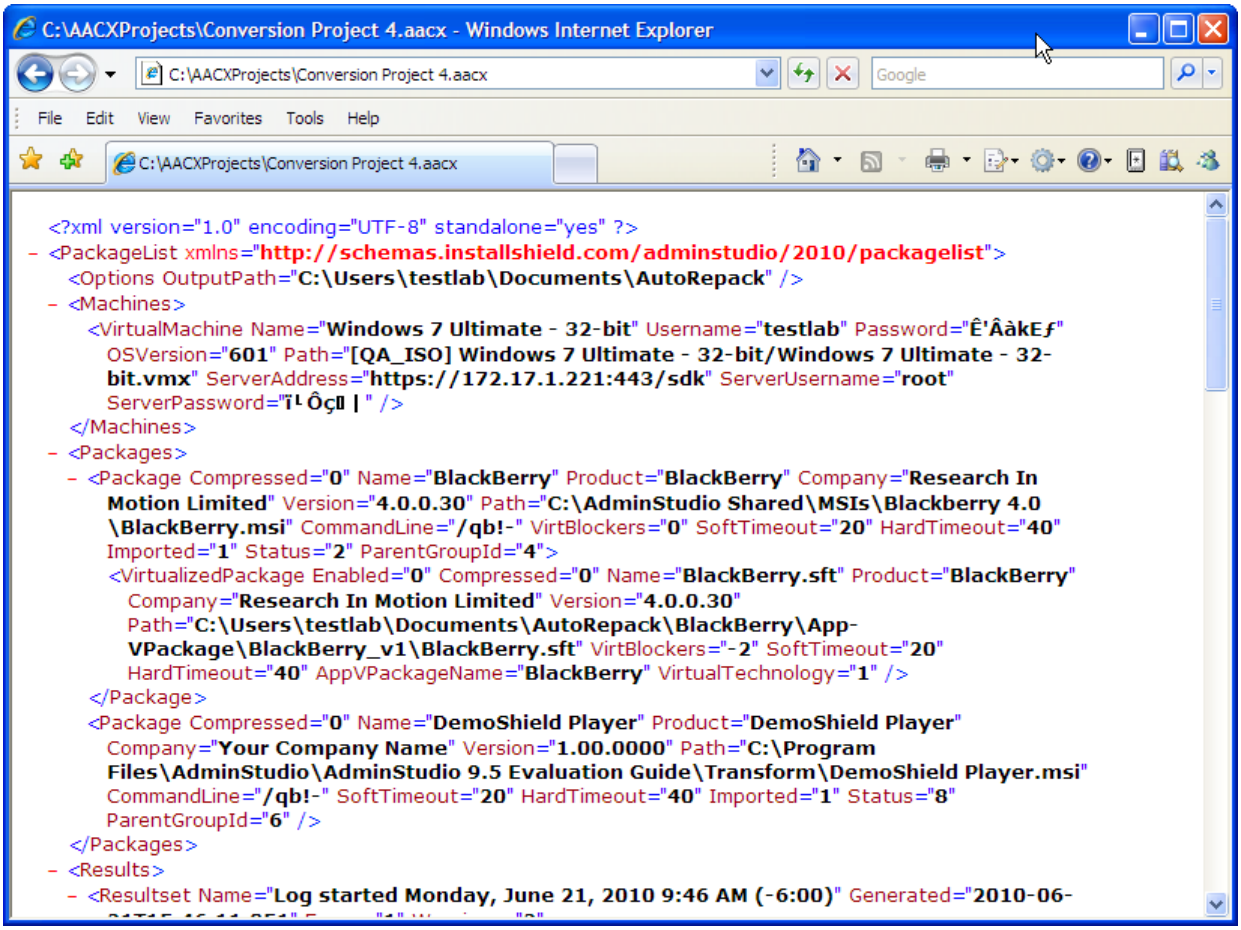


Figure 10-9: Sample Project File

Note • *To both launch the Automated Application Converter and open a project, you can double-click a project file in Windows Explorer.*

As shown in the following example, each project file is enclosed within a <PackageList> element, and the major sub-elements of a project file are <Options>, <Machines>, <Packages>, and <Results>. All of the settings specified in the Automated Application Converter interface appear in this file.

Table 10-11 • Elements Comprising an Automated Application Converter Project File

Elements

```
<PackageList>
    <Options />
    <Machines>
        <VirtualMachine />
        <VirtualMachine />
    </Machines>
    <Packages>
        <Package>
            <VirtualizedPackage />
        </Package>
        <Package>
            <RepackagedPackage />
        </Package>
    </Packages>
    <Results>
        <Resultset>
            <Machines>
                <UseMachine />
                <UseMachine />
            </Machines>
            <Packages>
                <UsePackage />
                <UsePackage />
            </Packages>
            <Messages>
                <LogItem />
                <LogItem />
            </Messages>
            <Result>
                <Messages>
                    <LogItem />
                    <LogItem />
                </Messages>
            </Result>
        </Resultset>
    </Results>
</PackageList>
```

The following table describes the major elements of a project file:

Table 10-12 • Major Elements of an Automated Application Converter Project File

Element	Description
PackageList	The root element of an Automated Application Converter project file is the <PackageList> element: `<PackageList xmlns="http://schemas.installshield.com/adminstudio/2010/packagelist">` The PackageList element identifies the location of the XML namespace used by the application. XML namespaces provide a method to avoid element name conflicts.
Options	The <Options> element of a project file identifies the output location of the converted packages and identifies the currently selected output formats: `<Options OutputPath="C:\Users\testlab\Documents\AutoRepack" Msi="1" Citrix="1" />`

Machines	The <Machines> element contains multiple <VirtualMachine> elements, which identify the virtual machines that you have added to the project:
	
	`<Machines>`
	`  <VirtualMachine Name="Windows 7 Enterprise - 32-bit" Username="testlab" Password="Êé`
	`  <VirtualMachine Name="Windows XP" Username="testlab" Password="Ê'ÂàkEƒ" Path="[`
	`</Machines>`

Packages	The <Packages> element contains multiple <Package> elements, which identify the packages that you have added to the project:
	
	`<Packages>`
	`  <Package Compressed="0" Name="LicModInj" Product="LicModInj" Company="Your`
	`  <VirtualizedPackage Enabled="0" Compressed="0" Name="LicModInj.sft" Pro`
	`  <ErrorVirtualizedPackage Enabled="0" Compressed="0" Name="LicModInj.pro`
	`  </Package>`
	`  <Package Compressed="0" Name="MathPlot" Product="MathPlot" Company="Install`
	`  <Package Compressed="0" Name="BlackBerry" Product="BlackBerry" Company="Res`
	`  <VirtualizedPackage Enabled="0" Compressed="0" Name="BlackBerry.sft" Pr`
	`  <VirtualizedPackage Enabled="0" Compressed="0" Name="BlackBerry.profile`
	`  </Package>`
	`</Packages>`
	
	Each <Package> element can have multiple <RepackagedPackage>, <VirtualizedPackage>, and <ErrorVirtualizedPackage> elements.

Table 10-12 • Major Elements of an Automated Application Converter Project File

Element	Description

Results	The <Results> element contains multiple <Resultset> elements, each of which contains information on a conversion run:
	
	`<Results>`
	`  <Resultset Name="Log started Friday, June 11, 2010 2:56 PM (-6:00)" Generated="`
	`  <Machines>`
	`  <UseMachine Name="Windows 7 Ultimate - 32-bit" Path="[QA_ISO] Windows 7`
	`  <UseMachine Name="Windows 7 Ultimate - 64-bit" Path="[QA_ISO] Windows 7`
	`  </Machines>`
	`  <Packages>`
	`  <UsePackage Name="InstallShield MSIDiff" Path="C:\Program Files\AdminSt`
	`  <UsePackage Name="LicModInj" Path="C:\Program Files\AdminStudio\9.5\Qui`
	`  <UsePackage Name="MathPlot" Path="C:\Program Files\AdminStudio\9.5\Quic`
	`  </Packages>`
	`  <Messages>`
	`  <LogItem Id="4300" Flags="4" Message="Processing packages..." Time="201`
	`  <LogItem Id="4301" Message="Using virtual machine Windows 7 Ultimate -`
	`  <LogItem Id="4301" Message="Using virtual machine Windows 7 Ultimate -`
	`  </Messages>`
	`  <Result Name="InstallShield MSIDiff" Generated="2010-06-11T20:56:02.284" Er`
	`  <Messages>`
	`  <LogItem Id="4336" Flags="2" Message="Package InstallShield MSIDift`
	`  <LogItem Id="4345" Message="Virtualizing package..." Time="2010-06-`
	`  <LogItem Id="4347" Message="Converting to Microsoft App-V." Time="2`
	`  </Messages>`
	`  </Result>`
	`  <Result Name="LicModInj" Generated="2010-06-11T20:56:02.285" Warnings="7" C`
	`  <Messages>`
	`  <LogItem Id="4336" Flags="2" Message="Package LicModInj will be dir`
	`  <LogItem Id="4345" Message="Virtualizing package..." Time="2010-06-`
	`  <LogItem Id="4347" Message="Converting to Microsoft App-V." Time="2`
	`  </Messages>`
	`  </Result>`
	`  </Resultset>`
	`</Results>`
	
	• Each <Resultset> element of a <Results> element includes information on <Machines>, <Packages>, <Messages>, and multiple <Result> elements for that run.
	• Each <Result> element of a <Resultset> element contains information on the conversion run for an individual package.

Using Automated Application Converter in Evaluation Mode

When using the Automated Application Converter in Evaluation mode, you can only use one virtual machine and convert up to three packages during one repackaging/virtualization conversion run. Even if more than one virtual machine is selected, only one will be used, and even if more than three packages are selected, only three will be processed.

Managing Virtual Machines

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- Microsoft Hyper-V Server
- VMware ESX or ESXi Server
- VMware Workstation 6.5 or later

You can use the Virtual Machine Import Wizard to add “clean” virtual machines to the **Machines** tab of the Automated Application Converter, making them available for use during automated repackaging, conversion to App-V 5.0 format using the App-V Sequencer, and testing of App-V 5.0 packages.

Information about managing virtual machines is presented in the following sections:

- Virtual Machine System Requirements
- Preparing Your Virtual Machines for Use With the Automated Application Converter
- VMware vSphere API Requirement on the AdminStudio Machine
- Adding Virtual Machines Using the Virtual Machine Import Wizard
- Editing Virtual Machine Properties on the Machines Tab

Virtual Machine System Requirements

Automated Application Converter performs automated repackaging on virtual machines. This section lists the virtual machine platform and virtual machine image system requirements.

- Supported Virtual Machine Platforms
- VMware Requirements
- Microsoft Hyper-V Server Requirements
- Virtual Machine Image Requirements

Supported Virtual Machine Platforms

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- VMware ESX/ESXi Server, Version 5.5 or later
- VMware Workstation 6.5 or later
- Microsoft Hyper-V Server 2008 R2 or later

VMware Requirements

As described above, Automated Application Converter supports automated repackaging on VMware ESX/ESXi Server and VMware Workstation.

- VMware vSphere API Requirement
- VMware ESX/ESXi Server Permission Requirements

VMware vSphere API Requirement

In order for Automated Application Converter to perform automated repackaging, it needs to communicate with the virtualization technology that you are using. If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation), the VMware vSphere API needs to be installed on the same machine as the Automated Application Converter. You can do this by either installing VMware Workstation on that machine or by downloading and installing the VMware vSphere API from the following location:

http://www.vmware.com/support/developer/vSphere-api

***Note** • When using VMware Workstation, it is recommended that you install VMware Workstation on the same machine as Automated Application Converter so that Automated Application Converter will use the version of the vSphere API that was designed for that specific version of VMware Workstation. Although it is likely that newer versions of the vSphere API will also work, it seems that the best approach is for Automated Application Converter to use the version of the vSphere API that was bundled with your version of VMware Workstation.*

VMware ESX/ESXi Server Permission Requirements

If you plan to use a VMware ESX/ESXi Server in conjunction with Automated Application Converter, make sure that the account that you use to log in to this server has the permissions/roles needed to automatically open a VM using VMware vSphere API. The account needs to either have an administrator role assigned or, at least, have the following three roles assigned:

- All Privileges/Virtual Machine/State/Create Snapshot
- All Privileges/Virtual Machine/State/Delete Snapshot
- All Privileges/Virtual Machine/Interaction/Console Interaction

If the login account does not have these permissions/roles, Automated Application Converter will be unable to automatically boot up a virtual machine on that server.

Microsoft Hyper-V Server Requirements

As described above, Automated Application Converter supports automated repackaging on Microsoft Hyper-V Server. When preparing a Hyper-V Server for use with Automated Application Converter, make sure that the following conditions are met:

- **Configuration tools**—Verify that the Hyper-V configuration tools are installed on the Hyper-V server machine. These tools can be installed using the Microsoft Hyper-V Management Console.
- **Connection**—Verify that you can successfully connect to the Hyper-V Server from the machine where AdminStudio Automated Application Converter is installed.
- **Permissions**—Make sure that the Hyper-V Server user has the permissions required to perform operations on the Hyper-V machines.
- **Configuration settings**—Connecting to a WMI namespace on a remote computer running Windows Vista or Windows Server 2008 may require changes to configuration settings. Check the following configuration settings on the AdminStudio machine as well as on the Hyper-V Server machine:
  - Windows Firewall Settings
  - User Account Control (UAC) Settings
  - DCOM Settings
  - Common Information Model Object Manager (CIMOM) Settings

  ***Note** • For detailed information, see Connecting to WMI Remotely at:*

  http://msdn.microsoft.com/en-us/library/aa822854(VS.85).aspx

Virtual Machine Image Requirements

Automated Application Converter uses virtual machines to perform automated repackaging. These virtual machines have the following requirements:

Virtual Machine System Requirements

When creating a virtual machine image that will be hosted on one of the virtual machine platforms listed above, the recommended minimum requirements should meet those required by the applications you are trying to repackage. Since you repackage on the target deployment platform, the virtual machine image should closely resemble the target deployment environment.

Preparing Your Virtual Machines for Use With the Automated Application Converter

You need to prepare each virtual machine that you are going to use with the Automated Application Converter to perform automated repackaging by running the Virtual Machine Preparation setup and by creating a snapshot. For instructions, see Preparing Your Virtual Machines for Use With the Automated Application Converter.

Preparing Your Virtual Machines for Use With the Automated Application Converter

Information about preparing virtual machines for use with Automated Application Converter are presented in the following topics:

- Preparing Virtual Machines
- Running the Virtual Machine Preparation Setup
- Taking a Snapshot
- VMware-Specific Snapshot Configuration Option

Preparing Virtual Machines

Automated Application Converter uses virtual machine snapshots to perform repackaging, to perform conversion to App-V 5.0 format using the App-V 5.0 Sequencer, and to test App-V 5.0 virtual packages.

- Preparing a Snapshot for Repackaging
- Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer
- Preparing a Snapshot for App-V 5.0 Testing Using the App-V 5.0 Client

You can add multiple virtual machines to Automated Application Converter and each of those virtual machines can have multiple snapshots.

Preparing a Snapshot for Repackaging

To prepare a virtual machine snapshot for use by Automated Application Converter to perform automated repackaging, perform the following steps:

Task ***To prepare a snapshot for repackaging:***

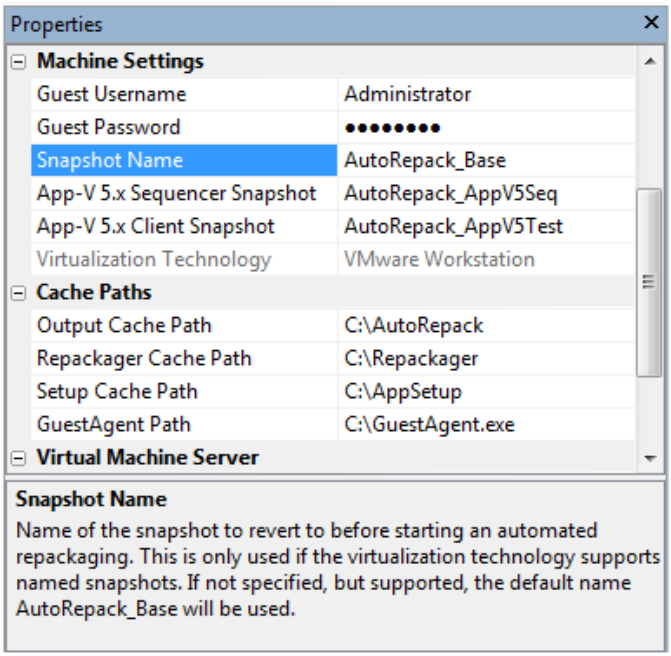
1. Launch the virtual machine and run the Virtual Machine Preparation setup, as described in Running the Virtual Machine Preparation Setup.
2. At the end of the Virtual Machine Preparation setup, you will be prompted to restart the virtual machine. Restart the virtual machine and verify that you are automatically logged in and that `GuestAgent.exe` is launched:

Note • *The Guest Agent (`GuestAgent.exe`) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the AdminStudio Repackager in an automated fashion.*

3. Shut down the virtual machine.
4. Take a snapshot, as described in Taking a Snapshot. If your virtualization technology supports named snapshots, name the snapshot `AutoRepack_Base`, which is the default name that the Automated Application Converter will be looking for when performing repackaging.

Note • *If you assign a snapshot name other than `AutoRepack_Base`, after you add the virtual machine to the Automated Application Converter, you need to enter that snapshot name in the **Snapshot Name** field in the **Properties** window of the **Machines** tab for that virtual machine.*

5. If the virtual machine containing this snapshot is not already added to Automated Application Converter, proceed with the steps in Adding Virtual Machines Using the Virtual Machine Import Wizard.
6. Open the **Machines** tab of Automated Application Converter and select the virtual machine that contains this snapshot.
7. In the **Properties** window under **Machine Settings**, enter the name of this snapshot in the **Snapshot Name** field. If no name is entered, the default value of `AutoRepack_Base` will be used.



Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer

To prepare a virtual machine snapshot for use by Automated Application Converter to perform automated conversion to App-V 5.0 format using the App-V 5.0 Sequencer, perform the following steps:

Task ***To prepare a snapshot for conversion to App-V 5.0 format using the App-V 5.0 Sequencer:***

1. Launch the virtual machine and run the Virtual Machine Preparation setup, as described in Running the Virtual Machine Preparation Setup.
2. At the end of the Virtual Machine Preparation setup, you will be prompted to restart the virtual machine. Restart the virtual machine and verify that you are automatically logged in and that `GuestAgent.exe` is launched:

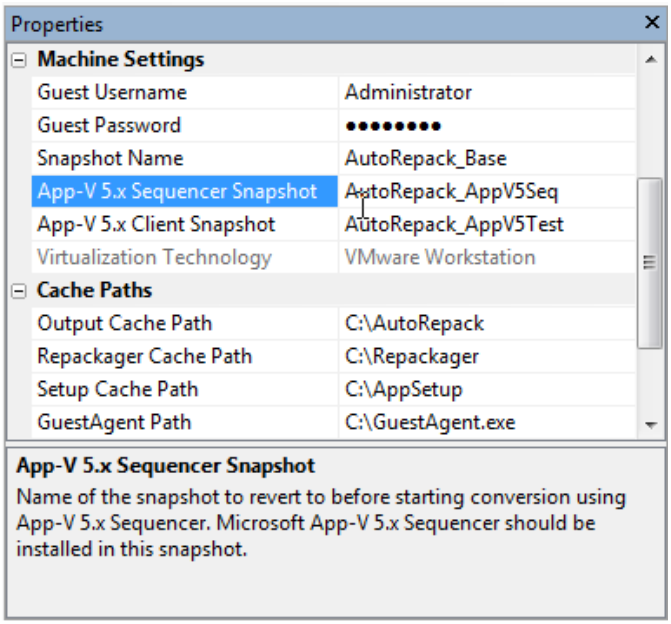
Note • *The Guest Agent (`GuestAgent.exe`) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the Microsoft App-V 5.0 Sequencer in an automated fashion.*

3. Make sure that this snapshot meets the following requirements:

Requirement	Description
Operating System	Windows 7 SP1 or later The following updates are also required for computers running the following operating systems: ● **Windows 7 or Windows 7 SP1 (32-bit or 64-bit)** —Download and install KB2533623: http://go.microsoft.com/fwlink/?LinkId=286100 ● **Windows Server 2008 R2 SP1**—Download and install KB2533623: http://go.microsoft.com/fwlink/?LinkId=286102
PowerShell	Windows PowerShell 3.0: http://www.microsoft.com/en-us/download/details.aspx?id=34595
.NET	Microsoft .NET Framework 4 (Full Package): http://www.microsoft.com/en-us/download/details.aspx?id=17718

Note • *These prerequisites are already installed for computers that run Windows 8 and Windows Server 2012.*

4. Run the Microsoft App-V 5.0 Sequencer installer (which you received when you purchased Microsoft App-V 5.0).
5. Shut down the virtual machine.
6. Take a snapshot, as described in Taking a Snapshot. If your virtualization technology supports named snapshots, give the snapshot a name that indicates its purpose, such as `AutoRepack_AppV5Seq`.
7. If the virtual machine containing this snapshot is not already added to Automated Application Converter, proceed with the steps in Adding Virtual Machines Using the Virtual Machine Import Wizard.
8. Open the **Machines** tab of Automated Application Converter and select the virtual machine that contains this snapshot.
9. In the **Properties** window under **Machine Settings**, enter the name of this snapshot in the **App-V 5.x Sequencer Snapshot** field.



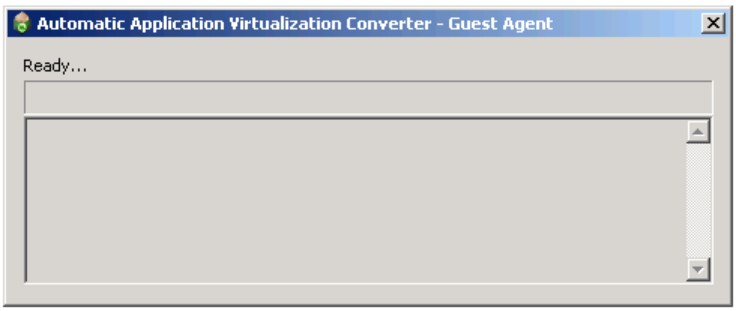
Preparing a Snapshot for App-V 5.0 Testing Using the App-V 5.0 Client

To prepare a virtual machine snapshot for use by Automated Application Converter to perform testing of App-V 5.0 packages, perform the following steps:

Note • *It is not necessary to create a snapshot for App-V 4.x package testing; you will have the option of choosing to install the App-V 4.x client at the time of testing.*

Task ***To prepare a snapshot for testing of App-V 5.0 packages:***

1. Launch the virtual machine and run the Virtual Machine Preparation setup, as described in Running the Virtual Machine Preparation Setup.
2. At the end of the Virtual Machine Preparation setup, you will be prompted to restart the virtual machine. Restart the virtual machine and verify that you are automatically logged in and that `GuestAgent.exe` is launched:



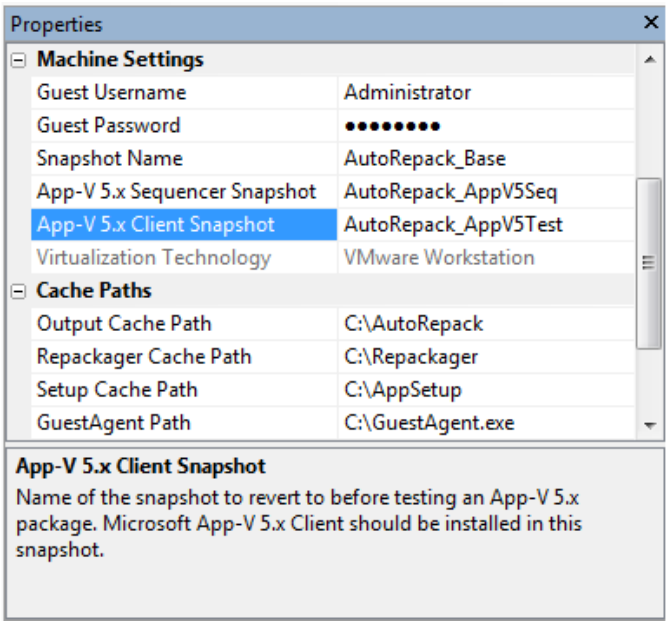
***Note** • The Guest Agent (`GuestAgent.exe`) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the Microsoft App-V 5.0 Client in an automated fashion.*

3. Make sure that this snapshot meets the following requirements:

Requirement	Description
Operating System	Windows 7 SP1 or later If you are using Windows 7 or Windows 7 SP1 (32-bit or 64-bit), download and install KB2533623: http://go.microsoft.com/fwlink/?LinkId=286100
PowerShell	Windows PowerShell 3.0: http://www.microsoft.com/en-us/download/details.aspx?id=34595
.NET	Microsoft .NET Framework 4 (Full Package): http://www.microsoft.com/en-us/download/details.aspx?id=17718

***Note** • These prerequisites are already installed for computers that run Windows 8 and Windows Server 2012.*

4. Run the Microsoft App-V 5.0 Client installer (which you received when you purchased Microsoft App-V 5.0).
5. Shut down the virtual machine.
6. Take a snapshot, as described in Taking a Snapshot. If your virtualization technology supports named snapshots, give the snapshot a name that indicates its purpose, such as `AutoRepack_AppV5Test`.
7. If the virtual machine containing this snapshot is not already added to Automated Application Converter, proceed with the steps in Adding Virtual Machines Using the Virtual Machine Import Wizard. When you add this machine, make sure that you set its **Purpose** property to either **Any** or **Testing**. If this machine's **Purpose** property is set to **Repackaging**, this machine will not be available when performing testing.
8. Open the **Machines** tab of Automated Application Converter and select the virtual machine that contains this snapshot.
9. In the **Properties** window under **Machine Settings**, enter the name of this snapshot in the **App-V 5.x Client Snapshot** field.



Running the Virtual Machine Preparation Setup

On each virtual machine that you are going to use to perform automated repackaging, you need to run the Virtual Machine Preparation setup, an application that will enable automatic login. When you install AdminStudio, you will find the Virtual Machine Preparation setup in the following location:

```
C:\Program Files (x86)\AdminStudio\2022 R2\Repackager\VirtualMachinePrep\VMCfg.exe
```

You need to run this application one time on all of the virtual machines that you are going to use with the Automated Application Converter.

Note • *If you do not run the Virtual Machine Preparation setup on the virtual machines you want to use, the Automated Application Converter will be unable to connect to them.*

Taking a Snapshot

After you have run the Virtual Machine Preparation setup on a virtual machine, you need to shut it down and create a snapshot. This enables the Automated Application Converter to revert the virtual image to a clean state after each repackaging run.

Note • *If you do not take a snapshot of the virtual image, the Automated Application Converter will be unable to revert the image to a clean state after completing a repackaging run. Therefore, while the first repackage on the virtual machine would be on a clean image, all subsequent repackaging runs would be run on a "dirty" virtual image.*

Links to instructions on how to create a snapshot of a virtual machine are presented in the following table.

Note • *Make sure* that the following steps are executed before starting the conversion process:

a. **Windows updates** are up to date on the virtual machine before taking snapshot.

b. ***Firewall*** *needs to be* ***disabled*** *on guest agent machine.*

:

Table 10-13 • Instructions for Taking a Snapshot of a Virtual Machine

Platform	Link
VMware Workstation	 ***To take a snapshot:*** **1.** Open VMware Workstation. **2.** On the **VM** menu, point to **Snapshot** and click **Take Snapshot**. **3.** Name the snapshot `AutoRepack_Base`. **4.** You can optionally add a description to record notes about the virtual machine state captured in the snapshot. **5.** Click **OK**.
VMware ESX or ESXi Server	To take a snapshot on the VMware ESX or ESXi server, perform the following steps: ***To take a snapshot:*** **1.** Open the VMware infrastructure client. **2.** On the **Inventory** menu, point to **Virtual Machine** and **Snapshot**, and then click **Take Snapshot**. **3.** Name the snapshot `AutoRepack_Base`. **4.** You can optionally add a description to record notes about the virtual machine state captured in the snapshot. **5.** Click **OK**. ***Note*** • *For more information, see the VMware Knowledge Base article entitled Understanding virtual machine snapshots in VMware ESX:* *http://kb.vmware.com/kb/1015180*

VMware-Specific Snapshot Configuration Option

When configuring a VMware Workstation or VMware ESX/ESXi Server image, there is a setting that controls what VMware does with a virtual machine when it is powered off. In VMware Workstation, the option is set on the **Options** tab of the **Virtual Machine Settings** dialog box:

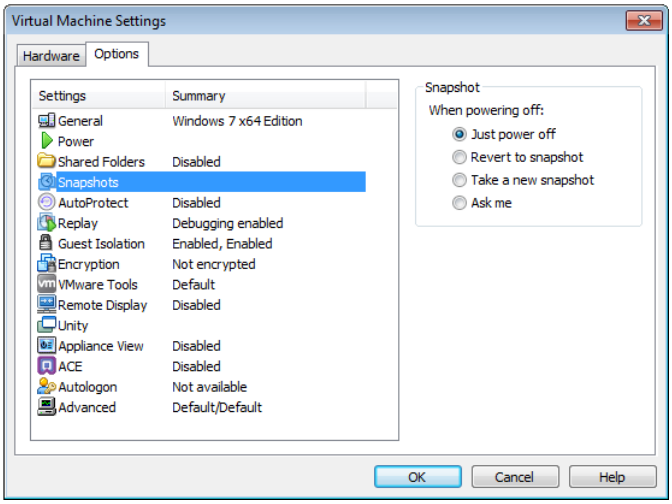


Figure 10-10: Snapshots Options on VMware Workstation Virtual Machine Settings Dialog Box

It is recommended that you set the **Snapshot** / **When powering off** option to **Just power off**. Do not select the **Ask me** option; selecting this option would block the use of this virtual image by Automated Application Converter until the prompt is dismissed by the user.

VMware vSphere API Requirement on the AdminStudio Machine

In order for Automated Application Converter to perform automated repackaging, it needs to communicate with the virtualization technology that you are using. If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation), the VMware vSphere API needs to be installed on the same machine as the Automated Application Converter. You can do this by either installing VMware Workstation on that machine or by downloading and installing the VMware vSphere API from the following location:

http://www.vmware.com/support/developer/vSphere-api

Note • *When using VMware Workstation, it is recommended that you install VMware Workstation on the same machine as Automated Application Converter so that Automated Application Converter will use the version of the vSphere API that was designed for that specific version of VMware Workstation. Although it is likely that newer versions of the vSphere API will also work, it seems that the best approach is for Automated Application Converter to use the version of the vSphere API that was bundled with your version of VMware Workstation.*

Adding Virtual Machines Using the Virtual Machine Import Wizard

You can add one or multiple virtual machines to the Automated Application Converter to use to perform automated repackaging during conversion to virtual packages.

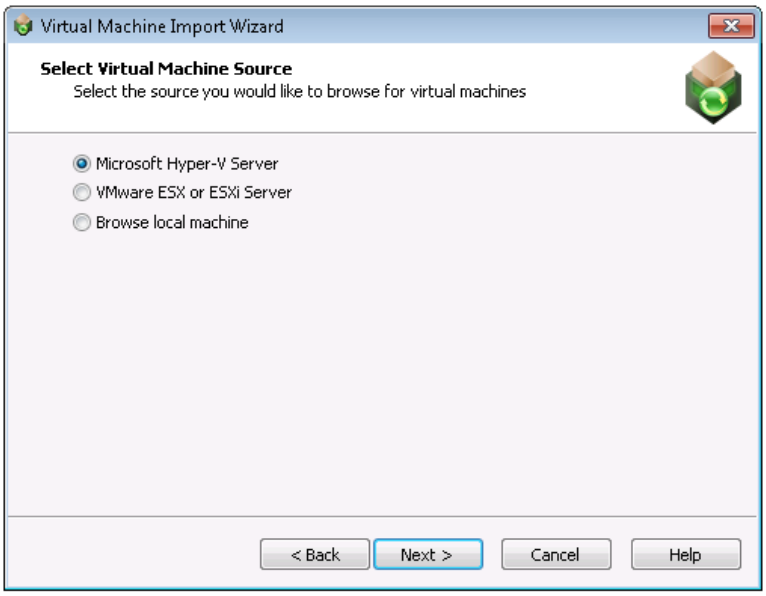
You have the option of selecting just one virtual machine to use for all repackaging, or selecting an operating system group of multiple virtual machines that can be used simultaneously to speed up the repackaging of multiple setups.

If you have specified a group of multiple virtual machines, a package in the conversion list is assigned to each virtual machine. Then, when a virtual machine finishes repackaging a package, it is reverted to its clean snapshot image, and then starts repackaging the next package in the list.

To add virtual machines to the **Machines** tab using the Virtual Machine Import Wizard, perform the following steps.

Task ***To add virtual machines:***

1. Open the **Machines** tab of the Automated Application Converter.
2. Click **Add Machine**. The **Welcome** panel of the **Virtual Machine Import Wizard** opens.
3. Click **Next**. The **Select Virtual Machine Source** panel opens, prompting you to select the type of virtual machine that you are adding.

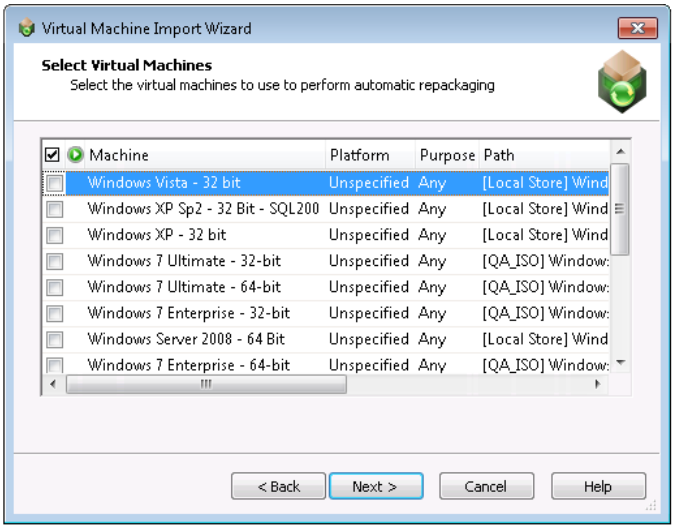


4. Select one of the following options and click **Next**.
   - **Microsoft Hyper-V Server**—Select this option to add a virtual image from a Microsoft Hyper-V Server.
   - **VMware ESX or ESXi Server**—Select this option to add a virtual image from a VMware ESX or ESXi Server.
   - **Browse local machine**—Select this option to add a virtual image from a local installation of VMware Workstation

5. Based upon your selection on the **Select Virtual Machine Source** panel, enter the following information:

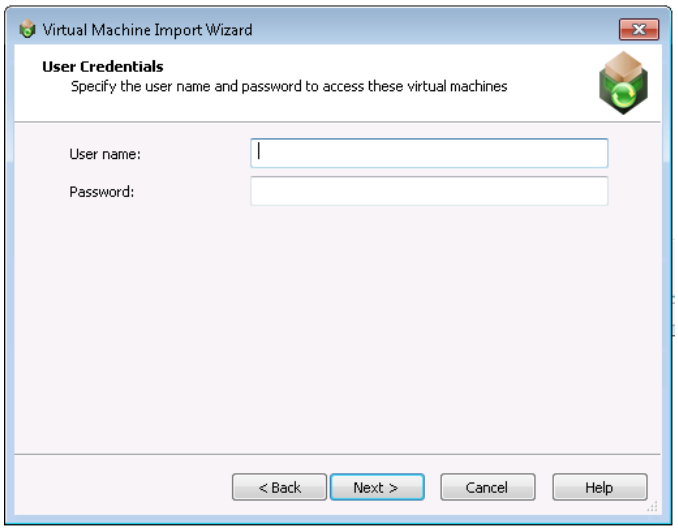
Virtual Machine Source	Steps to Take
Microsoft Hyper-V Server	On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, enter the following information: • **Server Name**—Enter the server name of the Microsoft Hyper-V Server that you want to connect to. • **Authentication**—Select **Windows Authentication** if you want to use the credentials of the logged in user to login to the Hyper-V Server. Select **Server Authentication** if you want to connect to the Hyper-V Server using the specified **User name** and **Password**.
VMware ESX or ESXi Server	On the **Select Virtual Machines from VMware ESX or ESXi Server** panel, enter the following information: • **Server Name**—Enter the name of the VMware ESX or ESXi server. • **User name**—Enter the login ID for the VMware ESX or ESXi server. • **Password**—Enter the password for the VMware ESX or ESXi server.
Browse local machine	On the **Select Virtual Machines** panel, do one of the following: ***To add an individual virtual machine:*** 1. Click **Browse Files**. The **Select Virtual Machine Image File** dialog box opens. 2. Select the virtual machine image you want to add to the project and click **Open**. ***To add all of the virtual machines in a specific directory:*** 1. Click **Browse Folders**. The **Browse for Folder** dialog box opens. 2. Select a directory that contains the virtual machine images that you want to add to your project and click **OK**.

When you have finished this step, the virtual machines will be listed (but not selected) on the **Select Virtual Machines** panel.



6. On the **Select Virtual Machines** panel, select the virtual machine images that you want to use to perform automated repackaging.
7. For each selected image, click in the **Platform** column and identify its platform.
8. By default, virtual machines that you add to the **Machines** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options:
   - **Repackaging**—Virtual machine will only be used to perform automated repackaging.
   - **Testing**—Virtual machine will only be used to test packages. You test a package by right-clicking on it on the **Packages** tab and selecting **Launch Package for Testing** from the shortcut menu. You will then be prompted to install and run that package on a virtual machine.
   - **Any**—Make this virtual machine available for use during both automated repackaging and package testing.

   ***Note** • The **Launch Package for Testing** functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*
9. Click **Next**. The **User Credentials** panel opens, prompting you to specify the login credentials to use to access the selected virtual machines.



10. Enter the user credentials and click **Next**. The Virtual Machine Import Wizard Complete panel opens.
11. Click **Finish** to close the wizard and add the selected virtual machines to your project.

Editing Virtual Machine Properties on the Machines Tab

By default, the list of machines on the **Machines** tab lists the **Machine**, **Platform**, **Purpose**, and **Path** columns. Additional properties can be viewed and edited in the **Properties** window.

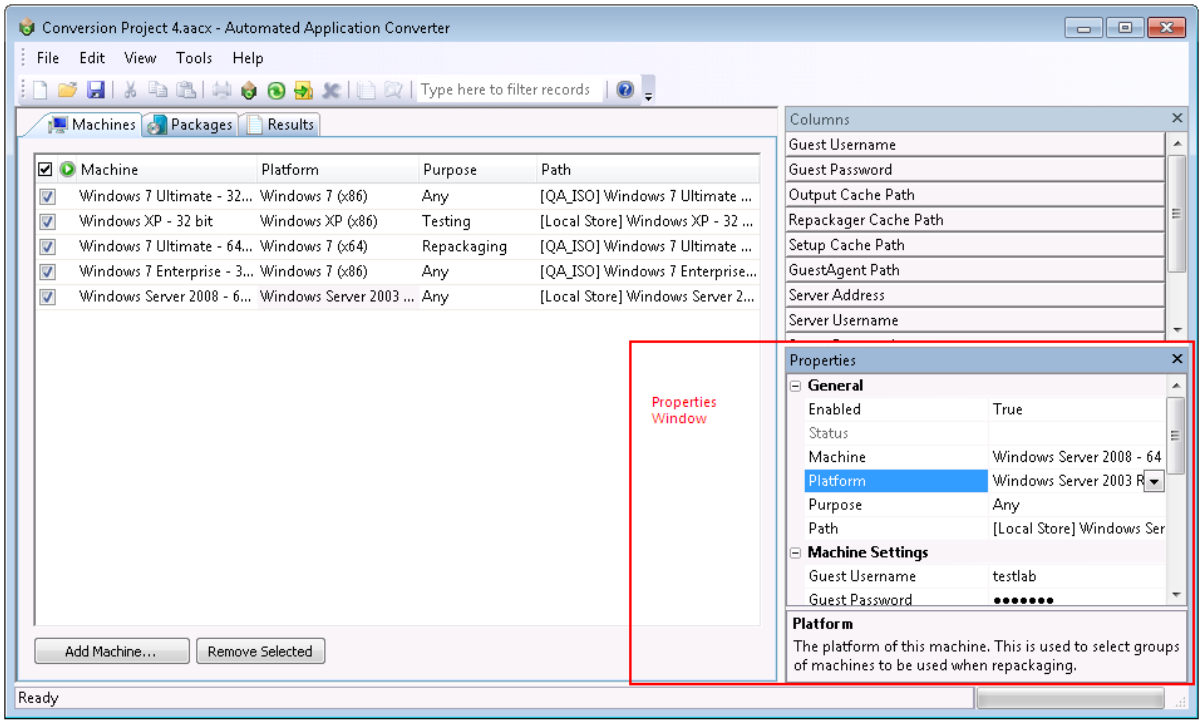


Figure 10-11: Properties Window on the Machines Tab

To edit the properties of a virtual machine, perform the following steps:

Task ***To edit a virtual machine's properties:***

1. Open the **Machines** tab.
2. Select a virtual machine in the list.
3. Click in the list or in the **Properties** window to edit the following properties:

Property	Description
Enabled	Set to **True** to enable the machine or **False** to disable it. When enabled, it is available for use by Automated Application Converter.
Status	Lists a status to indicate whether the machine is idle, is in use, or has suffered an unrecoverable error in the last attempt to use it.
Machine	Name of the virtual machine.
Platform	Field that identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.

Property	Description
Purpose	By default, virtual machines that you add to the **Packages** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options: • **Repackaging**—Virtual machine will only be used to perform automated repackaging. • **Testing**—Virtual machine will only be used to test packages. You test a package by right-clicking on it on the **Packages** tab and selecting **Launch Package for Testing** from the shortcut menu. You will then be prompted to install and run that package on a virtual machine. • **Any**—Make this virtual machine available for use during both automated repackaging and package testing. This is the default value. ***Important*** *• If the **Purpose** column is not listed in the **Machines** list, you can edit the **Purpose** value in the Properties window.* ***Note*** *• The **Launch Package for Testing** functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*
Path	Path on the server or file system to this virtual machine image file is located.
Guest Username	The user name to use to login to this virtual machine.
Guest Password	The password to use to login to this virtual machine.
Snapshot Name	Name of the snapshot to revert to before starting an automated repackaging session. This is only used if the virtualization technology supports named snapshots. If this value is not specified, but named snapshots are supported on the virtualization technology, the default name of `AutoRepack_Base` will be used.

Property	Description
App-V 5.x Sequencer Snapshot	Enter the name of the snapshot to revert to before starting conversion using the App-V 5.x Sequencer. ***Important*** *• If you do not specify a snapshot name and then attempt to perform conversion using the* ***App-V 5.x with Sequencer*** *package creation method, you will receive the following error message and the conversion will fail.* ***`Error: No snapshot name was specified for App-V 5.x Sequencer conversion. Please specify this for at least one machine in the Machines tab and retry.`*** ***Important*** *• Both the Microsoft App-V 5.x Sequencer and the Virtual Machine Preparation client must be installed on this snapshot. For more information, see Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer.*
App-V 5.x Client Snapshot	Enter the name of the snapshot to revert to before testing an App-V 5.x package. This snapshot will be used if the user right-clicks on this App-V 5.0 virtual package on the **Packages** tab of Automated Application Converter and selects **Launch Package for Testing** from the shortcut menu. ***Note*** *• If you do not specify a snapshot name and then attempt to test an App-V 5.x package by selecting* ***Launch Package for Testing*** *from the shortcut menu, the following error will appear in the output window:* *`Error: No snapshot name was specified for testing with App-V 5.x Client. Please specify a snapshot name in the Machines tab and retry.`* ***Important*** *• Both the Microsoft App-V 5.x client and the Virtual Machine Preparation client must be installed on this snapshot. For more information, see Preparing a Snapshot for App-V 5.0 Testing Using the App-V 5.0 Client.*
Virtualization Technology	The virtualization technology powering this virtual machine.
Output Cache Path	Specify the location for the repackaged output on the virtual machine. By default, this value is `C:\AutoRepack`.
Repackager Cache Path	Specify the location where Repackager will be installed on the virtual machine. By default, this value is `C:\Repackager`.
Setup Cache Path	Specify the location where the package will be copied to on the virtual machine. By default, this value is `C:\AppSetup`.
GuestAgent Path	Specify the location where the `GuestAgent.exe` file will be installed on the virtual machine. By default, this value is `C:\GuestAgent.exe`.

Property	Description
Server Address	The address of the virtual machine server on which this virtual machine is found. This may be a host name or a URL.
Server Username	The user name of the account used to access the virtual machine server.
Server Password	The password of the account used to access the virtual machine server.
Add Machine	Click to launch the Virtual Machine Import Wizard, which you can use to add virtual machines to the **Machines** tab.
Remove Selected	Click to remove the selected virtual machine from this list. ***Note*** *• A virtual machine is selected for removal when you click on it and it becomes highlighted, not by selecting the virtual machine's check box. Use the* ***Ctrl*** *key to select multiple machines.*

Connecting to Active Virtual Machines

When the Automated Application Converter is connected to a virtual machine and it is performing repackaging, you can use Remote Desktop to open that virtual machine directly from the Automated Application Converter interface to check on the progress of the repackaging run.

Task ***To open a virtual machine from the Automated Application Converter interface:***

1. Add virtual machines to your project file, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
2. Begin a conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard.
3. Open the **Machines** tab or the **Packages** tab.
   - On the **Machines** tab, right-click on a machine that is currently performing repackaging. The machine will have a status of Running ( ), which means that the Automated Application Converter has connected to the virtual machine and the `GuestAgent.exe` is running.
   - On the **Packages** tab, right-click on a package that is currently being repackaged on a virtual machine. The package will have a status of Running ( ).
4. On the shortcut menu, select **Connect to Machine**. The virtual machine opens in a Remote Desktop window.

Tip *• If you have selected a package that has a status of Running but which does not require repackaging, the* ***Connect to Machine*** *selection will be disabled.*

Managing Packages to Convert

You can use the **Package Import Wizard** to add packages to your Automated Application Converter project for conversion.

You can also modify package properties on the **Packages** tab.

- Adding Packages from an AdminStudio Application Catalog
- Adding Packages from a Local Machine or Network
- Editing Package Properties on the Packages Tab

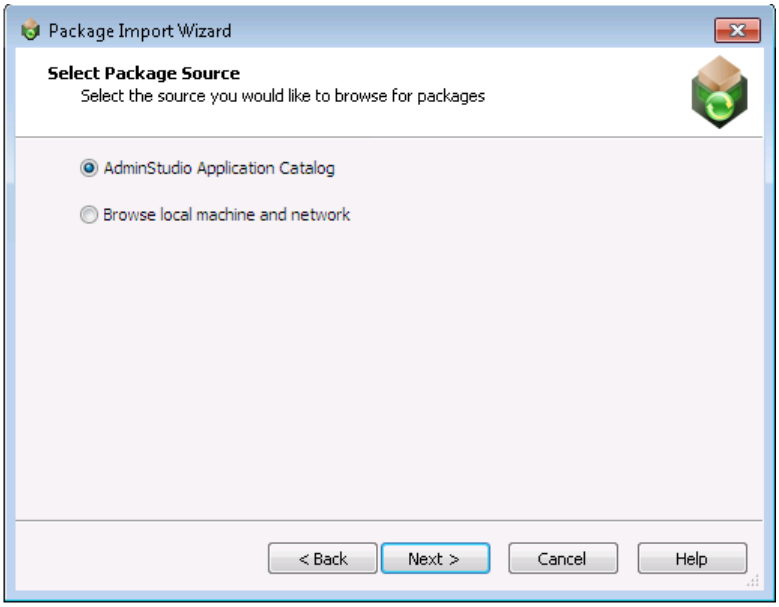
Adding Packages from an AdminStudio Application Catalog

To add packages from an AdminStudio Application Catalog to Automated Application Converter so that you can convert them to virtual applications, perform the following steps:

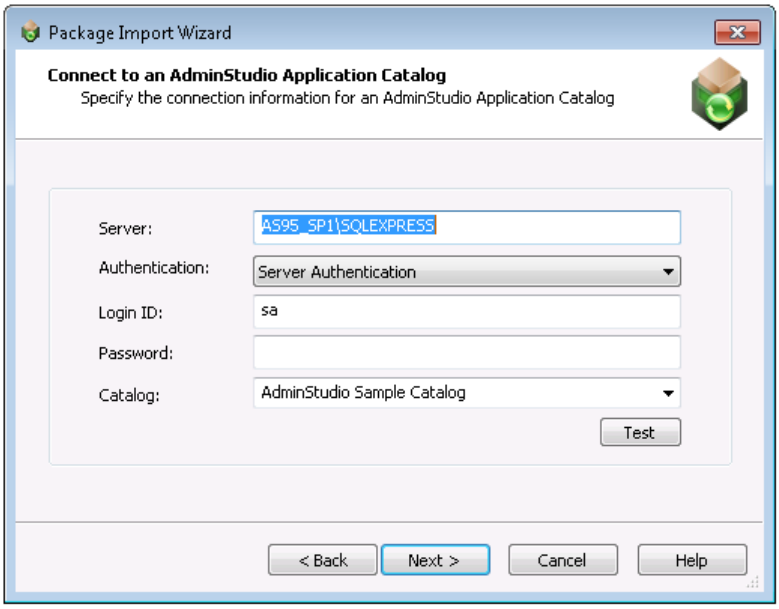
Tip • *To select packages from Microsoft Configuration Manager to convert to virtual applications, first import those packages into the Application Catalog, as described in Importing From ConfigMgr (Formerly called as System Center Configuration Manager).*

Task ***To add packages from an AdminStudio Application Catalog:***

1. Launch the Automated Application Converter.
2. Add one or more virtual machines, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
3. Open the **Packages** tab.
4. Click **Add Packages**. The **Package Import Wizard** opens.
5. Click **Next**. The **Select Package Source** panel opens.

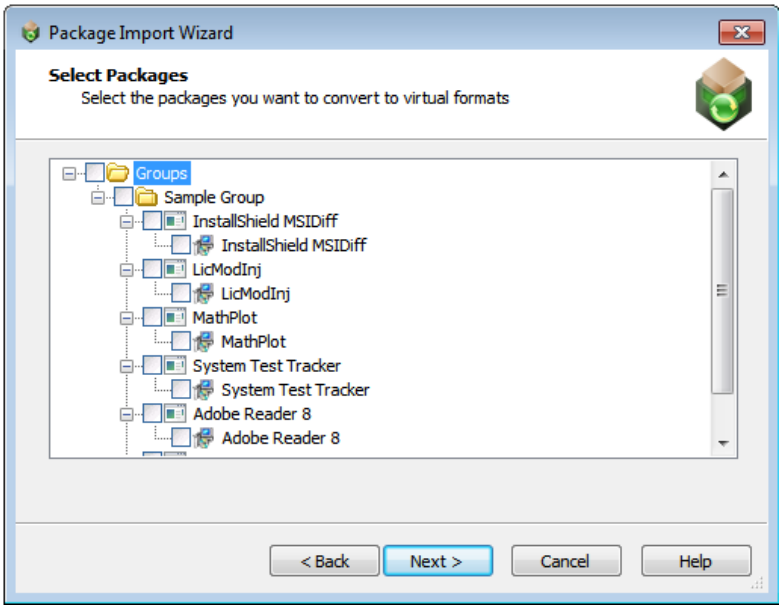


6. Select **AdminStudio Application Catalog** and click **Next**. The **Connect to an AdminStudio Application Catalog** panel opens.

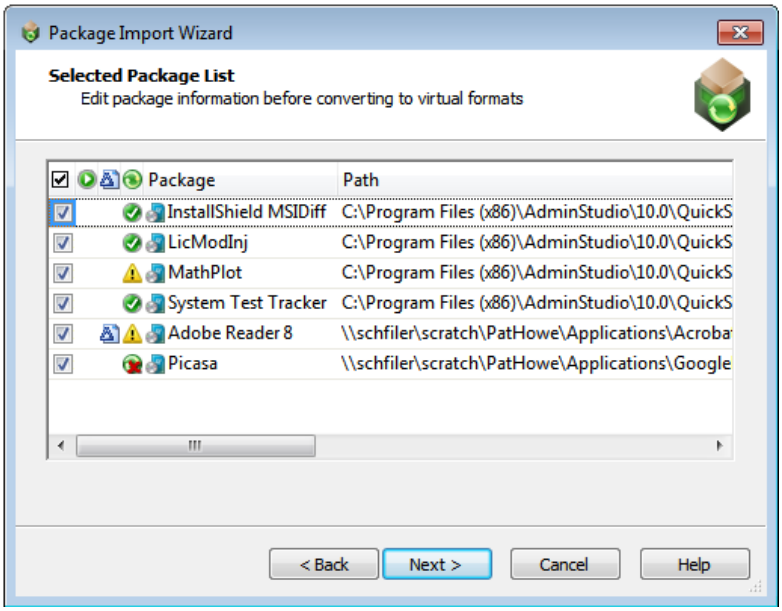


7. In the **Server** field, enter the name of the Server that you want to connect to.
8. From the **Authentication** list, select one of the following options:
   - **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Application Catalog.
   - **Server Authentication**—Choose this option if you want to use server login identification to log into this Application Catalogs server. Then enter the appropriate **Login ID/User name** and **Password**.
9. Enter the name of the existing AdminStudio Application Catalog database that you want to connect to in the **Catalog** field.

10. Click **Next**. The **Select Packages** panel opens, listing all of the packages found in the Application Catalog, but with none of them selected.



11. Select the packages that you want to add to this project and click **Next**. The **Selected Package List** panel opens.



An icon in the Virtualization Readiness column identifies whether the package requires repackaging prior to conversion to a virtual application:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. S ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

***Note** • If you want to override this setting, click in the **Virtualization Readiness** column and make a selection from the list.*

12. If a transform file (`.mst`) is located in the same directory as the selected Windows Installer package, one of the following icons is listed in the Transform column:

    - One transform is being added with this package.
    - Multiple transforms are being added with this package.

    If multiple transforms are associated with this package, you should click the browse button in the **Transform** column to open the **MST** dialog box and specify which transform files you want to add and the order that you want the transforms applied. For more information, see MST Dialog Box.

***Note** • You can also edit the listed transform files by clicking the browse button in the **Transform** field in the **Properties** window of the **Packages** tab after you have added the packages.*

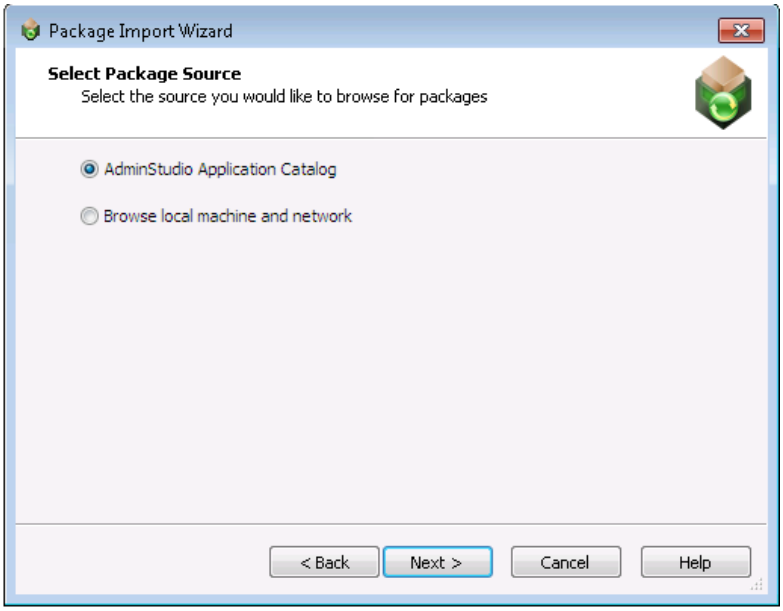
13. Make sure that the packages that you want to convert are selected and click **Next**. The **Package Import Wizard Complete** panel opens.
14. Click **Finish** to close the wizard and add the selected packages to your project.
15. To proceed with the conversion, see Performing a Conversion Using the Application Conversion Wizard.

Adding Packages from a Local Machine or Network

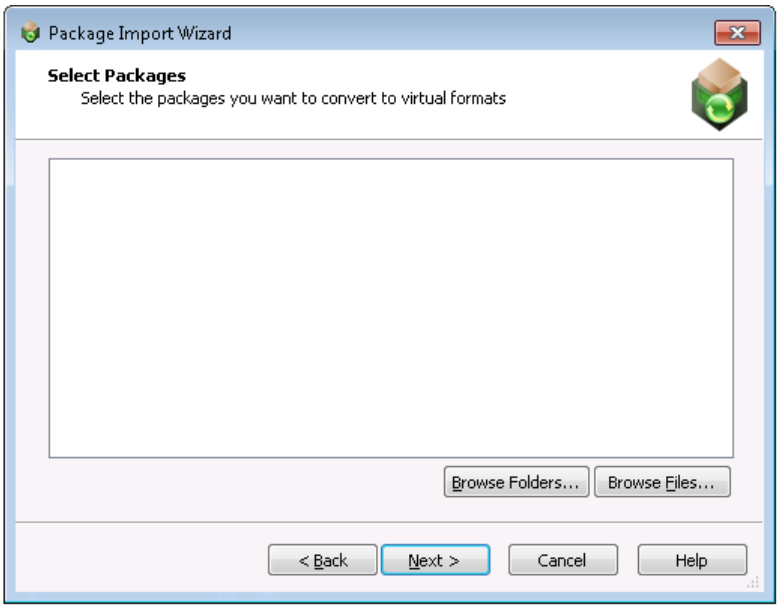
To add packages from a a local machine or network to Automated Application Converter so that you can convert them to virtual applications, perform the following steps:

Task ***To select packages from a local machine or network:***

1. Launch the Automated Application Converter.
2. Add one or more virtual machines, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
3. Open the **Packages** tab.
4. Click **Add Packages**. The **Package Import Wizard** opens.
5. Click **Next**. The **Select Package Source** panel opens.



6. Select **Browse local machine and network** and click **Next**. The **Select Packages** panel opens with no packages listed.



7. If you want to select one package to add, perform the following steps:

   a. Click **Browse Files**. The **Select Package Installation File** dialog box opens, prompting you to select the package you want to convert.

   b. Select the installation file (`.msi` or `.exe`) or installation script (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) you want to convert and click **Open**.The Automated Application Converter adds the selected package to the list on the **Select Packages** panel.

Note • *You can use installation scripts to run more complex installation scenarios.*

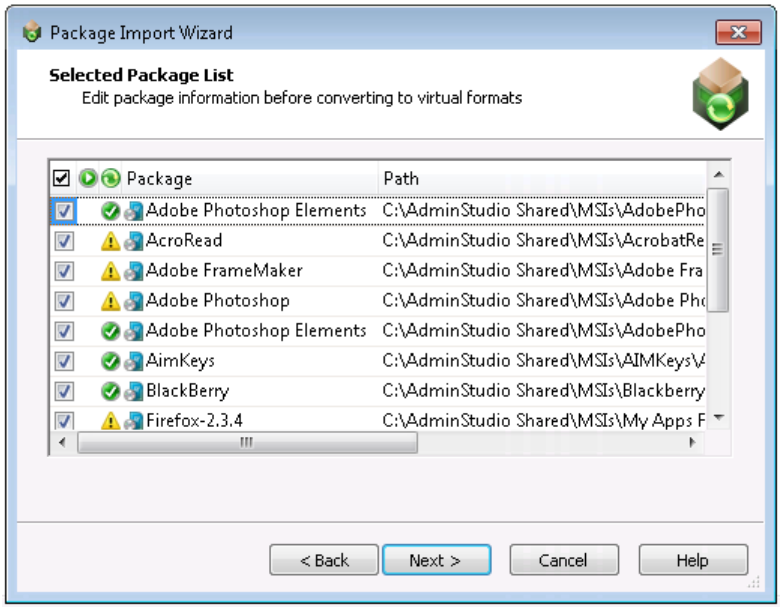
If you want to select a directory of packages to add, perform the following steps:

a. Click **Browse Folders.** The **Browse for Folder** dialog box opens, prompting you to select the directory containing the packages you want to convert.

b. Select the directory that contains the installation files (`.msi` or `.exe`) and/or installation scripts (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) you want to convert and click **Open**. The Automated Application Converter searches the selected directory and its subdirectories to locate the installation files and/or scripts and adds them to the list on the **Select Packages** panel.

Important • *The Automated Application Converter uses specific rules to determine which packages in the selected directory and its subdirectories would be added to the list on the **Select Packages** panel, and which of those files are automatically selected. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.*

8. On the **Select Packages** panel, click **Next**. The **Selected Package List** panel opens.



9. On the **Selected Package List** panel, an icon in the Virtualization Readiness column identifies whether the package requires repackaging prior to conversion to a virtual application:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*

Icon	Meaning	Description
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** • *For more information, see Application Virtualization Compatibility Tests.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

Note • *If you want to override this setting, click in the Virtualization Readiness column and make a selection from the list.*

10. Make sure that the packages that you want to convert are selected and click **Next**. The **Package Import Wizard Complete** panel opens.
11. Click **Finish** to close the wizard and add the selected packages to the project.
12. To proceed with the conversion, see Performing a Conversion Using the Application Conversion Wizard.

Editing Package Properties on the Packages Tab

By default, the list of packages on the **Packages** tab lists the **Package**, **Path**, and **Command Line** properties. Additional properties can be viewed and edited in the **Properties** window.

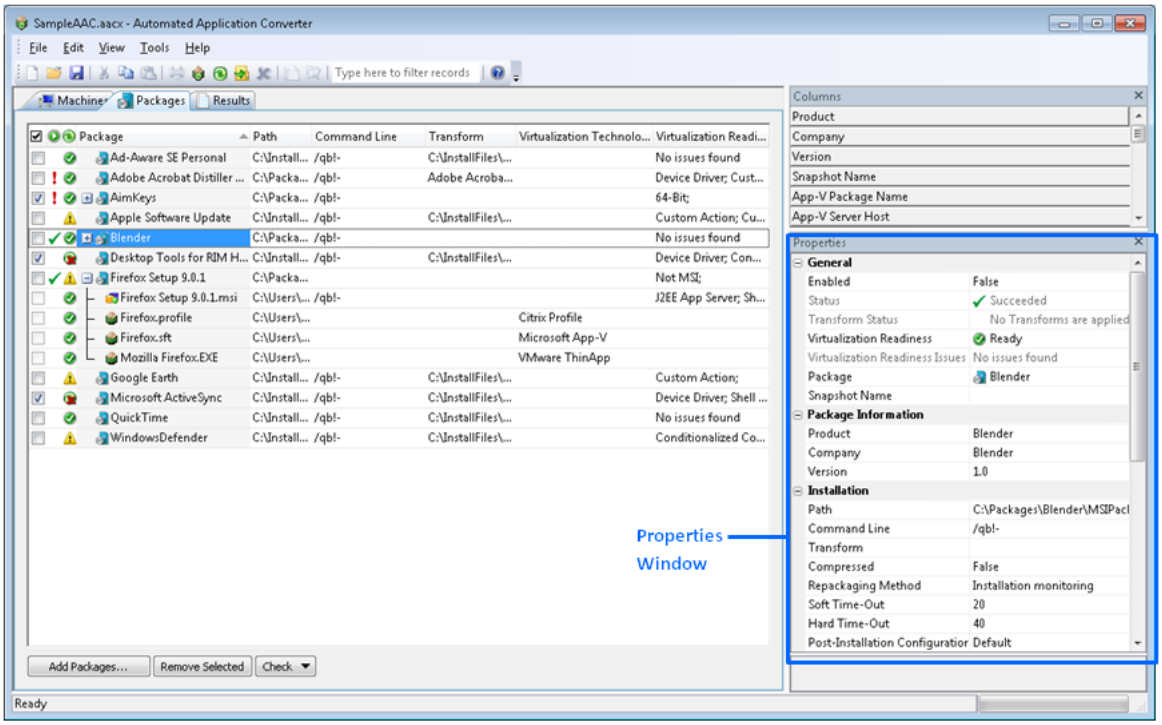


Figure 10-12: Properties Window on the Packages Tab

The properties that you may want to edit depend upon the virtual format or packages you are converting to:

- Setting MSIX Signing Options
- Setting General Package Properties
- Setting Package Properties for Conversion to App-V Format

Setting MSIX Signing Options

Before you use Automated Application Converter to convert a Windows Installer packages and legacy setup packages (.exe) to a MSIX packages, you must select either one of the following:

- Certificate File (.pfx)
- Certificate Store

Certificate File (.pfx)

In the **Properties** window, under **MSIX Signing Options**, select **Certificate File (.PFX)** from the **Select Certificate for Signing** list.

Enter the **Certificate File** path or browse and select the certificate path and then enter the **Password**.

Certificate Subject auto populates the certificate subject details.

Enter the valid **Time Stamp Server URL**. This allows packages to be accepted even after the certificate is no longer valid.

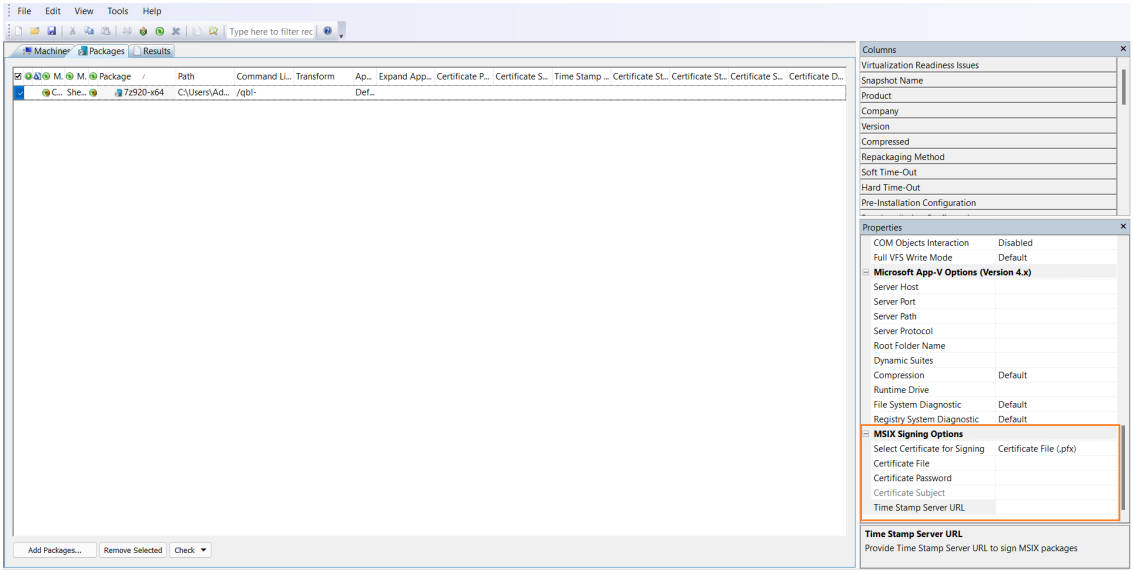


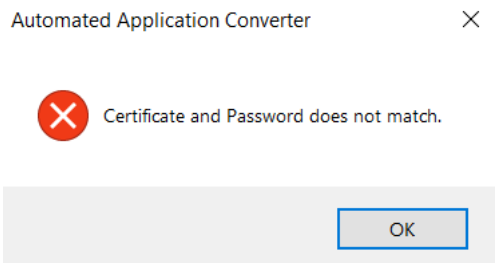
Figure 10-13: Select Certificate File as Signing Option

Note • *If different Certificate details are provided in both places, ie. under **Properties** window and **Tools >> Option**, then the package level certificate details will be taken as a priority.*

Note • *Before conversion make sure that the Company Name is matching with the Certificate Subject.*

Error Message

Before MSIX conversion, if **Certificate File** and **Password** details are invalid, the below error message appears.



Certificate Store

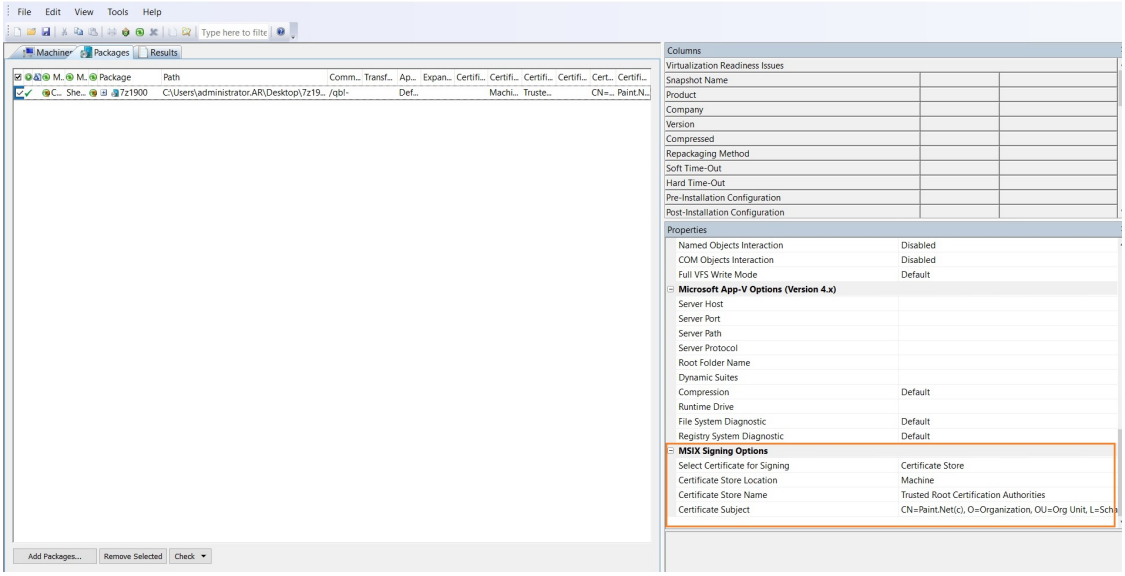
Note • *If you select the Certificate Store option, make sure that the Certificate has been imported. For more details on importing the certificate, see Import Certificate.*

In the **Properties** window, under **MSIX Signing Options**, select **Certificate Store** from the **Select Certificate for Signing** list.

Enter the below details:

- From the **Certificate Store Location** drop down, select either **User** or **Machine**.
- From the **Certificate Store Name** drop down, select store name from the list.
- From the **Certificate Subject** drop down, select certificate subject from the list.

Note • *Before conversion make sure that the Company Name is matching with the Certificate Subject.*



Setting General Package Properties

Before you use Automated Application Converter to convert a Windows Installer package to a virtual package, you may want to adjust the following property settings on the package:

- Specifying a Package's Repackaging Snapshot
- Editing the Installation Command Line
- Specifying a Package's Compression Setting
- Selecting the Repackaging Method
- Specifying Time Out Settings
- Enabling Manual Installation During Repackaging

- Enabling Pre-Installation and Post-Installation Configuration

Note • *Prior to using the Application Conversion Wizard to convert a package to a virtual package, you also need to prepare a virtual machine to use in the conversion as described in Preparing Your Virtual Machines for Use With the Automated Application Converter.*

Specifying a Package's Repackaging Snapshot

When you initiate the conversion of a package and repackaging is required, Automatic Application Converter will do the following to determine which snapshot on the selected virtual machine to use:

- **Snapshot Name provided**—If a snapshot name is entered in the virtual machine's **Snapshot Name** property on the **Machines** tab, Automated Application Converter will attempt to launch that snapshot.
- **Snapshot Name not provided**—If no snapshot name is entered in the virtual machine's **Snapshot Name** property on the **Machines** tab, Automated Application Converter will attempt to launch a snapshot named `AutoRepack_Base`.

If you want to convert an individual package using a different snapshot, you need to enter the snapshot name in the package's **Snapshot Name** property on the **Packages** tab.

Task ***To specify a package's repackaging snapshot:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Machine Settings**, enter a snapshot name in the **Snapshot Name** field.

Editing the Installation Command Line

When you add a package to the **Packages** tab, the command line parameters that are needed to silently install this package are entered in the package's **Command Line** property. By default, the **Command Line** property's value is:

```
/qb!-
```

If you would like to edit this command line, perform the following steps.

Task ***To specify a package's installation command line:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, edit the parameters in the **Command Line** field.

Specifying a Package's Compression Setting

You can specify whether a package is compressed by making a selection in the package's **Compressed** property on the **Packages** tab.

If a package is compressed, only the single installation file will be copied to the virtual machine for repackaging. If a package is not compressed, the entire folder tree, including the selected package file, will be copied to the virtual machine.

To specify a package's compression setting, perform the following steps:

Task ***To specify a package's compression setting:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the **Compressed** property to one of the following values:
   - **False**—Package is not compressed. The entire folder tree, including the selected package file, will be copied to the virtual machine.
   - **True**—Package is compressed. Only the single installation file will be copied to the virtual machine for repackaging.

Selecting the Repackaging Method

You can specify the repackaging method that you want to use when performing repackaging of this package by making a selection in the package's **Repackaging Method** property on the **Packages** tab.

To specify a package's repackaging method setting, perform the following steps:

Task ***To specify a package's repackaging method setting:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the **Repackaging Method** property to one of the following values:
   - **Installation monitoring**—Repackager monitors system changes as a package is installed, and that data is converted into a Windows Installer package.
   - **Single-step snapshot**—Repackager first takes an initial system snapshot, then runs the installation, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package.

Note • *For more information, see Repackaging Methods.*

Specifying Time Out Settings

You can specify the length of time during the package installation portion of repackaging that you want to permit to elapse before you are notified (**Soft Time-Out** property) or the length of time before Automated Application Converter considers the process to be a failure (**Hard Time-Out** property) on the **Packages** tab.

To specify a package's time out settings, perform the following steps:

Task ***To specify a package's time out settings:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the following properties:
   a. **Soft Time-Out**—Set this property to the number of minutes that you want to allot for the package to install before the user would be notified. After this time period elapses, the user will be notified, just in case there are pending dialogs for the user to dismiss or if some other user interaction is required. The default value is `20`.
   b. **Hard Time-Out**—Set this property to the number of minutes that you want to allot for the package to install before it is considered a failure. If this time period elapses, Automated Application Converter would consider the installation a failure and would move to the next package. The default value is `40`.

Enabling Manual Installation During Repackaging

Rather than have Automated Application Converter automatically perform silent package installation on the virtual machine during repackaging, you can choose to perform this installation manually. Manual installation can be used for more complex installations such as installations that require user input or installations which consist of more than one executable file. You can specify manual installation of a package by setting the package's **Manual Install** property on the **Packages** tab.

Important • *The **Manual Install** property is ignored during App-V 5.0 conversion using the Microsoft App-V Sequencer.*

To specify manual installation of a package, the following steps:

Task ***To specify manual installation of a package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the Manual Install property to one of the following values:
   - **Disabled**—Disable manual installation.
   - **Enabled**—Enable manual installation.

Documenting Interactive Repackaging Steps Using the Microsoft Step Recorder Tool

You can use the Microsoft Steps Recorder documentation tool with Automated Application Converter to automatically record the step-by-step actions that you take on the virtual machine during repackaging. This information, which is saved in a web archive (`.mht`) file, includes a text description of where you clicked on each screen, along with a screen capture for each click.

This feature may be useful when you are attempting to repackage a complex installer which requires user interaction on the virtual machine that Automated Application Converter launches to perform repackaging.

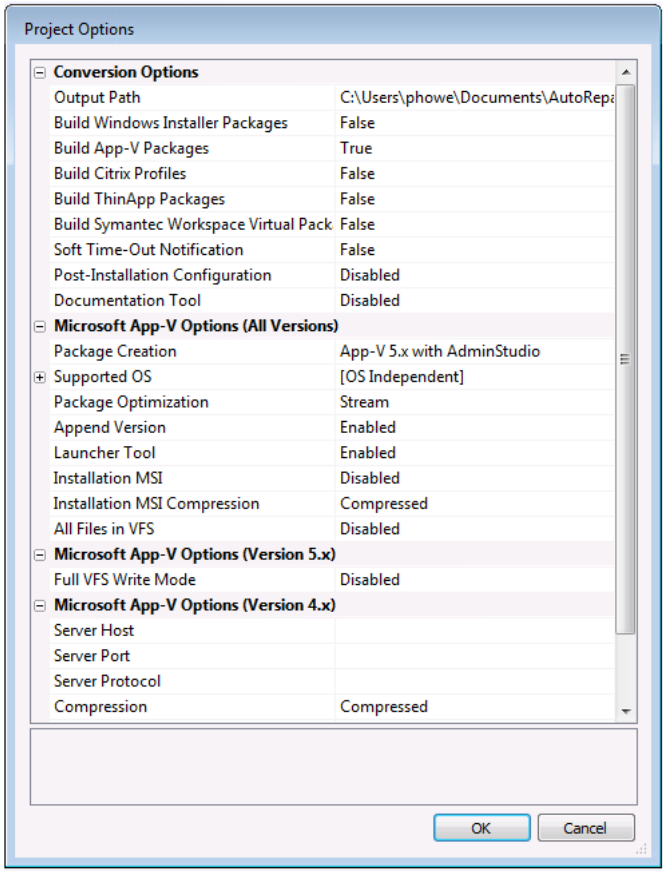
Note • *If repackaging is performed silently, without user interaction, no steps are recorded and no web archive file is created.*

Enabling the Documentation Tool Globally

To enable the Microsoft Steps Recorder documentation tool for all packages, perform the following steps:

Task ***To enable the Microsoft Steps Recorder documentation tool for all packages:***

1. On the **Tools** menu, click Options. The **Project Options** dialog box opens.



2. Under **Conversion Options**, set **Documentation Tool** to **Enabled**.
3. Click **OK**.

Enabling the Documentation Tool for an Individual Package

To enable the Microsoft Steps Recorder documentation tool for an individual package, perform the following steps:

Task ***To enable the documentation tool for a package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the **Documentation Tool** property to **Enabled**.

Note • *If the* ***Documentation Tool*** *is set to* ***Default****, this package will use the* ***Documentation Tool*** *setting that is defined on the* ***Project Options*** *dialog box.*

Reviewing the Recorded Web Archive File

After you have repackaged a package using Automated Application Converter which required you to interact with the repackaging process on the virtual machine, a web archive file will be created. This recorded web archive file will be copied to the following directories:

```
AutoRepack/Repackaged
AutoRepack/MSI Virtual
AutoRepack/VirtualFormatPackage
```

Note • *The location of the* AutoRepack *directory is specified on the* ***Project Options*** *dialog box in the* ***Output Path*** *field under* ***Conversion Options****.*

To review this recorded web archive file, perform the following steps:

Task ***To review a web archive file:***

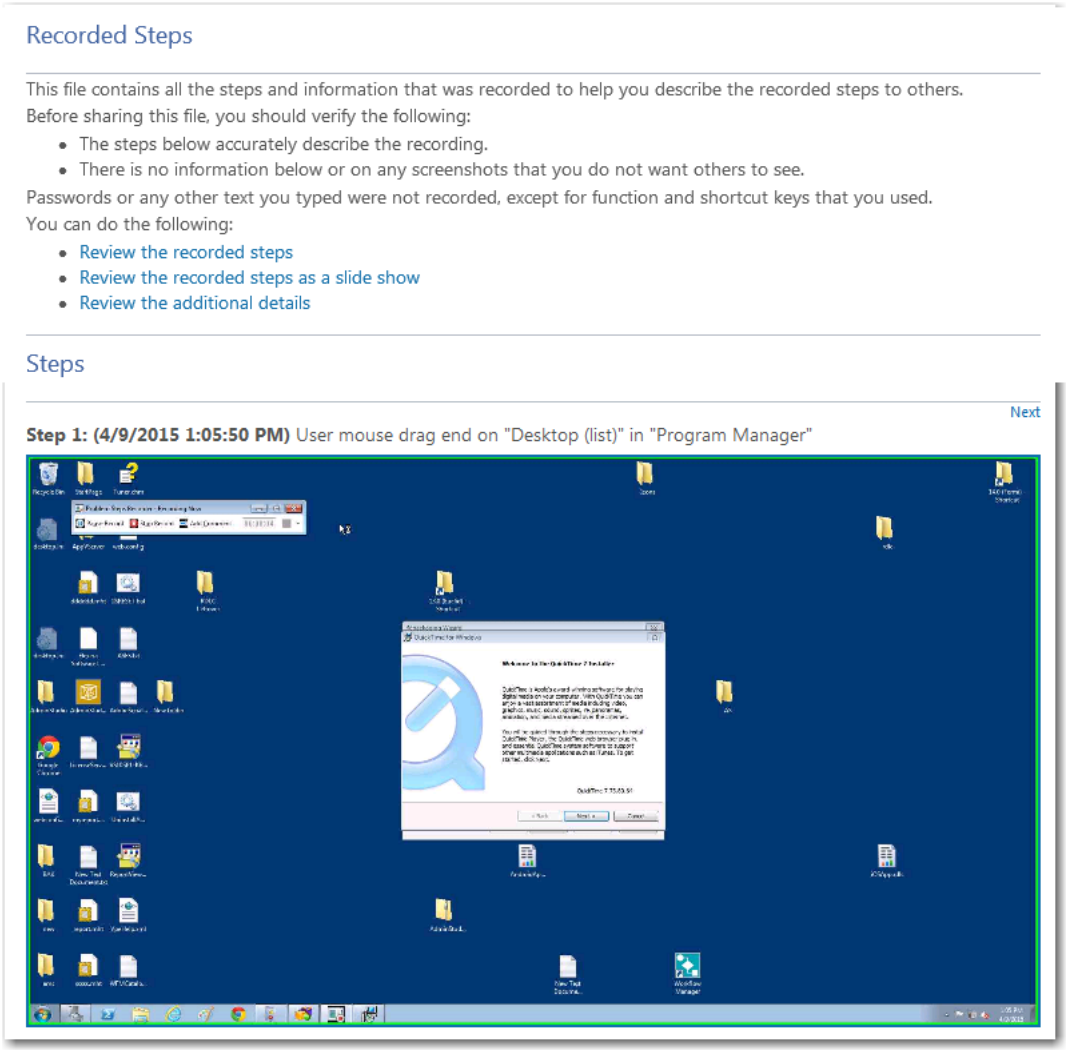
1. Open the appropriate directory and locate the following web archive (`.mht`) file:

   *InstallerName*_Recording_*YYYYMMDD_TIME*.mht

   For example:

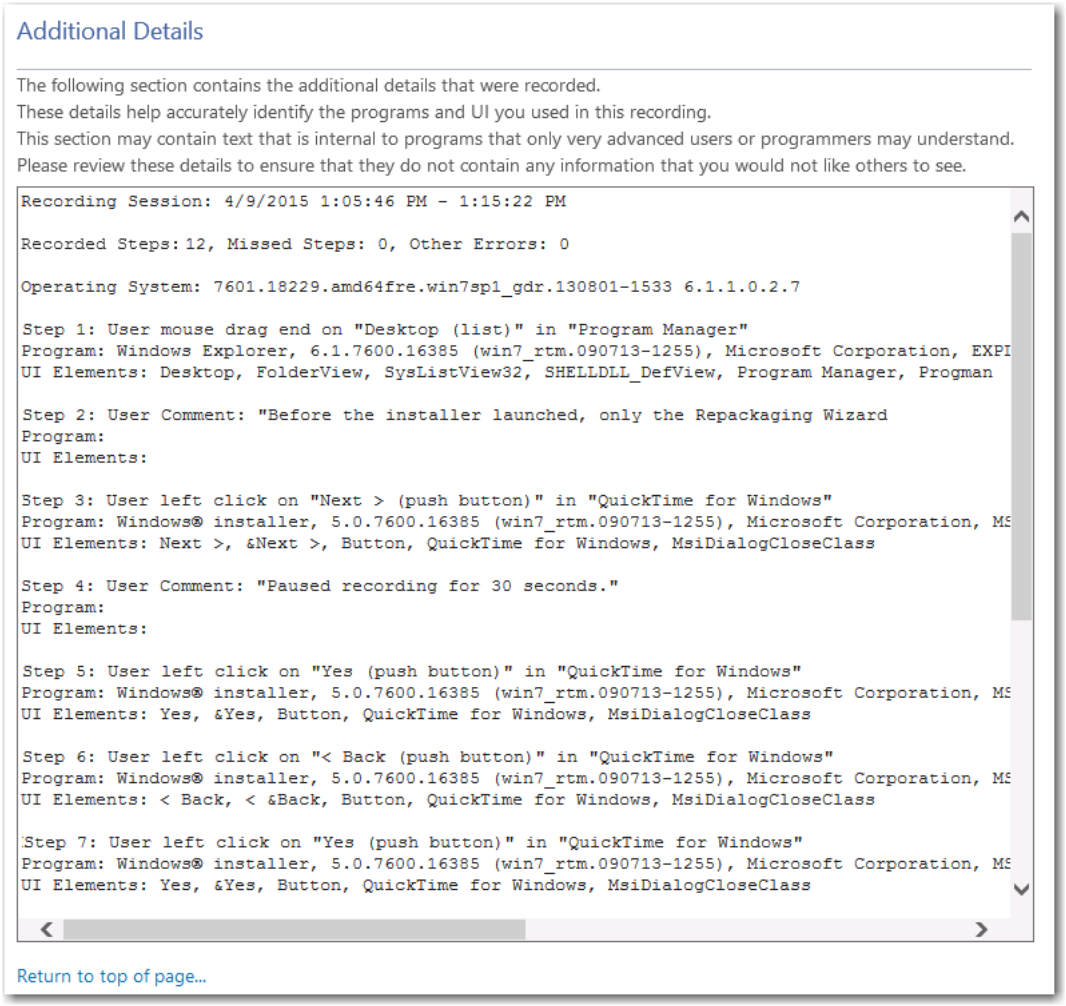
   QuickTime_Recording_20150409_1015.mht

2. Double-click the file to open it. The file opens in a browser window.
3. In the **Recorded Steps** section, scroll down to view all of the steps that you performed during repackaging along with screen captures of each step.



Tip • *If you want to view all of the screens as a slide show instead of scrolling through them, click* ***Review the recorded steps as a slide show****.*

4. Review the information in the **Additional Details** area, which contains a text description of the steps that were taken, along with information that is internal to the application for which repackaging was performed.



Enabling Pre-Installation and Post-Installation Configuration

You can instruct Automated Application Converter to pause during repackaging to enable you to manually perform configuration steps either prior to package installation on the virtual machine or after installation.

- Enabling Pre-Installation Configuration
- Enabling Post-Installation Configuration

Enabling Pre-Installation Configuration

When converting some packages, you may want to perform some manual configuration steps prior to installing that package during the repackaging process. This could be useful when a particular dependency, such as Java runtime, needs to be installed and should not be captured as part of the application capture process.

To enable pre-installation configuration of a package, the following steps:

Task ***To enable pre-installation configuration:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the **Pre-Installation Configuration** property to one of the following values:
   - **Disabled**—Disable pre-installation configuration.
   - **Enabled**—Enable pre-installation configuration.

Enabling Post-Installation Configuration

When converting some packages, you may want to perform some manual configuration steps just after the package is installed on the virtual machine during the repackaging process but before it is converted to the target formats. For example, you may want to manually launch the application and perform some “first use” selection steps.

Important • *The **Post-Installation Configuration** property is ignored during App-V 5.0 conversion using the Microsoft App-V Sequencer.*

To enable post-installation configuration of a package, the following steps:

Task ***To enable post-installation configuration:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Installation**, set the **Post-Installation Configuration** property to one of the following values:
   - **Default**—Use the project-level behavior that is specified through the **Post-Installation Configuration** field on the **Project Options** dialog box. This is the default value.
   - **Disabled**—Disable post-installation configuration. The repackaging process does not pause after installing the product.
   - **Enabled**—Enable post-installation configuration. The repackaging process pauses after the installation of the product to allow you to launch the product and set up various application settings such as update settings and file associations. You can also perform other system configuration tasks. Once you are done with configuration, you can click a button to have the repackaging proceed with the capture and convert process.

     Important • *If you select the **Enabled** option, ensure that the value that you enter for the **Hard Time-Out** setting allows enough time to configure the application.*

Setting Package Properties for Conversion to App-V Format

Before you use Automated Application Converter to convert a Windows Installer package to an App-V 4.x or 5.x package, you may want to adjust the following property settings on the package:

- Overriding the Name of the App-V Package
- Selecting the App-V Conversion Method
- Specifying the App-V Package's Primary Application Directory
- Specifying the App-V Package's Supported Operating Systems
- Specifying How to Optimize the App-V Package
- Specifying Whether to Append the Version Number to the App-V Package File Name
- Specifying the Diagnostic Tools to Include With the App-V Package
- Choosing to Expand the App-V 5.0 Package Before Sequencing
- Entering Comments for an App-V Package
- Setting the App-V 4.x Package's Server Location
- Specifying the App-V Package's Root Folder Name
- Enabling Dynamic Suiting for an App-V 4.x Package
- Specifying an App-V 4.x Package's Compression Setting
- Designating an App-V Package as an Upgrade
- Specifying an App-V 4.x Package's Client Runtime Drive
- Setting an App-V Package's VFS Options

Overriding the Name of the App-V Package

By default, the App-V package is given the name of its Windows Installer package (which is displayed in the **Product** field under **Package Information** in the **Properties** window).

For example, if virtual package contains multiple applications, you could specify a name that identifies the entire package. For example, **`Microsoft Office`** could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.

To override the default name of an App-V package, perform the following steps:

Task ***To override the name of an App-V package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, enter a new name in the **Name** field (a maximum of 64 characters).

Selecting the App-V Conversion Method

When using Automated Application Converter to convert a Windows Installer package to App-V format, you have the option of converting it to either an App-V 4.x or 5.0 package. You can also choose whether to use AdminStudio's virtual converter or the Microsoft App-V Sequencer to convert a package to App-V 5.0 format. You make these specifications by setting the **Package Creation** property.

To set the **Package Creation** property, perform the following steps:

Task ***To specify a package's App-V conversion method:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the **Package Creation** property to one of the following values:
   - **Default**—Use the method that is selected in the **Package Creation** field on the **Project Options** dialog box. This is the default value.

     Note • *For more information, see Setting Default Project Properties.*
   - **App-V 4.6 with AdminStudio**—When converting this package to App-V format, use AdminStudio to convert it to an App-V 4.6 package.
   - **App-V 5.x with AdminStudio**—When converting this package to App-V format, use AdminStudio to convert it to an App-V 5.x package.
   - **App-V 5.x with Sequencer**—When converting this package to App-V format, use Microsoft Sequencer to convert it to an App-V 5.x package.

Note • *For more information, see Comparison of the App-V 5.0 Conversion Methods,*

Specifying the App-V Package's Primary Application Directory

You can specify the App-V package's primary application directory on the package's **Properties** window.

To specify an App-V package's supported operating systems, perform the following steps:

Task ***To specify an App-V package's primary application directory:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the **Primary Application Directory** property to one of the following values:

- **App-V 5.x conversions using the Sequencer**—Specify the absolute folder path to the expected main installation location of the package to be converted. If no value is specified, then the App-V 5.0 package will be created with all files in the virtual file system (VFS) folder.

  Important • *When you use the **App-V 5.x with Sequencer** option, it is highly recommended that you enter a value for the **Primary Application Directory** property. If you do not, then all of the converted files will end up in the virtual file system (VFS) folder.*

- **App-V 4.x or 5.x conversions using AdminStudio**—Specify the main installation directory which will be used to set up the root/mount folder mapping.

  For example, for Yahoo Messenger, AdminStudio automatically detects `C:\Program Files\Yahoo!` as the primary installation directory. However, you may prefer that the primary installation directory be `C:\Program Files\Yahoo!\Messenger`, because this directory is more correct for Messenger.

  In this case, you can enter this new path in the **Primary Application Directory** property field, and it will be honored by the AdminStudio converter as long as this path exists in the Windows Installer Package. If it does not exist, then AdminStudio will fall back to use the directory it found during automatic detection.

  Note • *When using the **App-V 4.x/5.x with AdminStudio** package creation options, this field is optional.*

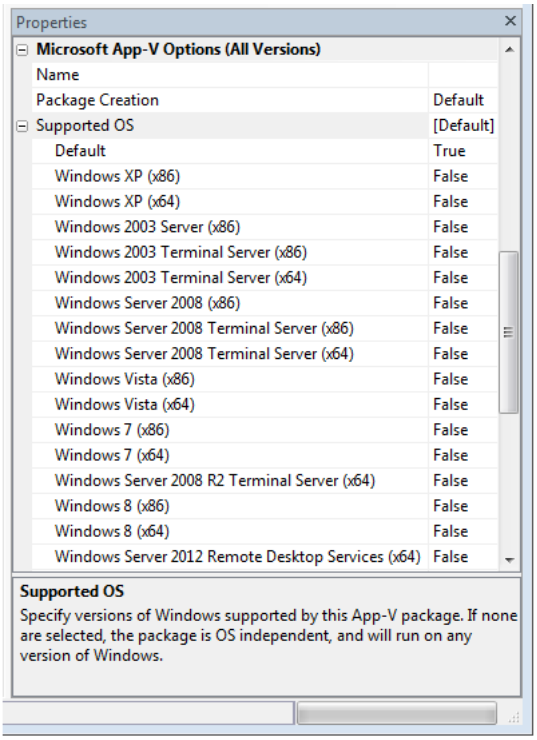
Specifying the App-V Package's Supported Operating Systems

You can specify the operating systems that the App-V package will support by editing the **Supported OS** property on the package's **Properties** window.

To specify an App-V package's supported operating systems, perform the following steps:

Task ***To specify an App-V package's supported operating systems:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, expand the **Supported OS** property to display the available operating systems.



3. Make the following selections:

   - **To accept default values**—To accept the default values for **Supported OS** that are set on the **Project Options** dialog box, set **Default** to **True**. When you make this selection, all operating systems and **OS Independent** will automatically switch to **False**, and the word `[Default]` will be listed next to **Supported OS**.
   - **If the App-V package is operating-system-dependent** (meaning that it only supports some of the listed operating systems), select **True** next to the supported operating systems. If any of the listed operating systems are set to **True**, the value for **Default** and for **Supported OS** will automatically switch to **False**, and the selected operating systems will be listed in brackets next to **Supported OS**.
   - **If the App-V package is operating-system-independent** (meaning that it supports all listed operating systems), set **OS Independent** to **True**. When you make this selection, all operating systems and **Default** will automatically switch to **False**, and `[OS Independent]` will be listed next to **Supported OS**.

Important • *When setting the **Supported OS** property for App-V 5.0 packages, keep in mind that the packages are limited to the supported operating systems of the App-V 5.0 client:*

- *Windows 7 and later*
- *Windows Server 2008 R2 and later*

Specifying How to Optimize the App-V Package

You can use the **Package Optimization** property to control the performance and network traffic associated with running an App-V package. The package optimization option you select determines how quickly the App-V package will launch, and how often additional functionality will need to be streamed to the client while the App-V package is being used.

The files in an App-V package can be grouped into two feature blocks:

- **Feature block 1**—Feature block 1 must contain the core functionality of the App-V package that is necessary to launch the application. At application launch, all of the files in feature block 1 are streamed to the client in one unit.
- **Feature block 2**—Feature block 2 can contain additional functionality of the App-V package that is not necessary to launch the application. While the App-V package is being used, the files in feature block 2 can be streamed in small packets on an as-needed basis.

By setting the **Package Optimization** property, you can either choose to include all App-V package files in feature block 1 (**Offline** option), to improve launch speed, you can choose to group the files into two feature blocks: feature block 1 and feature block 2 (**Stream** option).

To specify an App-V package's optimization settings, perform the following steps:

Task ***To specify an App-V package's optimization settings:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the Package Optimization property to one of the following values:
   - **Default**—Use the method that is selected in the **Package Optimization** field on the **Project Options** dialog box. This is the default value.
   - **Offline**—When the package is optimized for offline use, the entire package is included in feature block 1. If you choose this option, all files in the App-V package will be included in feature block 1 and will be streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable users to use the App-V package when not connected to the App-V server and if you want to eliminate network traffic when the App-V package is being used.

     Note • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*
   - **Stream**—When the package is optimized for streaming use, only the shortcut targets which are included in feature block 1 are streamed to the client at start up. Feature block 2 can contain additional functionality of the App-V package that is not necessary to launch the application. While the App-V package is being used, the files in feature block 2 are streamed in small packets on an as-needed basis. This option provides a relatively quick launch time while limiting network traffic during application use.

     Note • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent use of the application.*

Specifying Whether to Append the Version Number to the App-V Package File Name

You can use the **Append Version** property to specify whether to want to append the package version number to the App-V package file name.

Task ***To specify whether to append the version number to the App-V package file name:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the **Append Version** property to one of the following values:
   - **Enabled**—Append the package version to the SFT file name.
   - **Disabled**—Leave the package version off of the SFT file name.
   - **Default**—Use the setting that is defined on the **Project Options** dialog box.

Specifying the Diagnostic Tools to Include With the App-V Package

You can choose to include the following diagnostic tools with your App-V package:

- **Launcher Tool**—The App-V Package Launcher can be used to quickly publish the App-V package to the local machine for testing. This tool is available for both App-V 4.x and 5.x packages. You can use the App-V package Launcher to test a newly built App-V package before moving it to a deployment server.
- **File System Diagnostic**—The Windows Command Prompt tool enables you to look at the file system for the application while it is running in its virtual environment. You can use it to investigate the file system and launch other tools within the virtual environment context. This tool is only available for App-V 4.x packages.
- **Registry System Diagnostic**—The Registry Editor tool enables you to look at the registry for the application while it is running in its virtual environment. This tool is only available for App-V 4.x packages.

You can use the App-V Package Launcher to test a newly built App-V package before moving it to a deployment server. And if, during testing, you received an error message stating that the application cannot load a DLL, you could use the **File System Diagnostic** and **Registry System Diagnostic** tools to troubleshoot the problem.

Tip • *Because the App-V 5.x Application Launcher has a built-in Windows Command Prompt tool, it is not necessary to include the* ***File System Diagnostic*** *or* ***Registry System Diagnostic*** *tools with App-V 5.x packages.*

To specify which diagnostic tools to include with an App-V package, perform the following steps:

Task ***To specify the diagnostic tools to include with an App-V package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the **Launcher Tool** property to one of the following values:
   - **Enabled**—Include the App-V Package Launcher when you build an App-V package.
   - **Disabled**—Do not include the App-V Application Launcher when you build an App-V package.
   - **Default**—Use the setting on the **Project Options** dialog box.

3. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, set the **File System Diagnostic** property to one of the following values:

   - **Enabled**—Include the Windows Command Prompt application with your App-V package so that you can browse the virtual file system at runtime from within the virtual environment. If this option is selected, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment. You can use `Virtual File System.osd` to view the existing files and folders on the computer plus the files and folders for the virtual package. A shortcut to the command prompt will be added to the App-V package.
   - **Disabled**—Do not include the Windows Command Prompt application with your App-V package.
   - **Default**—Use the setting on the **Project Options** dialog box.

4. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, set the **Registry System Diagnostic** property to one of the following values:

   - **Enabled**—Include the Registry Editor (`regedit.exe`) with your App-V package so that you can browse the registry at runtime from within the virtual environment. If this option is selected, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment. You can use `Virtual Registry.osd` to view the existing registry on the computer plus the registry for the virtual package. A shortcut to the registry option will be added to the App-V package.
   - **Disabled**—Do not include the Registry Editor (`regedit.exe`) application with your App-V package.
   - **Default**—Use the setting on the **Project Options** dialog box.

Choosing to Expand the App-V 5.0 Package Before Sequencing

You can use the **Expand App-V Package** property to specify whether to want to expand an existing App-V 5.x package on the system before performing sequencing. This is useful for specifying middleware and dependency App-V packages such as Java runtime.

Note • *This option is only used when **App-V 5.x with Sequencer** is the chosen package conversion method.*

Task ***To choose to expand an App-V 5.0 package before sequencing:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (Version 5.x)**, click the browse button next to the **Expand App-V Package** property and select the existing App-V package to expand on the system before performing the sequencing.

Entering Comments for an App-V Package

You can use the **Comments** property to specify text for the App-V package comments/description section.

Note • *This setting is ignored when using the* ***App-V 5.x with Sequencer*** *package conversion method.*

Task ***To enter comments for an App-V package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, enter a brief description of the App-V package. This text will appear in the App-V package comments/description section.

Setting the App-V 4.x Package's Server Location

To set the server location for an App-V 4.x package, perform the following steps:

Task ***To set the server location for an App-V 4.x package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, specify the following properties:

Property	Description
Server Host	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter **%SFT_SOFTGRIDSERVER%** to indicate an environment variable. ***Note*** • *If you enter* **%SFT_SOFTGRIDSERVER%**, *you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Server Port	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.
Server Path	Specify the relative path on the virtual application server where the software package is stored and from which it will be streamed. ***Note*** • *This information is required to create a package if the* `.sft` *file will be stored in a subdirectory of* `CONTENT`; *otherwise, this information is not required.*

Property	Description
Server Protocol	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: • **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. • **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. • **FILE**—The App-V package are streamed from a file share. • **HTTP**—The hypertext transport protocol streams the App-V package. • **HTTPS**—The secure hypertext transport protocol streams the App-V package.

Specifying the App-V Package's Root Folder Name

You can use the **Root Folder Name** property to override the unique 8.3 root folder name for a product and version. During run time, the Application Virtualization Client mounts the package's file system to the App-V virtual drive; the Q drive is the default. The long and short names of the root folder must be unique because two packages with the same root folder name cannot be deployed simultaneously.

To specify the **Root Folder Name** property, perform the following steps:

Task ***To specify the Root Folder Name property:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, specify the root folder of the App-V package's file system in the **Root Folder Name** property.

Enabling Dynamic Suiting for an App-V 4.x Package

You can use the **Dynamic Suites** property to enable dynamic suiting for an App-V package.

The point of application virtualization is to minimize the system dependencies that an application has on the underlying physical system. Many applications have common system dependencies on plug-ins or middleware, such as Adobe Reader or ODBC drivers.

Dynamic Suite Composition (DSC) is a Microsoft Application Virtualization feature that enables applications to be virtualized separately from the plug-ins and middleware applications that they rely on, while still enabling them to communicate with those plug-ins and middleware applications within the virtual environment. The primary App-V package and the dependency App-V packages in the dynamic suite will run and interact with one another as if they were all installed locally on a computer. You would only need to deploy common system components once on each client, making them available for use by many App-V packages, rather than to include them with each of the App-V packages that are dependent upon them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary App-V packages.

To enable dynamic suiting for an App-V package, perform the following steps:

Task ***To enable dynamic suiting:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, locate the **Dynamic Suites** property and enter a semicolon-delimited list of OSD or SFT files to be dynamically suited with this package, or click the ellipsis button (...) and select the OSD or SFT files to be suited. If a file must be present for this package to work properly, append the following to the file name:

   ```
   :MANDATORY
   ```

Specifying an App-V 4.x Package's Compression Setting

You can use the **Compression** property to specify whether to compress the data files in this App-V package.

Task ***To specify whether to compress an App-V 4.x package:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (Version 4.x)**, set the **Compression** property to one of the following options:
   - **Compressed**—Compress the App-V package.
   - **Uncompressed**—Do not compress the App-V package.
   - **Default**—Use the **Compression** property option that is selected on the **Project Options** dialog box.

Designating an App-V Package as an Upgrade

You can use the Upgrade Package property to indicate that this App-V package is an upgrade to a previous package. To do this, you select the previous App-V package.

Note • *This setting is ignored when using the* ***App-V 5.x with Sequencer*** *package conversion method.*

To designate an App-V package as an upgrade, perform the following steps:

Task ***To designate an App-V package as an upgrade:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, locate the **Upgrade Package** property.
3. If this package is an upgrade package that should update an earlier version of the application, click the browse button (...) browse to the earlier version.

Specifying an App-V 4.x Package's Client Runtime Drive

You can use the **Runtime Drive** property to specify an App-V 4.x package's client runtime drive.

Task ***To specify an App-V package's client runtime drive:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, enter the App-V client runtime drive in the **Runtime Drive** field.

   If you do not enter a value, one of the following values will be used:

   - **If a value is set on the Project Options dialog box**, that value will be used.
   - **If no value is set on the Project Options dialog box**, the default value of `Q:\` will be used.

Setting an App-V Package's VFS Options

You can use the **All Files in VFS** property to specify whether the App-V package's files should exist in Windows retargetable folders within the virtual file system (VFS) folder of the App-V package.

Task ***To set an App-V package's VFS options:***

1. On the **Packages** tab, select the package you want to edit. Package properties are displayed in the **Properties** window.
2. In the **Properties** window, under **Microsoft App-V Options (All Versions)**, set the **All Files in VFS** property to one of the following options:

   - **Disabled**—All of the files that correspond with the main installation directory of the application will be put in the root folder of the App-V package's file system. This method adheres to Microsoft App-V best practices. This is the default setting.
   - **Enabled**—All of the files in the package will be put into Windows retargetable folders within the VFS (virtual file system) folder of the App-V package. This option overrides the effect of specifying a value for **Primary Application Directory**.

     Note • *This option is generally not recommended, but there may be applications for which it is necessary. For example, if a virtualized application does not work as expected, and if it is possible that the application cannot find one of its files because it is searching in a hard-coded path, you may want to select the* ***Enabled*** *option.*

About Repackaging Windows Installer Packages

As a general rule, Windows Installer setups should not be repackaged. Instead, they should either be edited in InstallShield Editor, or, as Microsoft recommends, by creating a transform.

However, some IT organizations may elect to repackage Windows Installer packages in order to simplify them, which should make them more reliable and less likely to violate the organization's and Microsoft's recommended best practices. You can use the Automated Application Converter to automatically repackage a group of Windows Installer packages by selecting the **Windows Installer Packages (*.msi)** option on the **Select Output Formats** panel of the Application Conversion Project Wizard and the Application Conversion Wizard.

If you choose to repackage a Windows Installer package, you need to keep in mind that you may no longer be able to:

- Directly deploy vendor-provided patches for this package, OR
- Use any vendor-provided automatic updating service for this package.

Therefore, you should only consider repackaging a Windows Installer package if your IT staff is also willing to invest resources into periodically repackaging that application's vendor patches into an updated Windows Installer package.

***Note** • Tightly-controlled organizations probably would not want to have automatically-updating software, so the inability to use an automatic updating service may not be of concern to them.*

Using the Application Conversion Wizard to Perform Automated Package Conversion

Before you can perform conversion using the Application Conversion Wizard, you need to have already added virtual machines and packages to Automated Application Converter, as described in the following tasks:

- Adding Virtual Machines Using the Virtual Machine Import Wizard
- Adding Packages from an AdminStudio Application Catalog
- Adding Packages from a Local Machine or Network

This section explains how to use the Application Conversion Wizard to perform a conversion run using selected packages and virtual machines that you have already added to Automated Application Converter:

- Performing a Conversion Using the Application Conversion Wizard
- Viewing Conversion Results

***Note** • You also have the option of adding virtual machines, adding packages, and performing conversion during the same wizard run by using the **Application Conversion Project Wizard**, as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion.*

***Note** • Automated Application Converter now support Windows Services for MSIX Packages. A Windows Service installed while converting a legacy package format (MSI/EXE) will be captured and packaged into the MSIX package upon conversion.*

Performing a Conversion Using the Application Conversion Wizard

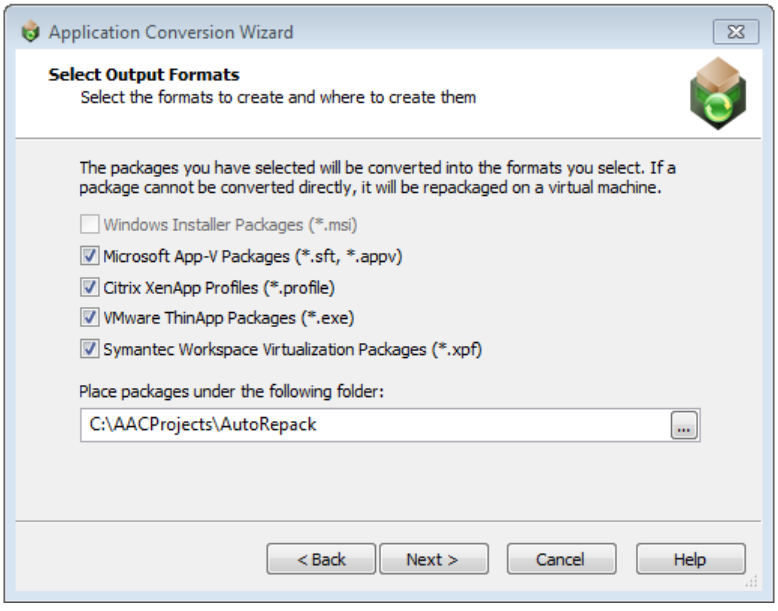
To use the Application Conversion Wizard to perform a conversion run using the selected packages and virtual machines, perform the following steps.

Task ***To perform a conversion using the Application Conversion Wizard:***

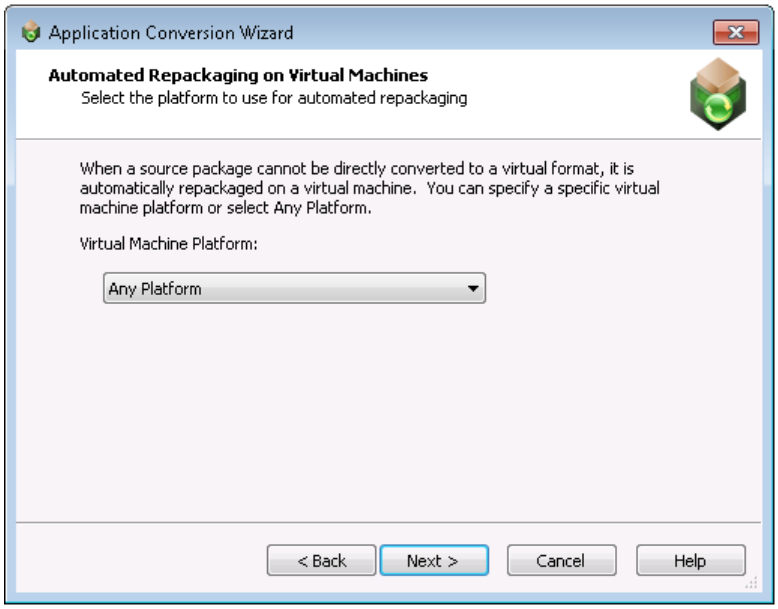
1. Select virtual machines to use for automated repackaging, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
2. Select packages to convert to virtual applications or to repackage, as described in Adding Packages from an AdminStudio Application Catalog or Adding Packages from a Local Machine or Network.
3. Set package properties as described in Setting MSIX Signing Options, Setting General Package Properties and Setting Package Properties for Conversion to App-V Format.

   Important • *Please note the following:*

   - *When converting to App-V format, you need to first indicate the App-V version you want to convert to and the conversion method you want to use by setting the* ***Package Creation*** *property to one of the following values (as described in Selecting the App-V Conversion Method): App-V 4.6 with AdminStudio, App-V 5.x with AdminStudio, or App-V 5.x with Sequencer.*
   - *When converting to MSIX package, make sure Certificate File (.pfx) or Certificate Store details are given as shown in Setting MSIX Signing Options.*
4. On the **Tools** menu, select **Application Conversion Wizard**. The **Application Conversion Wizard Welcome** panel opens.
5. Click **Next**. The **Select Output Formats** panel opens.

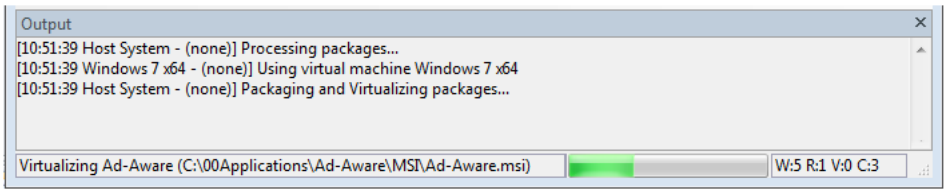


6. Select one or more of the following output formats:
   - Windows Installer Packages (*.msi)
   - Microsoft App-V Packages (*.sft)
   - Citrix XenApp Profiles (*.profile)
   - VMware ThinApp Packages (*.exe)
   - MSIX Packages (*.msix)
7. In the **Place packages under the following folder** field, specify the directory where you want to save the output packages.
8. Click **Next**. The **Automated Repackaging on Virtual Machines** panel opens.



9. From the **Virtual Machine Platform** list, select one of the following:
   - **Any Platform**—The Automated Application Converter will use any of the virtual machines that you have selected on the **Machines** tab to perform automated repackaging, regardless of platform.
   - **OS Platform**—If you select a specific operating system, the Automated Application Converter will use only those virtual machines that you have selected on the **Machines** tab that are of the selected operating system to perform automated repackaging.

   Important • *When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in the **Platform** field on the **Machines** tab and making a selection from the list.*
10. Click **Next**. The **Application Conversion Wizard Complete** panel opens.
11. Click **Finish** to close the wizard and begin converting the selected packages using the selected virtual machines. As conversion proceeds, there are several progress indicators at the bottom of the screen:



The following information is listed:

- **Messages**—Messages are listed in the **Output** window
- **Current file**—The name of the current file being processed is listed in the lower left.
- **Progress bar**—A progress bar is displayed at the bottom of the screen.
- **Count**—The count of packages in each of the following categories is displayed at the lower right:
  - **W:5**—Number of packages that are **W**aiting to be processed.
  - **R:1**—Number of packages that are being **R**epackaged.
  - **V:0**—Number of packages that are being converted or **V**irtualized into MSI and/or virtual packages.
  - **C:3**—Number of applications that have finished processing, including **C**ompleted and failed applications.

12. Proceed with Viewing Conversion Results.

Note • *If you have selected multiple virtual machines, the Automated Application Converter will attempt to connect to the first virtual machine in the list. If it successfully connects, conversion will proceed on that machine. If it fails to connect, it will move on to the next machine in the list.*

Viewing Conversion Results

To view the conversion results on the **Results** tab and in the AdminStudio Automated Application Converter Log report, perform the following steps:

Task ***To view conversion results:***

1. Perform conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard.
2. Open the **Results** tab.
3. For each listed package, view the information in the **Errors**, **Warnings**, and Results Icons ( ) columns, as described in Results Tab.
4. Select the top level node of a conversion run log (such as `Log started Monday, June 21, 2010...`).
5. Do one of the following:
   - Click the **Results** button on the toolbar.
   - Select **View Report** from the shortcut menu.

- Select **View Report** on the **Tools** menu.
- Press Ctrl+R.

The AdminStudio Automated Application Converter Log report opens. See AdminStudio Automated Application Converter Log Report for more information.

6. Open the **Packages** tab.
7. In the tree, locate one of the source packages that you converted and click the plus sign to expand the listing. The converted packages in the formats you selected are listed.
8. Continue with the steps in Testing Packages and Importing Converted Packages into the Application Catalog.

Testing Packages

After you use Automated Application Converter to create a package, you can test it prior to deployment. You can test any of the following package types:

- **Virtual package**—A virtual package that was converted from a Windows Installer package using the Automated Application Converter.
- **Repackaged MSI package**—A repackaged Windows Installer package that was converted from a source Windows Installer package using the Automated Application Converter.
- **Source package**—A source Windows Installer package that you have added to the **Packages** tab.

Information about testing packages is organized into the following topics:

- Testing App-V Packages
- Testing VMware ThinApp Packages
- Testing Citrix XenApp Packages
- Testing Repackaged and Source Windows Installer Packages
- Testing Repackaged and Source Windows Installer Packages

Testing App-V Packages

By default. when you build an App-V package, the App-V Application Launcher utility (`AppVLauncher.exe`) is placed in the same folder as the App-V package. The App-V Application Launcher is a convenient testing tool that makes it possible for you to reliably and accurately test your App-V packages before deployment on an App-V Server.

Figure 10-14: App-V Application Launcher Utility

If you do not want to include the launcher with an individual App-V package, set the package's **Launcher Tool** property to **Disabled**. You can set the default value for this property on the **Project Options** dialog box.

For information on performing both automated and manual testing of App-V packages, see the following topics:

- Performing Automated Testing of App-V Packages
- Performing Manual Testing of App-V Packages

Performing Automated Testing of App-V Packages

You can choose to launch an App-V package for testing on a virtual machine directly from Automated Application Converter.

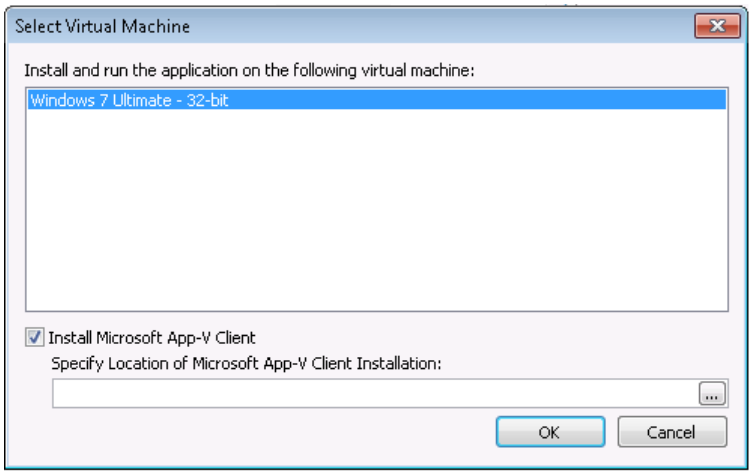
- Performing Automated Testing of an App-V 4.x Package
- Performing Automated Testing of an App-V 5.x Package

Performing Automated Testing of an App-V 4.x Package

You can choose to launch an App-V 4.x package for testing on a virtual machine directly from Automated Application Converter.

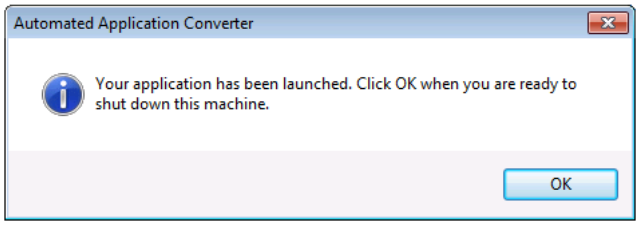
Task ***To launch an App-V package for testing on a virtual machine:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Launch the virtual machine that you want to use for testing.
3. On the **Packages** tab, right-click on an App-V 4.x package and select **Launch Package for Testing** from the shortcut menu. The **Select Virtual Machine** dialog box opens, prompting you to select the virtual machine that you want to use to test the selected package.



4. Select a virtual machine from the list.
5. If the Microsoft App-V 4.x client is not yet installed on the selected virtual machine, select the **Install Microsoft App-V Client** option and browse to the location of the Microsoft App-V 4.x Client installation file. Make sure that it is in a location that is accessible to the virtual machine.

6. Click **OK**. The following will occur:
   - The Automated Application Converter will connect to the selected virtual machine.
   - The virtual machine will reboot to the snapshot listed on the **Machines** tab in the **Snapshot Name** property of the selected virtual machine.
   - The Guest Agent will launch, and progress messages will be displayed.
   - The App-V 4.x package and launcher will be copied to the virtual machine.
   - If the **Install Microsoft App-V Client** option was selected on the **Select Virtual Machine** dialog box, the App-V client will be installed.
   - The App-V package will launch, and the following message will be displayed:



7. Test the application to determine whether it is operating properly.
8. When you have completed testing, click the **OK** button on the Automated Application Converter message dialog box to shut down the virtual machine. After you click OK, the following message will be displayed in the Automated Application Converter **Output** window:

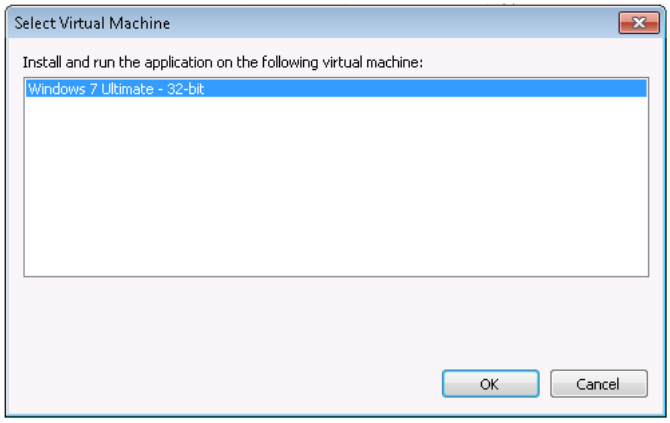
```
[10:54:58 Windows7 - MyApp.sft] Done running application. Shutting down machine..
```

Performing Automated Testing of an App-V 5.x Package

You can choose to launch an App-V 5.x package for testing on a virtual machine directly from Automated Application Converter.

Task ***To launch an App-V 5.x package for testing on a virtual machine:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Launch the virtual machine that you want to use for testing.
3. On the **Packages** tab, right-click on an App-V package and select **Launch Package for Testing** from the shortcut menu. The **Select Virtual Machine** dialog box opens, prompting you to select the virtual machine that you want to use to test the selected package.

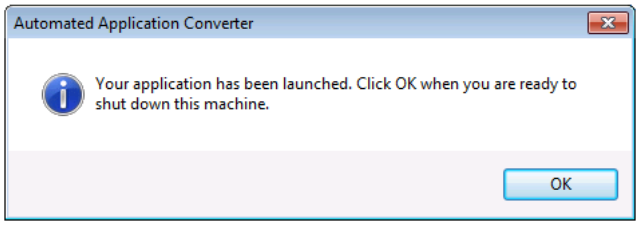


4. Select a virtual machine from the list.

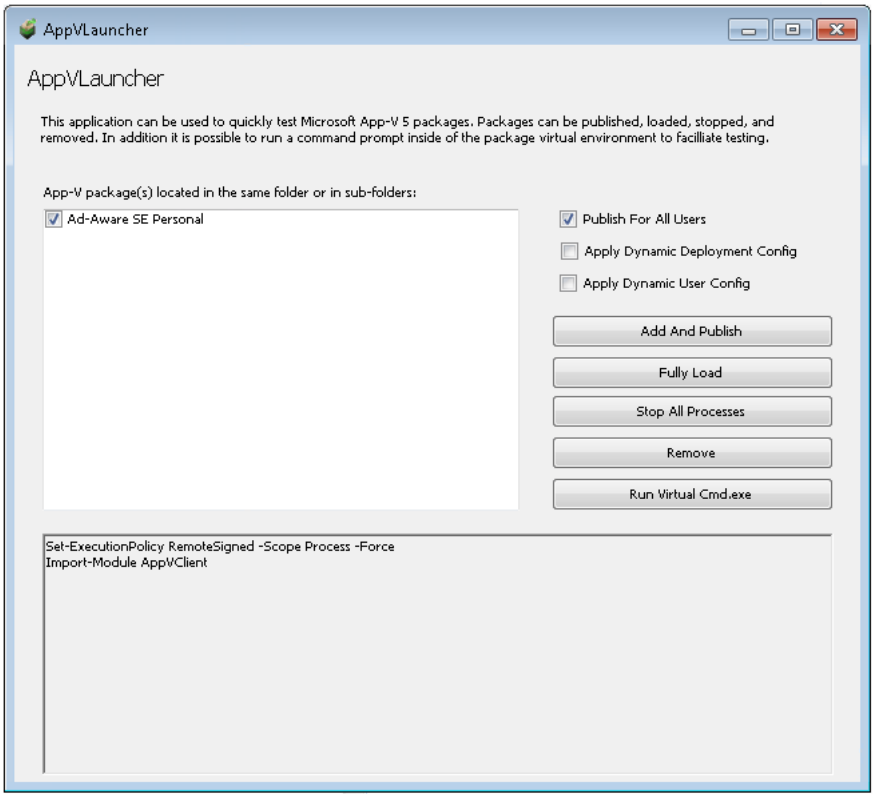
Important • *If you are testing an App-V 5.x package, the Microsoft App-V 5.x client must already be installed on the virtual machine.*

5. Click **OK**. The following will occur:

   - The Automated Application Converter will connect to the selected virtual machine.
   - The virtual machine will reboot to the snapshot listed on the **Machines** tab in the **App-V 5.x Client Snapshot** property of the selected virtual machine.
   - The Guest Agent will launch, and progress messages will be displayed.
   - The App-V package and launcher will be copied to the virtual machine.
   - The following message will be displayed:



   - The App-V 5.x Application Launcher will open, and will list the App-V packages located in the folder that was copied to the virtual machine:



6. In the App-V package list, select the App-V packages that you want to test.
7. Click the **Add and Publish** button to publish this package to the App-V client so that it can be tested. All of the entry points into the package will be published, including shortcuts and file type extensions, among others.
8. On the **Start** menu, click the shortcut for this application to launch it.
9. Test the application to determine whether it is operating properly.
10. When you are finished testing the application, click the **Remove** button to un-publish the application.

    In some instances, you may receive an error message stating that a process is still running. If you get an error, click the **Stop All Processes** button to kill all processes.
11. When you have completed testing, click the **OK** button on the Automated Application Converter message dialog box to shut down the virtual machine. After you click **OK**, the following message will be displayed in the Automated Application Converter **Output** window:

    ```
    [10:54:58 Windows7 - MyApp.appv] Done running application. Shutting down machine..
    ```

Performing Manual Testing of App-V Packages

You can manually test an App-V package by copying the package and launcher to a machine where the appropriate App-V client is installed and then opening the App-V Application Launcher.

- Performing Manual Testing of an App-V 4.x Package
- Performing Manual Testing of an App-V 5.x Package

Performing Manual Testing of an App-V 4.x Package

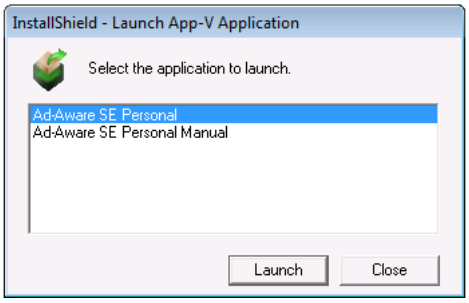
You can manually test an App-V 4.x package by copying the package and launcher to a machine where the appropriate App-V 4.x client is installed and then opening the App-V 4.x Application Launcher.

Task ***To manually test an App-V 4.x package:***

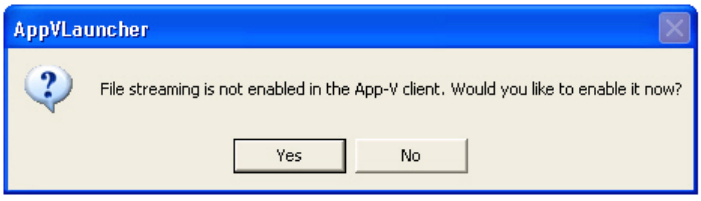
1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Locate the folder containing the App-V 4.x package and copy the entire folder to a virtual machine where the appropriate App-V 4.x client is installed.

   Tip • *A quick way to jump to the folder containing an App-V package is to right-click on the package on the Automated Application Converter* ***Packages*** *tab and then select* ***Explore*** *from the shortcut menu.*
3. Double-click on the App-V 4.x Application Launcher utility (`AppVLauncher.exe`). The App-V 4.x Application Launcher will attempt to launch that application.

   - **If an App-V 4.x package has only one target defined** (that is, if the App-V 4.x package has only one `.osd` file), the App-V Application Launcher starts the App-V package.
   - **If the App-V 4.x package has more than one target defined** (that is, if the App-V 4.x package has two or more `.osd` files), the App-V Application Launcher displays a dialog box that lists each target, and it lets you select the one that you want to launch.

     Select the application you want to test and click **Launch**.
4. If App-V file streaming in the App-V Client is not enabled, you will be prompted to enable it:

5. Click **Yes**. The App-V package will open

***Note** • The first time that you use the App-V Application Launcher to run an application in an App-V package, the entire package is published to that machine; this includes all of the package's shortcuts and file extension associations in the package. If you then use the App-V Application Launcher to run any application in the App-V package again, the App-V Application Launcher unpublishes the package (and its shortcuts and file extension associations) before republishing the package.*

***Note** • Also note that the AppVLauncher.exe file requires elevation. If you want to be able to test your App-V package in a locked-down environment where end users will not have elevated privileges, you may want to use the App-V Application Launcher once to launch and publish your App-V package with elevated privileges. Once you have done that, you can use the published shortcuts and file extension associations to start your application.*

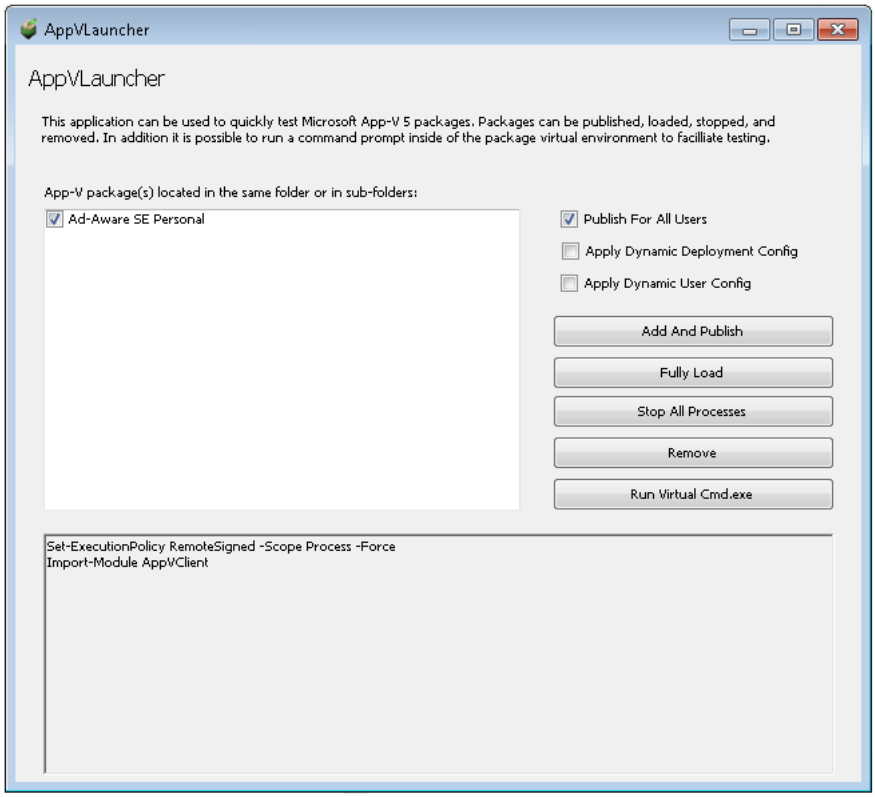
Performing Manual Testing of an App-V 5.x Package

You can manually test an App-V 5.x package by copying the package and launcher to a machine where the App-V 5.x Client is installed and then opening the App-V Application Launcher.

Task ***To manually test an App-V 5.x package:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Locate the folder containing the App-V 5.x package and copy the entire folder to a virtual machine where the App-V 5.x Client is installed.

   ***Tip** • A quick way to jump to the folder containing an App-V package is to right-click on the package on the Automated Application Converter **Packages** tab and then select **Explore** from the shortcut menu.*
3. Double-click on the App-V Application Launcher utility (AppVLauncher.exe). The **App-V 5.x Launcher** dialog box opens and lists the available packages in the same folder or subfolders:



4. Select the package that you want to launch and then click **Add and Publish**. All of the integration points with the system such as shortcuts and file type associations will be created. After the package is published to the App-V 5.x Client, it is ready to launch.
5. Optionally, click **Fully Load** to locally cache all of the App-V package's files (which is similar to setting a package's **Package Optimization** setting to **Offline**).
6. Click the shortcut to open the package.
7. When you are finished testing, either exit the application or click **Stop All Processes.**
8. To remove or "uninstall" the package, click **Remove**.
9. If you want to run a command line within the virtual environment, click **Run Virtual Cmd.exe**.

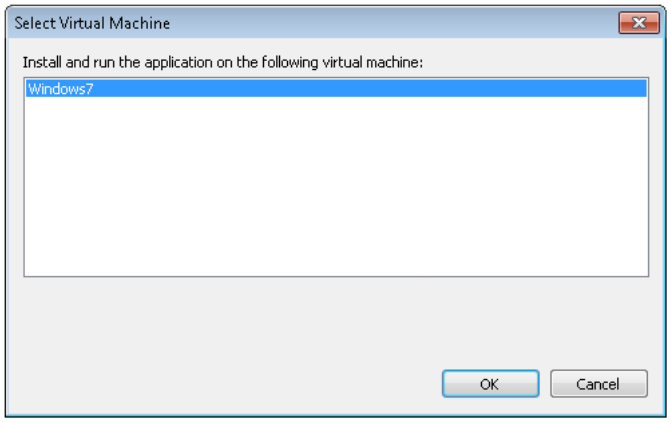
Testing VMware ThinApp Packages

You can choose to launch a VMware ThinApp package for testing on a virtual machine directly from Automated Application Converter.

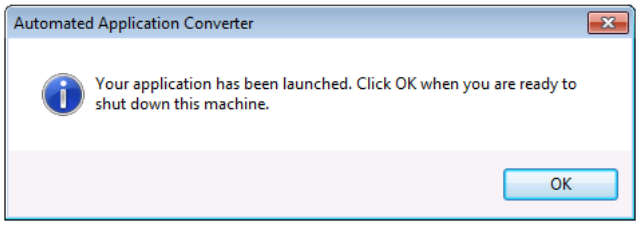
Task ***To launch a VMware ThinApp package for testing:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Launch the virtual machine that you want to use for testing.

3. On the **Packages** tab, right-click on a VMware ThinApp package and select **Launch Package for Testing** from the shortcut menu. The **Select Virtual Machine** dialog box opens, prompting you to select the virtual machine that you want to use to test the selected package.



4. Select a virtual machine from the list that has VMware ThinApp installed on it.
5. Click **OK**. The following will occur:
   - The Automated Application Converter will connect to the selected virtual machine.
   - The virtual machine will reboot to the snapshot listed on the **Machines** tab in the **Snapshot Name** property of the selected virtual machine.
   - The Guest Agent will launch, and progress messages will be displayed.
   - The VMware ThinApp package will be copied to the virtual machine.
   - The VMware ThinApp package will launch, and the following message will be displayed:



6. Test the application to determine whether it is operating properly.
7. When you have completed testing, click the **OK** button on the Automated Application Converter message dialog box to shut down the virtual machine. After you click OK, the following message will be displayed in the Automated Application Converter Output window:

```
[10:48:25 Windows7 - MyApp.exe] Done running application. Shutting down machine.
```

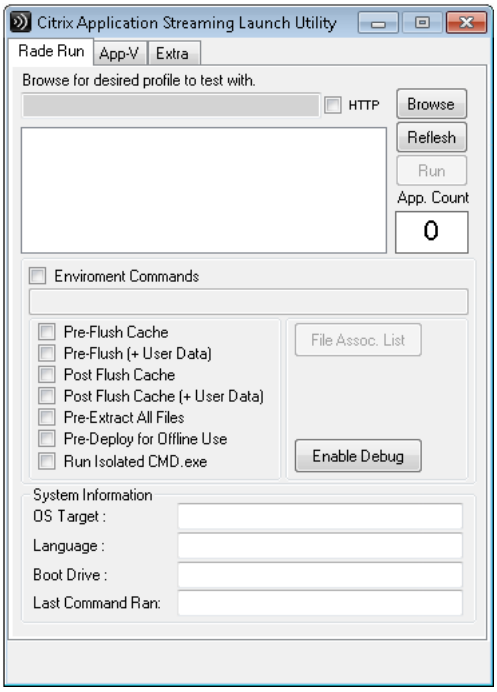
Testing Citrix XenApp Packages

Before you deploy a Citrix XenApp package on a XenApp server, you can test it using the **Citrix Application Streaming Launch Utility**. To test Citrix XenApp packages prior to deployment, perform the following steps:

Task ***To test Citrix XenApp packages:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. Copy the entire folder containing the Citrix package to a virtual machine where the **Citrix Offline Plugin** and the **Citrix Application Streaming Launch Utility** have been installed.

   Tip • *For installation instructions for these two utilities, see Application Streaming Launch Tool in the Citrix Knowledge Center.*
3. Launch the **Citrix Application Streaming Launch Utility**.



4. Click **Browse** and select the `.profile` file that you want to test. The application is listed in the box.
5. Select the application in the list and click **Run**. The application launches.
6. Perform testing.

Testing Repackaged and Source Windows Installer Packages

You can choose to launch a Windows Installer package for testing on a virtual machine directly from Automated Application Converter. You can test both of the following types of Windows Installer packages:

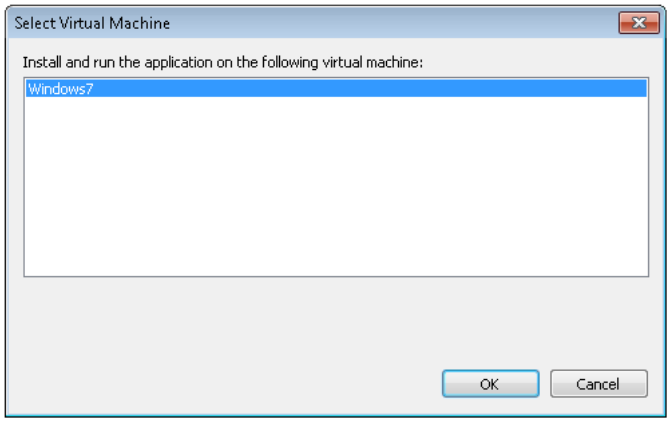
- **Repackaged MSI package**—A repackaged Windows Installer package that was converted from a source Windows Installer package using the Automated Application Converter.

- **Source package**—A source Windows Installer package that you have added to the **Packages** tab.

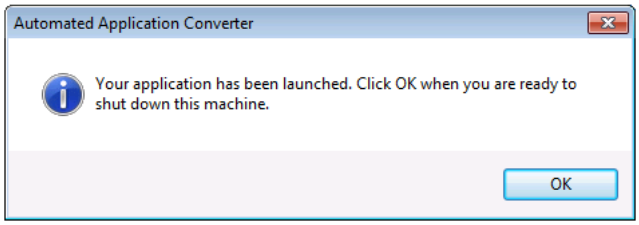
To test a Windows Installer package, perform the following steps:

Task ***To launch a Windows Installer package for testing:***

1. Do one of the following:
   - Perform package conversion as described in Using the Application Conversion Project Wizard to Perform an End-to-End Conversion or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
   - Add a Windows Installer package to the Packages tab, as described in Adding Packages from an AdminStudio Application Catalog or Adding Packages from a Local Machine or Network.
2. Launch the virtual machine that you want to use for testing.
3. On the **Packages** tab, right-click on a Windows Installer package and select **Launch Package for Testing** from the shortcut menu. The **Select Virtual Machine** dialog box opens, prompting you to select the virtual machine that you want to use to test the selected package.



4. Select a virtual machine from the list and click **OK**. The following will occur:
   - The Automated Application Converter will connect to the selected virtual machine.
   - The virtual machine will reboot to the snapshot listed on the **Machines** tab in the **Snapshot Name** property of the selected virtual machine.
   - The Guest Agent will launch, and progress messages will be displayed.
   - The Windows Installer package will be copied to the virtual machine.
   - The Windows Installer package will launch, and the following message will be displayed:



5. Test the application to determine whether it is operating properly.
6. When you have completed testing, click the **OK** button on the Automated Application Converter message dialog box to shut down the virtual machine. After you click OK, the following message will be displayed in the Automated Application Converter Output window:

   ```
   [10:48:25 Windows7 - MyApp.msi] Done running application. Shutting down machine.
   ```

Importing Converted Packages into the Application Catalog

After you convert packages to virtual packages or repackaged Windows Installer packages, your next step is to import those packages into the Application Catalog so that you can perform testing and then distribute them to your desired distribution system.

You can import these packages one at a time or you can import all of the packages in a specified directory. For detailed instructions, see the following help topics:

- Importing a Single Package File
- Importing a Folder of Multiple Applications

Publishing Converted Packages to a Distribution System

You can use the Distribution Wizard to publish an application or group of applications from the Application Catalog to a distribution system. The following distribution systems and package types are supported:

Table 10-14 • Supported Package Types Per Distribution System

Distribution System	Supported Package Types
ConfigMgr **(Formerly called as System Center Configuration Manager)**	• Windows Installer packages • App-V (4.x and 5.0) packages • MSIX packages
Citrix XenApp Server	• Citrix XenApp profiles • App-V 4.x packages
Symantec Altiris Management Server	• Windows Installer packages • VMware ThinApp packages

Important • *When publishing applications to one of these distribution systems, the selected applications' supported packages will be published. However, if an application contains packages of other deployment types, those packages will be ignored.*

For more information, see Distributing Applications Using the Distribution Wizard.

Note • *You can also distribute an individual Windows Installer or App-V 4.x package to System Center 2007 Configuration Manager using the Package Distribution Wizard, as described in Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager).*

Setting Default Project Properties

You can specify project-wide default values for individual package properties on the **Project Options** dialog box, which is opened by selecting **Options** on the **Tools** menu.

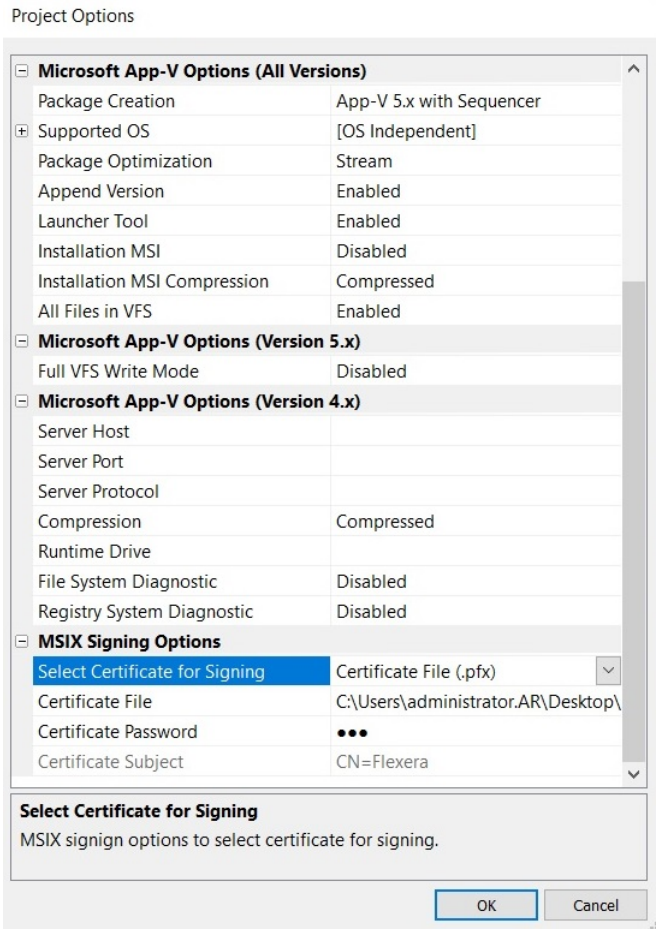


Figure 10-15: Project Options Dialog Box

When packages are added to the **Packages** tab, they will inherit the property values defined on the **Project Options** dialog box. However, properties for individual packages can be overridden in the package's **Properties** window on the **Packages** tab, as described in Editing Package Properties on the Packages Tab.

The default properties that are set on the **Project Options** dialog box include:

- **Output path**—You can specify the default output location for converted packages.
- **Default output formats**—You can specify the default package conversion types (App-V, Citrix XenApp, VMware ThinApp, Windows Installer) for the conversion wizards:
  - **Application Conversion Project Wizard**—Packages will be converted to each of the specified formats.
  - **Application Conversion Wizard**—The specified formats will be selected, by default, on the **Select Output Formats** panel.
- **App-V version and method**—You can specify the default App-V conversion method: App-V 4.6 with AdminStudio, App-V 5.x with AdminStudio, or App-V 5.x with Sequencer.
- **MSIX Signing Options**—You can specify the signing options: Certificate File (.pfx), or Certificate Store.

To specify project-wide default values for individual package properties, perform the following steps:

Task ***To set default project properties:***

1. On the **Tools** menu, select **Options**. The **Project Options** dialog box opens.

   The **Project Options** dialog box includes properties that are grouped into the following sections:

   - Conversion Options
   - Microsoft App-V Options (All Versions)
   - Microsoft App-V Options (Version 4.x)
   - Microsoft App-V Options (Version 5.x)
   - MSIX Signing Options
2. Set project options as described in Project Options Dialog Box.
3. Click **OK** to save your selections.

Tip • *You can also specify global default settings for any App-V virtual setting in the* `ISVirtualPackage` *table by editing the* `settings.xml` *file. For more information, see Specifying Global Default Virtual Conversion Settings.*

Capturing Virtualization Context

Sometimes it is necessary to repackage a Windows Installer package before you can successfully virtualize it (as described in Virtualization Conversion Error Messages).

When some Windows Installer packages are repackaged, some of their data (such as files or registry entries) are excluded according to the normal Repackager exclusion settings. For example, files destined for the `\Windows\Installer` folder are typically excluded. However, this type of information is occasionally necessary for a small set of applications which use Windows Installer APIs to determine whether they have been successfully or completely installed.

In order to get these settings into the generated virtual packages, when Repackager builds a Windows Installer package, it produces two `.msi` files: `packagename.msi` and `packagename.context.msi`, with `packagename.context.msi` containing the additional settings.

Figure 10-16: Repackaged Output: application.msi and application.context.msi

Repackager then converts the combined content of the main `.msi` and the `context.msi` into the desired virtual package.

Important • *If you are not converting a package to a virtual package, you can ignore its `.context.msi` file.*

Note • *Context data is not displayed in the Repackager interface when viewing captured Files/Registry details.*

How is the Context Data Configured

Context data is configured in a settings file in the `AdminStudio Shared` folder called `isrepackager.context.ini`. It is identical in syntax to the familiar `isrepackager.ini` file which is used for exclusion settings. Data that matches the settings in the `context.ini` file is captured—not into the main application `.msi` file, but rather into a separate `context.msi` file. When creating a virtual package, Repackager combines the data in both the main `.msi` file and the `.context.msi` file to produce the final virtual package.

Reference

This section describes each of the user interface elements and Wizard panels that you might encounter when using the Automated Application Converter. The help topics in this Reference section are the same detailed documentation that is displayed when you press the F1 key or click the **Help** button while working in a dialog box.

Reference information is organized into the following sections:

Table 10-15 • Organization of Automated Application Converter Reference Section

Section	Description
Automated Application Converter User Interface	Contains information about the main Automated Application Converter interface, including tabs, menus, and the toolbar.
Wizards	Contains a panel-by-panel reference for each Wizard in the Automated Application Converter.
Dialog Boxes	Provides specific help for each dialog box in the Automated Application Converter.

Table 10-15 • Organization of Automated Application Converter Reference Section

Section	Description
Command Line Support	Explains how to run the Automated Application Converter project file via command line.
Specifying Global Default Virtual Conversion Settings	Explains how to set global default settings for any App-V virtual setting in the `ISVirtualPackage` table by editing the `settings.xml` file.

Automated Application Converter User Interface

Information on the Automated Application Converter user interface is presented in the following sections:

- Packages Tab
- Machines Tab
- Results Tab
- Menus & Toolbar Buttons
- Output Window
- Column Selector and Properties Windows
- AdminStudio Automated Application Converter Log Report
- Using List Features

Packages Tab

On the **Packages** tab, you select the packages that you want to virtualize/repackage. On this tab, you can also set package properties and view package status.

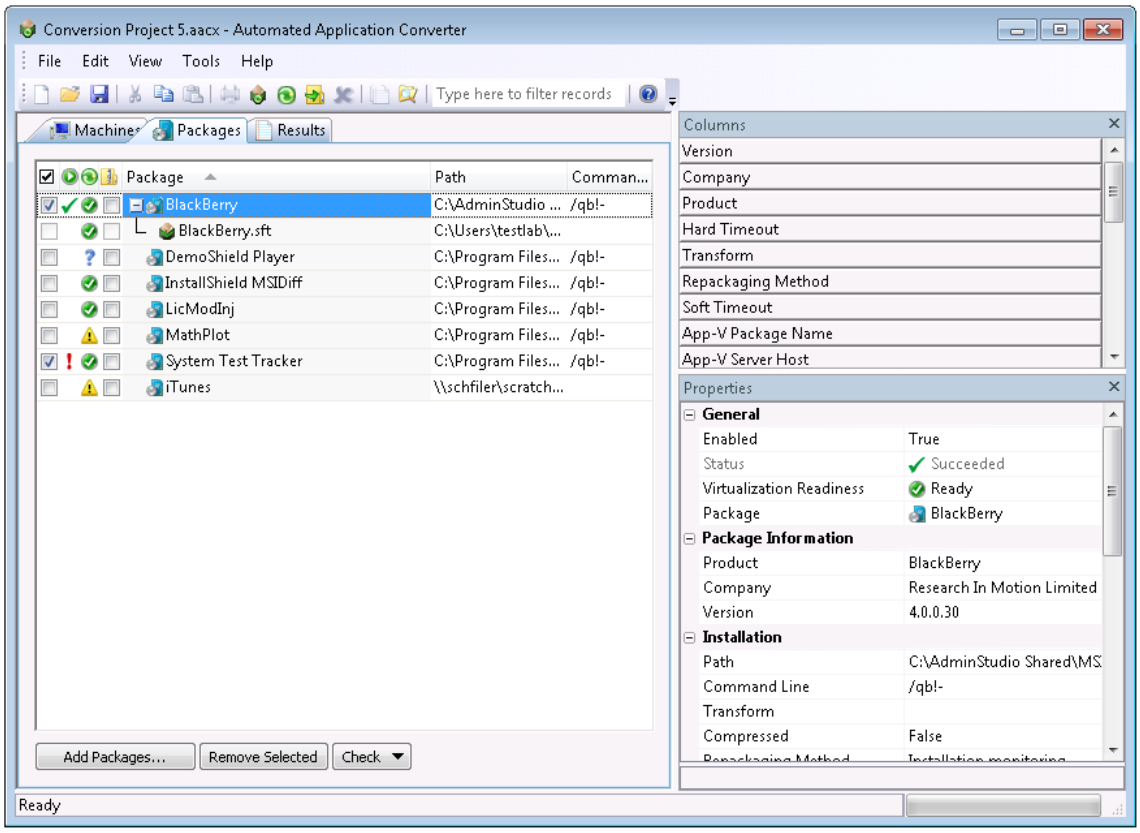


Figure 10-17: Packages Tab

This section includes the following information about the **Packages** tab:

- Adding Packages to the List
- Viewing Package Information on the Packages Tab
- Packages Tab Properties
- Icons Used on the Packages Tab
- Shortcut Menu Commands on Packages Tab

Adding Packages to the List

You add packages to this list using the Package Import Wizard or the Application Conversion Project Wizard. For instructions, see the following topics:

- Adding Packages from an AdminStudio Application Catalog
- Adding Packages from a Local Machine or Network

Viewing Package Information on the Packages Tab

By default, the **Packages** tab lists the **Status**, **Virtualization Readiness**, **Path**, and **Command Line** columns for each selected package. Additional columns of information can be viewed by selecting one of the fields in the Column Selector area and dragging it onto the list. Also, the values for these fields for the selected package can be viewed in the **Properties** window.

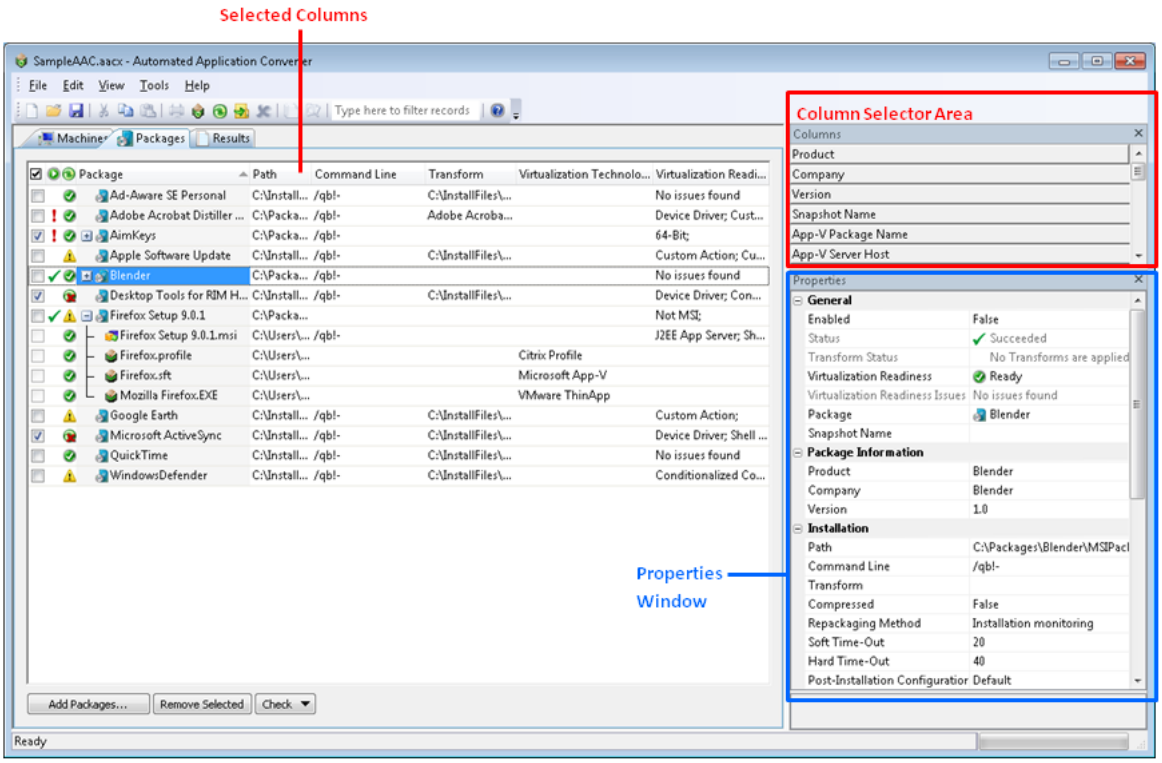


Figure 10-18: Column Selector Area on the Packages Tab

Note • *You can sort these lists, change the columns that are displayed, change the column order, resize the columns, and group these lists by a specific column. See Using List Features for more information.*

Packages Tab Properties

The **Packages** tab includes the following categories of properties, most of which can be edited in the package list or in the Properties Window:

- General
- Package Information
- Installation
- Microsoft App-V Options (All Versions)
- Microsoft App-V Options (Version 5.x)
- Microsoft App-V Options (Version 4.x)

- MSIX Signing Options
- Packages Tab Buttons

General

The **General** category includes the following properties:

Table 10-16 • Packages Tab / General Category of Properties Window

Property	Description
Enabled	To select a package for conversion, click the check box in this column of the list. ***Note*** • *This column corresponds with the **Enabled** field under **General** in the Properties window, which can be set to either **True** or **False**.*
Status	Displays an icon to indicate the status of the package when it is being repackaged or virtualized, that the process has completed, or that the Automated Application Converter encountered an error during the process. See Icons Used on the Packages Tab.
Transform Status	Indicates whether any transforms are associated with the listed Windows Installer package. Automated Application Converter automatically adds all of the `.mst` files located in the same directory as the selected `.msi` file. If a transform is associated with the selected package, one of the following two icons are displayed in this column: • One transform is being added with this package. • Multiple transforms are being added with this package. If multiple transforms are associated with this package, you should click the browse button in the **Transform** column (or the browse button in the **Transform** field in the **Properties** window) to open the **MST** dialog box and specify which transform files you want to add and the order that you want the transforms applied. For more information, see MST Dialog Box.

Table 10-16 • Packages Tab / General Category of Properties Window

Property	Description
Virtualization Readiness	When you add a package to the **Packages** tab, the Automated Application Converter does a quick check to identify that package's virtualization readiness status and assigns it one of the following icons: • **Ready** —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • **Requires repackaging** —Package must be repackaged before it can be successfully virtualized. • **Virtualization not supported** —Automated Application Converter has determined that virtualization is not supported. • **Virtualization not recommended** —Automated Application Converter has determined that this package is not recommended for virtualization. • **Unknown** —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging. ***Note*** • *You can click on the icon in this column to override the Virtualization Readiness status that was automatically assigned to this package by the Automated Application Converter.* *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize that package, you must first override the status and change it to **Ready** or **Requires repackaging**.*
Virtualization Readiness Issues	This read-only setting lists the issues that were found for this package. This is used to make the virtualization readiness recommendation.
Package	Lists the name of the package, as determined by the value in the **Product** property under **Package Information**. This is used as part of the output path for repackaged or virtualized results. It is also used in reports to refer to this package. If the package has been repackaged or converted to a virtual package, those output files are also listed below the source file in a tree structure, with an icon identifying the file type. See Icons Used on the Packages Tab.
Snapshot Name	Specify an override snapshot name to use instead of the snapshot that the machine is configured to use.

Package Information

The **Package Information** category includes the following properties:

Table 10-17 • Packages Tab / Package Information Category of Properties Window

Property	Description
Product	Name of the package as provided by the company who manufactured it.
Company	Name of the company who manufactured this package.
Version	Version of the package.
Virtualization Technology	Indicates the virtualization format of the virtual package. Options are: • Microsoft App-V • Citrix XenApp • VMware ThinApp • Windows Installer Package ***Note*** • *This property is only displayed when a virtual package or repackaged Windows Installer package is selected under its original package.*

Installation

The **Installation** category includes the following properties:

Table 10-18 • Packages Tab / Installation Category of Properties Window

Property	Description
Path	When a parent package is selected, this property lists the location from where the package was selected locally or from where it was originally imported into the AdminStudio Application Catalog. If a child virtual package is selected, this property lists the location of the virtual package. ***Note*** • *It is recommended that you use a UNC path when importing packages into the Application Catalog.* *If you are adding packages from an AdminStudio Application Catalog installed on a machine other than the machine where the Automated Application Converter is installed, make sure that the package source path listed here is accessible to the Automated Application Converter machine. If it is also accessible to the virtual machines, repackaging can be performed more quickly.*
Command Line	Editable field that lists the command line parameters that will be used to run this installation silently during repackaging.

Table 10-18 • Packages Tab / Installation Category of Properties Window

Property	Description
Transform	This field can contain a semicolon-delimited list of transforms used to modify or install a Windows Installer package silently. To add a transform to this list, click the Browse button to open the **MST** dialog box. For more information, see MST Dialog Box. ***Note*** *• If the transform file is located in the same directory as the* `.msi` *file, only the* `.mst` *file name is listed in the* ***Transform*** *field. If you have added a transform file from another directory, the full path is listed in this field.*
Compressed	Indicates the compressed status of the package: • **False**—Indicates that the source `.msi` or `.exe` file is uncompressed. If this package is repackaged, the Automated Application Converter will copy all of the files in the same folder as the installation file to the virtual machine. • **True**—Indicates that the source `.msi` or `.exe` file is compressed. If this package is repackaged, the Automated Application Converter will copy only this single installation file to the virtual machine.
Repackaging Method	Indicates the repackaging method that will be used to repackage this package: • **Installation monitoring**—Repackager monitors system changes as a package is installed, and that data is converted into a Windows Installer package. • **Single-step snapshot**—Repackager first takes an initial system snapshot, then runs the installation, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package.
Soft Time-Out	Number of minutes allotted for the package to install before the user would be notified. After this time period elapses, the user will be notified, just in case there are pending dialogs for the user to dismiss or if some other user interaction is required. The default value is 20.
Hard Time-Out	Number of minutes allotted for the package to install before it is considered a failure. If this time period elapses, the Automated Application Converter would consider the installation a failure and would move to the next package. The default value is 40.
Pre-Installation Configuration	Indicate whether to allow manual configuration of the machine before beginning the capture process. This can be useful when a particular dependency, such as Java runtime, needs to be installed and should not be captured as part of the application capture process. Available options are: • **Disabled**—Disable pre-installation configuration. • **Enabled**—Enable pre-installation configuration.

Table 10-18 • Packages Tab / Installation Category of Properties Window

Property	Description
Post-Installation Configuration	Indicate whether you want to enable configuration of the application after it is installed on the virtual machine but before it is converted into the target formats. Available options are: • **Default**—Use the project-level behavior that is specified through the **Post-Installation Configuration** field on the **Project Options** dialog box. This is the default value. • **Disabled**—Disable post-installation configuration. The repackaging process does not pause after installing the product. • **Enabled**—Enable post-installation configuration. The repackaging process pauses after the installation of the product to allow you to launch the product and set up various application settings such as update settings and file associations. You can also perform other system configuration tasks. Once you are done with configuration, you can click a button to have the repackaging proceed with the capture and convert process. ***Important*** • *If you select the **Enabled** option, ensure that the value that you enter for the **Hard Time-Out** setting allows enough time to configure the application.* ***Note*** • *This option is ignored when performing conversion to App-V 5.0 using the Microsoft App-V 5.0 Sequencer.*
Manual Install	Indicate whether you want to manually perform the application installation tasks. Manual installation can be used for more complex installations. Available options are: • **Disabled**—Disable manual installation. • **Enabled**—Enable manual installation. ***Important*** • *This option is ignored during App-V 5.0 conversion using the Microsoft Sequencer.*
Documentation Tool	Indicate whether you want to enable or disable the documentation tool during repackaging. Available options are: • **Disabled**—Disable the documentation tool. This is the default value. • **Enabled**—Enable the documentation tool. • **Default**—Use the project-level behavior that is specified through the **Documentation Tools** field on the **Project Options** dialog box. This is the default value. ***Note*** • *For more information on the documentation tool, see Documenting Repackaging Steps Using the Microsoft Step Recorder Tool.*

Microsoft App-V Options (All Versions)

The **Microsoft App-V Options (All Versions)** category includes the following properties:

Table 10-19 • Packages Tab / Microsoft App-V Options (All Versions) Category of Properties Window

Property	Description
Name	Enter a name (a maximum of 64 characters) to override the name of the App-V package. By default, this matches the value of the **Product** property under **Package Information**. ***Tip*** *• If your virtual package contains multiple applications, you can specify the name that identifies the entire package. For example,* `Microsoft Office` *could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
Comments	Enter a short description of the App-V package. ***Note*** *• This setting is optional.*
Package Creation	Select one of the following options to identify an App-V conversion method: • **Default**—Use the method that is selected in the **Package Creation** field on the **Project Options** dialog box. This is the default value. • **App-V 4.6 with AdminStudio**—When converting this package to App-V format, use AdminStudio to convert it to an App-V 4.6 package. • **App-V 5.x with AdminStudio**—When converting this package to App-V format, use AdminStudio to convert it to an App-V 5.x package. • **App-V 5.x with Sequencer**—When converting this package to App-V format, use Microsoft Sequencer to convert it to an App-V 5.x package.

Table 10-19 • Packages Tab / Microsoft App-V Options (All Versions) Category of Properties Window

Property	Description
Primary Application Directory	Enter one of the following: • **App-V 5.x conversions using the Sequencer**—Specify the absolute folder path to the expected main installation location of the package to be converted. If no value is specified, then the App-V 5.0 package will be created with all files in the virtual file system (VFS) folder. • **App-V 4.x or 5.x conversions using AdminStudio**—Specify the main installation directory which will be used to set up the root/mount folder mapping. For example, for Yahoo Messenger, AdminStudio automatically detects `C:\Program Files\Yahoo!` as the primary installation directory. However, you may prefer that the primary installation directory be `C:\Program Files\Yahoo!\Messenger`, because this directory is more correct for Messenger. In this case, you can enter this new path in the **Primary Application Directory** property field, and it will be honored by the AdminStudio converter as long as this path exists in the Windows Installer Package. If it does not exist, then AdminStudio will fall back to use the directory it found during automatic detection. ***Important*** • *When you use the **App-V 5.x with Sequencer** option, it is highly recommended that you enter a value for the **Primary Application Directory** property. If you do not, then all of the converted files will end up in the virtual file system (VFS) folder.* ***Note*** • *When using the **App-V 4.x/5.x with AdminStudio** package creation options, this field is optional.*

Table 10-19 • Packages Tab / Microsoft App-V Options (All Versions) Category of Properties Window

Property	Description
Supported OS	Use to specify the operating systems that the App-V package will support: • **To accept default values**—To accept the default values for **Supported OS** that are set on the **Project Options** dialog box, set **Default** to **True**. When you make this selection, all operating systems and **OS Independent** will automatically switch to **False**, and the word `[Default]` will be listed next to **Supported OS**. • **If the App-V package is operating-system-dependent** (meaning that it only supports some of the listed operating systems), select **True** next to the supported operating systems. If any of the listed operating systems are set to **True**, the value for **Default** and for **Supported OS** will automatically switch to **False**, and the selected operating systems will be listed in brackets next to **Supported OS**. • **If the App-V package is operating-system-independent** (meaning that it supports all listed operating systems), set **OS Independent** to **True**. When you make this selection, all operating systems and **Default** will automatically switch to **False**, and `[OS Independent]` will be listed next to **Supported OS**. ***Important*** • *When setting the **Supported OS** property for App-V 5.0 packages, keep in mind that the packages are limited to the supported operating systems of the App-V 5.0 client:* • *Windows 7 and later* • *Windows Server 2008 R2 and later*
Package Optimization	Specify how to optimize the package: • **Default**—Use the method that is selected in the **Package Optimization** field on the **Project Options** dialog box. This is the default value. • **Offline**—When the package is optimized for offline use, the entire package is included in feature block 1 and will be streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable users to use the App-V package when not connected to the App-V server and if you want to eliminate network traffic when the App-V package is being used. • **Stream**—When the package is optimized for streaming use, only the shortcut targets which are included in feature block 1 are streamed to the client at start up. Feature block 2 can contain additional functionality of the App-V package that is not necessary to launch the application. While the App-V package is being used, the files in feature block 2 are streamed in small packets on an as-needed basis. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*

Table 10-19 • Packages Tab / Microsoft App-V Options (All Versions) Category of Properties Window

Property	Description
Append Version	Specify the default App-V versioning value. Available options are: • **Enabled**—Append the package version to the SFT file name. • **Disabled**—Leave the package version off of the SFT file name. • **Default**—Use the setting that is defined on the **Project Options** dialog box.
Launcher Tool	Set this option to **Enabled** to include the App-V Application Launcher when you build an App-V package. You can use the App-V Application Launcher to test a newly built App-V package before moving it to a deployment server. ***Note*** *• The default for the* ***Launcher Tool*** *property is set on the* ***Project Options*** *dialog box.*
Upgrade Package	If this package is an upgrade package that should update an earlier version of the application, click the ellipsis button (...) in this setting and browse to the earlier version.
All Files in VFS	Specify whether the files should exist in Windows retargetable folders within the virtual file system (VFS) folder of the App-V package. Available options are: • **Disabled**—All of the files that correspond with the main installation directory of the application will be put in the root folder of the App-V package's file system. This method adheres to Microsoft App-V best practices. This is the default setting. • **Enabled**—All of the files in the package will be put into Windows retargetable folders within the VFS (virtual file system) folder of the App-V package. This option overrides the effect of specifying a value for **Primary Application Directory**. ***Note*** *• The* ***Enabled*** *option is generally not recommended, but there may be applications for which it is necessary. For example, if a virtualized application does not work as expected, and if it is possible that the application cannot find one of its files because it is searching in a hard-coded path, you may want to select the* ***Enabled*** *option.*

Microsoft App-V Options (Version 5.x)

The **Microsoft App-V Options (Version 5.x)** category includes the following properties:

Table 10-20 • Packages Tab / Microsoft App-V Options (Version 5.x) Category of Properties Window

Property	Description
Expand App-V Package	Use this option to expand an existing App-V 5.x package on the system before performing the sequencing. This is useful for specifying middleware and dependency App-V packages such as Java runtime. Click the browse button and select the App-V package to expand.

Table 10-20 • Packages Tab / Microsoft App-V Options (Version 5.x) Category of Properties Window

Property	Description
Named Objects Interaction	Specify **Enabled** to enable all named objects to interact with the local system. The default setting is **Disabled** to keep these objects isolated from the local system. Named objects include application's events and mutexes among other things. This is an advanced setting that typically does not need to be changed from the default. It is only compatible with the **App-V 5.x with AdminStudio** approach for package creation.
COM Objects Interaction	Specify **Enabled** to allow programs on the local system to interact with all COM objects present in the virtual package. The default setting is **Disabled** in order to isolate all COM objects of the virtual package. This is an advanced setting that typically does not need to be changed from the default. It is only compatible with the **App-V 5.x with AdminStudio** approach for package creation.
Full VFS Write Mode	Set this option to **Enabled** to give the virtual application full write permissions to its VFS (virtual file system) files and folders. ***Note*** • *The **Full VFS Write Mode** feature was introduced in App-V 5.0 SP2 HotFix 4.*

Microsoft App-V Options (Version 4.x)

The **Microsoft App-V Options (Version 4.x)** category includes the following properties:

Table 10-21 • Packages Tab / Microsoft App-V Options (Version 4.x) Category of Properties Window

Properties	Description
Server Host	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter **%SFT_SOFTGRIDSERVER%** to indicate an environment variable. ***Note*** • *If you enter* **%SFT_SOFTGRIDSERVER%**, *you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Server Port	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.

Table 10-21 • Packages Tab / Microsoft App-V Options (Version 4.x) Category of Properties Window

Properties	Description
Server Path	Specify the relative path on the virtual application server where the software package is stored and from which it will be streamed. ***Note*** *• This information is required to create a package if the* `.sft` *file will be stored in a subdirectory of* `CONTENT`*; otherwise, this information is not required.*
Server Protocol	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: • **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. • **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. • **FILE**—The App-V package are streamed from a file share. • **HTTP**—The hypertext transport protocol streams the App-V package. • **HTTPS**—The secure hypertext transport protocol streams the App-V package.
Root Folder Name	Specify the root folder of the App-V package's file system. During run time, the Application Virtualization Client mounts the package's file system to the App-V virtual drive; the Q drive is the default. The long and short names of the root folder must be unique because two packages with the same root folder name cannot be deployed simultaneously.
Dynamic Suites	Enter a semicolon-delimited list of OSD or SFT files to be dynamically suited with this package, or click the ellipsis button (...) and select the OSD or SFT files to be suited. If a file must be present for this package to work properly, append the following to the file name: `:MANDATORY`
Compression	Specify whether to compress this App-V package by selecting one of the following options: • **Compressed**—Compress the App-V package. • **Uncompressed**—Do not compress the App-V package. • **Default**—Use the **Compression** property option that is selected on the **Project Options** dialog box.
Runtime Drive	Enter the App-V client runtime drive. If you do not enter a value, one of the following values will be used: • **If a value is set on the Project Options dialog box**, that value will be used. • **If no value is set on the Project Options dialog box**, the default value of `Q:\` will be used.

Table 10-21 • Packages Tab / Microsoft App-V Options (Version 4.x) Category of Properties Window

Properties	Description
File System Diagnostic	Set this property to **Enabled** if you want to include the Windows Command Prompt application when you build an App-V package so that you can browse the virtual file system at runtime from within the virtual environment. If this property is set to **Enabled**, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment. You can use `Virtual File System.osd` to view the existing files and folders on the computer plus the files and folders for the virtual package. ***Note*** *• The default value for the* ***File System Diagnostic*** *property is set on the* ***Project Options*** *dialog box.*
Registry System Diagnostic	Set this property to **Enabled** if you want to include the Registry Editor (`regedit.exe`) when you build an App-V package so that you can browse the registry at runtime from within the virtual environment. If this property is set to **Enabled**, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment. You can use `Virtual Registry.osd` to view the existing registry on the computer plus the registry for the virtual package. ***Note*** *• The default value for the* ***Registry System Diagnostic*** *property is set on the* ***Project Options*** *dialog box.*

MSIX Signing Options

The **MSIX Signing Options** category includes the following properties:

Table 10-22 • Packages Tab / MSIX Signing Options Category of Properties Window

Property	Description

Select Certificate for Signing	Use to specify the signing option:
	• **Certificate File (.pfx)** Specify a path and password to a valid PFX certificate file.
	• **Certificate Store** Specify the following certificate store details:
	
	Note *• If you select the Certificate Store option, make sure that the Certificate has been imported. For more details on importing the certificate, see Import Certificate.*
	  • **Certificate Store Location** Select either User or Machine.
	  • **Certificate Store Name** Select the store name from the drop-down list.
	  • **Certificate Subject** Specify the subject of the certificate.
	
	Note *• Certificate Subject populates with respect to the selected certificate.*

Packages Tab Buttons

The **Packages** tab includes the following buttons:

Table 10-23 • Packages Tab Buttons

Property	Description
Add Packages	Click this button to launch the Package Import Wizard, which you can use to add packages to the **Packages** tab.
Remove Selected	Click this button to remove the selected package from this list. You can also click the Delete key. ***Note*** *• A package is selected for removal when you click on it and it becomes highlighted, not by selecting the package's check box. Use the* ***Ctrl*** *key to select multiple packages.*

Icons Used on the Packages Tab

The following icons are used to display package status on the **Packages** tab:

Table 10-24 • Select Packages Panel

Column	Icon	Description
Status		Package is in the process of being repackaged.
		Package has been successfully repackaged.
		Repackaging has failed.
		Package is waiting in line to be repackaged or to be virtualized.
		A soft timeout has occurred, meaning that the package's **Soft Timeout** time period has elapsed. This could occur because the package is very large and is taking an unusually long time to repackage, or because you have set the **Soft Timeout** value too low, or because the installer is waiting for some kind of user input (meaning that the installation was not silent).
		The last conversion run of this package was cancelled.
Package		Identifies the source package (`.msi` or `.exe` file).
		Identifies the repackaged `.msi` file.
		Identifies the virtual application that was successfully created.

Table 10-24 • Select Packages Panel

Column	Icon	Description
Virtualization Readiness		**Ready** Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
		Requires repackaging Package must be repackaged before it can be successfully virtualized.
		Undetermined The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
		Virtualization not supported Automated Application Converter has determined that virtualization is not supported due to one of the following issues (described in Application Virtualization Compatibility Tests): • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.*
		Virtualization not recommended Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues (described in Application Virtualization Compatibility Tests): • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology.

Shortcut Menu Commands on Packages Tab

When you right-click an item in the **Package** list on the **Packages** tab, the following commands are available on the shortcut menu:

Table 10-25 • Shortcut Menu Commands on Packages Tab

Command	Description
Explore	Open the directory that contains the selected file.
Remove	Permanently removed the selected packages from the project.
Test Virtualization Readiness	Select to test the selected package for virtualization readiness. The possible results are: • —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • —Package must be repackaged before it can be successfully virtualized. • —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
Launch Package for Testing	Select to install and run the selected application on a virtual machine. See Testing Packages.
Connect to Machine	When the Running icon is displayed in the **Status** column, indicating that the package is in the process of being repackaged, you can select **Connect to Machine** from the shortcut menu to connect to the virtual machine via Remote Desktop on which this package is being repackaged.
Package Import Wizard	Select this option to launch the Package Import Wizard to import packages to this project.

Machines Tab

On the **Machines** tab of the Automated Application Converter, you add a list of clean virtual machine images to use during automated repackaging.

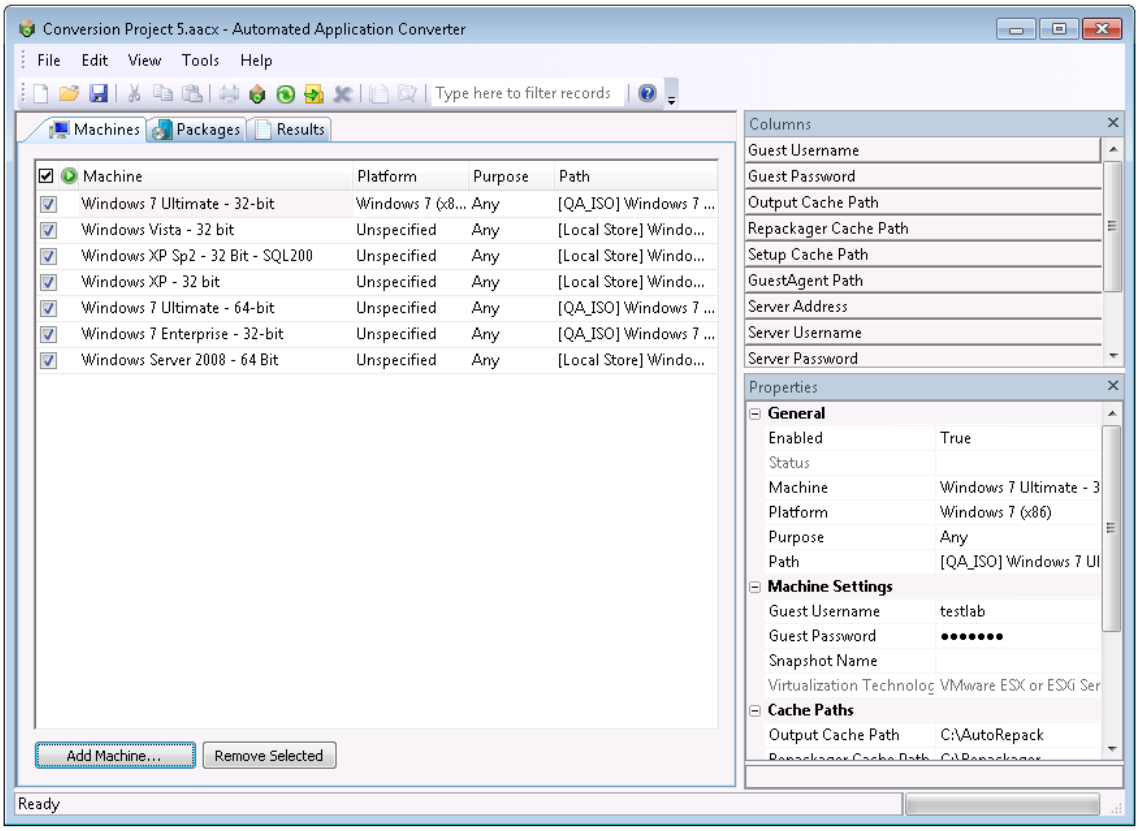


Figure 10-19: Machines Tab

This section includes the following information about the **Machines** tab:

- Adding Virtual Machines to the List
- Viewing Virtual Machine Information on the Machines Tab
- Machines Tab Properties
- Shortcut Menu Commands on Machines Tab

Adding Virtual Machines to the List

To add a virtual machine to the list, click **Add Machine** to open the Virtual Machine Import Wizard, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard. You will then be prompted for login information and other relevant data required to prepare the machine.

Note • *Before you add a machine to this list, you need to perform the steps listed in Preparing Your Virtual Machines for Use With the Automated Application Converter to enable automatic login and to create a clean snapshot.*

To perform repackaging, you have the option of selecting one virtual machine or multiple virtual machines (that can be used simultaneously to speed up the repackaging of multiple setups). You can also specify that you want to use only virtual machines of a specific operating system platform.

When each virtual machine finishes repackaging a package, it is reverted to its clean snapshot image, and then starts repackaging the next package in the list.

Viewing Virtual Machine Information on the Machines Tab

By default, the **Machines** tab lists the **Status**, **Machine**, **Platform**, **Purpose**, and **Path** columns for each machine. Additional columns of information can be viewed by selecting one of the fields in the **Column** selector area and dragging it onto the list. Also, the properties for these **Columns** for the selected machine can be viewed in the **Properties** window when that column is selected in the **Column** selector area.

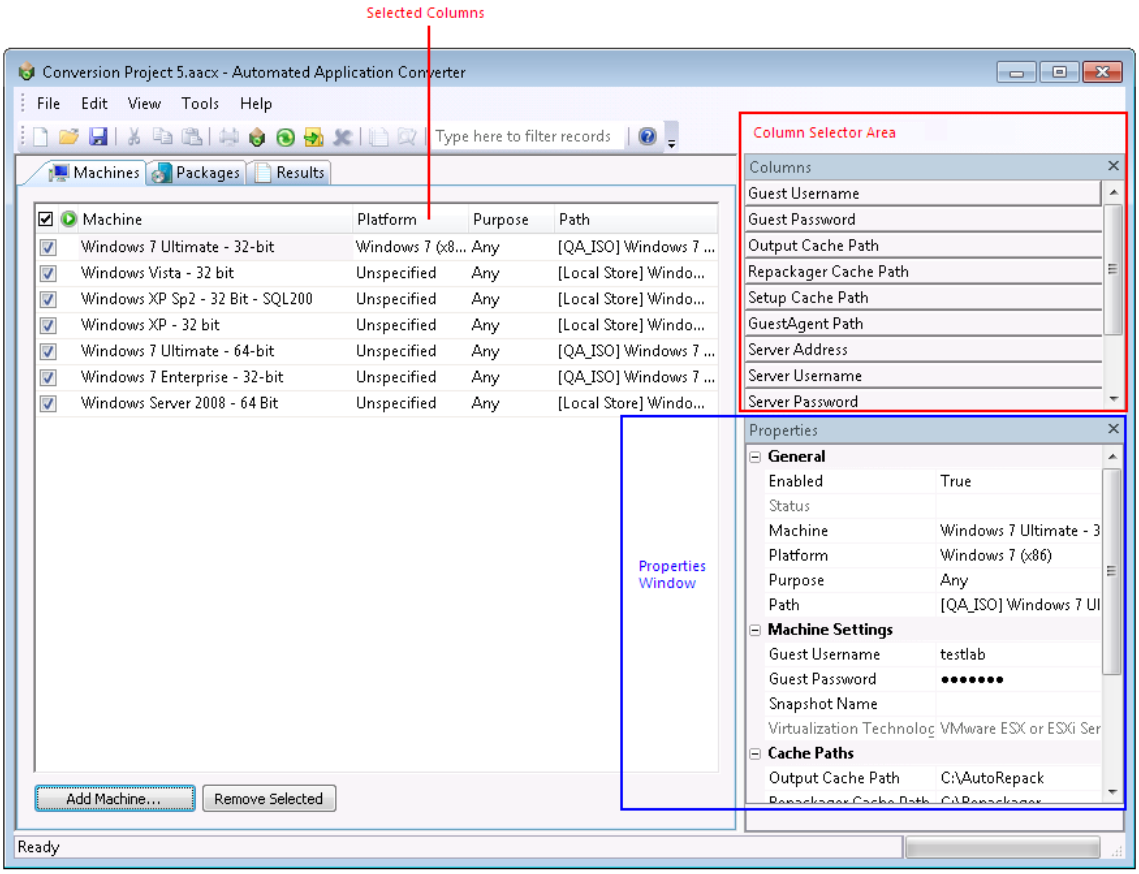


Figure 10-20: Column Selector Area on the Machines Tab

Note • *You can sort this list, change the columns that are displayed, change the column order, resize the columns, and group the list by a specific column. See Using List Features for more information.*

Machines Tab Properties

The **Machines** tab includes the following properties and information:

Table 10-26 • Machines Tab

Property	Description	
☑	Selection column. To select a virtual machine to use for automated repackaging, click the check box in this column.	
Status	The icon displayed in this column indicates the status of the virtual machine:	
		Virtual machine is in use.
		The Automated Application Converter encountered an error when attempting to connect to this virtual machine. or when rolling back to a snapshot on this virtual machine.
		The Automated Application Converter is waiting for the virtual machine to boot up.
	(No icon)	Virtual machine is not currently in use.
		The last conversion run on this virtual machine was cancelled.
Machine	Name of the virtual machine image.	
Platform	Field that identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.	
Path	Path on the server or file system to the virtual machine image file.	

Table 10-26 • Machines Tab (cont.)

Property	Description
Purpose	By default, virtual machines that you add to the **Packages** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options: • **Repackaging**—Virtual machine will only be used to perform automated repackaging. • **Testing**—Virtual machine will only be used to test packages. You test a package by right-clicking on it on the **Packages** tab and selecting **Launch Package for Testing** from the shortcut menu. You will then be prompted to install and run that package on a virtual machine. • **Any**—Make this virtual machine available for use during both automated repackaging and package testing. This is the default value. ***Important*** • *If the **Purpose** column is not listed in the **Machines** list, you can select it in the **Columns** area and drag it to the list, or you can edit the **Purpose** value in the Properties window.* ***Note*** • *The **Launch Package for Testing** functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*
Guest Username	The user name to use to login to this virtual machine.
Guest Password	The password to use to login to this virtual machine.
Output Cache Path	Specify the location for the repackaged output on the virtual machine. By default, this value is `C:\AutoRepack`.
Repackager Cache Path	Specify the location where Repackager will be installed on the virtual machine. By default, this value is `C:\Repackager`.
Setup Cache Path	Specify the location where the package will be copied to on the virtual machine. By default, this value is `C:\AppSetup`.
GuestAgent Path	Specify the location where the `GuestAgent.exe` file will be installed on the virtual machine. By default, this value is `C:\GuestAgent.exe`.
Server Address	The address of the virtual machine server on which this virtual machine is found. This may be a host name or a URL.
Server Username	The user name of the account used to access the virtual machine server.
Server Password	The password of the account used to access the virtual machine server.

Table 10-26 • Machines Tab (cont.)

Property	Description
Snapshot Name	Name of the snapshot to revert to before starting an automated repackaging session. This is only used if the virtualization technology supports named snapshots. If this value is not specified, but named snapshots are supported on the virtualization technology, the default name of `AutoRepack_Base` will be used.
Virtualization Technology	The virtualization technology powering this virtual machine.
App-V 5.x Sequencer Snapshot	Enter the name of the snapshot to revert to before starting conversion using the App-V 5.x Sequencer. ***Important*** • *Both the Microsoft App-V 5.x Sequencer and the Virtual Machine Preparation client must be installed on this snapshot. For more information, see Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer.*
App-V 5.x Client Snapshot	Enter the name of the snapshot to revert to before testing an App-V 5.x package. This snapshot will be used if the user right-clicks on this App-V 5.0 virtual package on the **Packages** tab of Automated Application Converter and selects **Launch Package for Testing** from the shortcut menu. ***Important*** • *Both the Microsoft App-V 5.x client and the Virtual Machine Preparation client must be installed on this snapshot. For more information, see Preparing a Snapshot for App-V 5.0 Testing Using the App-V 5.0 Client.*
Add Machine	Click to launch the Virtual Machine Import Wizard, which you can use to add virtual machines to the **Machines** tab.
Remove Selected	Click to remove the selected virtual machine from this list. ***Note*** • *A virtual machine is selected for removal when you click on it and it becomes highlighted, not by selecting the virtual machine's check box. Use the **Ctrl** key to select multiple machines.*

Shortcut Menu Commands on Machines Tab

When you right-click on an item in the **Machines** list on the **Machines** tab, the following commands are available on the shortcut menu:

Table 10-27 • Shortcut Menu Commands on Machines Tab

Command	Description
Connect to Machine	When the Running icon is displayed in the **Status** column, indicating that this virtual machine is currently being used to perform repackaging, you can select **Connect to Machine** from the shortcut menu to connect to this virtual machine via Remote Desktop.
Remove	Select to remove the selected machine from this project.
Machine Import Wizard	Select to add virtual machines to this project using the Virtual Machine Import Wizard.

Results Tab

On the **Results** tab, the results of each virtualization conversion run for this project are listed.

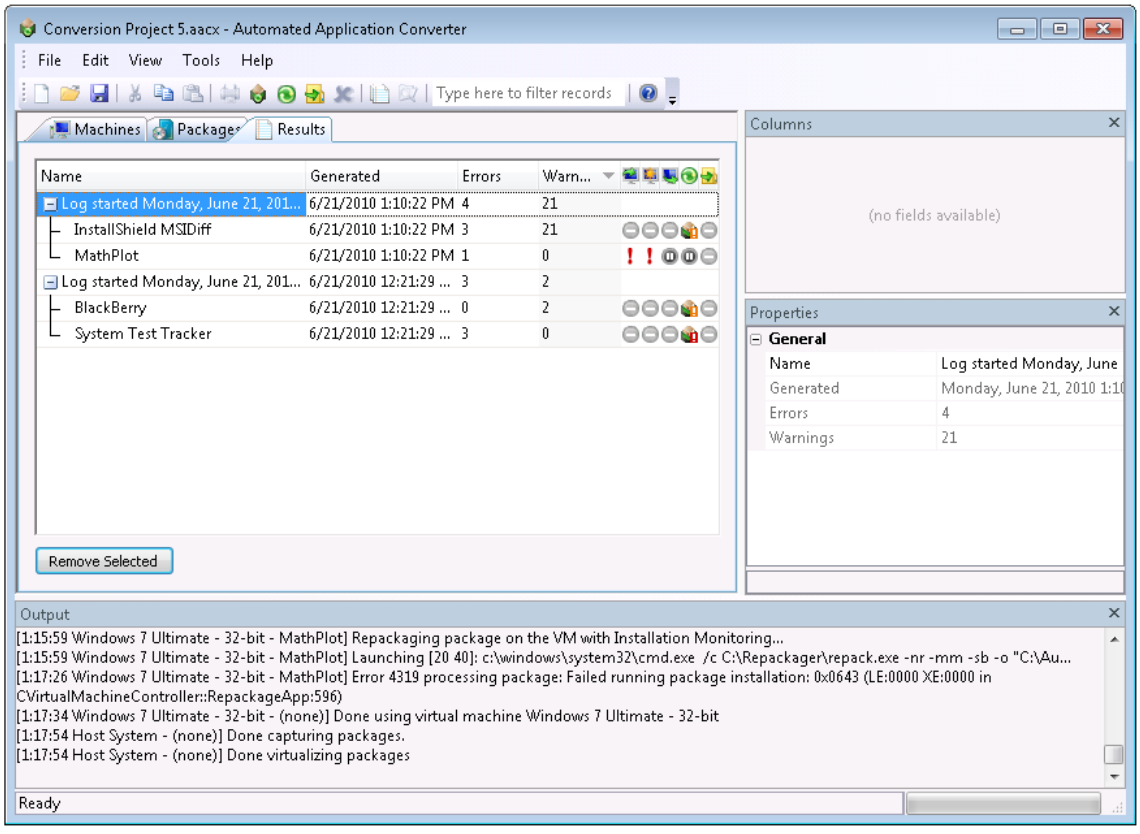


Figure 10-21: Results Tab

This section includes the following information about the **Results** tab:

- Results Tab Properties
- Icons Used on the Results Tab
- Shortcut Menu Commands on Results Tab

Results Tab Properties

The **Results** tab includes the following properties and information:

Table 10-28 • Results Tab

Property	Description
Name	List of logged results for each run of this project. The log is identified by the date and time it was started, and the packages that were part of this run are listed in a tree structure under the log title. Click the plus sign to expand the listing.
Generated	Date and time the conversion of each package began.
Errors	The number of errors generated for each package in this run is listed in this column next to each package. The cumulative sum of all errors generated for all of the packages in the run is listed in this column next to the parent Log row.
Warnings	The number of warnings generated for each package in this run is listed in this column next to each package. The cumulative sum of all warnings generated for all of the packages in the run is listed in this column next to the parent Log row.
Results Icons	Icons in these columns indicate the status of each of the steps of the repackaging and conversion process. See Icons Used on the Results Tab for detailed information.

Icons Used on the Results Tab

The following icons are used on the **Results** tab:

Table 10-29 • Icons Used on Results Tab

Column	Icon	Description
Copy In		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed.
Repackage **Copy Out**		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed, but warnings were encountered. View the results AdminStudio Automated Application Converter Log Report for detailed information on these warnings.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) failed. • **Copy In**—Error could have been caused by not being able to connect to the virtual machine. • **Repackage**—Error means that repackaging has failed. • **Copy Out**—Error could mean that you ran out of hard drive space at the package source location or that there is a permission problem preventing you from writing to the selected directory. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was skipped. Possible reasons that the operation was skipped could be: • **Repackaging not required**—Because repackaging was not required, these three operations were not required. • **Could not connect to virtual machine**—The Automated Application Converter could not successfully connect to the virtual machine, so therefore the **Repackage** and **Copy Out** operations were skipped.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is currently being performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is still being performed even though a warning was generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation was cancelled.

Table 10-29 • Icons Used on Results Tab (cont.)

Column	Icon	Description
Conversion Column		Package was converted to a virtual application successfully.
		Package was converted to a virtual application, but warnings were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Package was converted to a virtual application, but errors were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		The Automated Application Converter was unable to convert this package to a virtual application.
		Conversion is in progress.
		Conversion is in progress, but a warning has been generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		An error was generated when converting one of the virtual formats which caused it to fail. However, the conversion to another one of the selected virtual formats continues.
		Conversion was cancelled.

Shortcut Menu Commands on Results Tab

When you right-click on a log node on the **Results** tab, the following commands are available on the shortcut menu:

Table 10-30 • Shortcut Menu Commands on Results Tab

Command	Description
View Report	Select to view the AdminStudio Automated Application Converter Log Report for the selected run.
Explore	Select to open the directory where the selected log file is located.
Remove	Select to delete the selected log file.

Note • *If you have right-clicked on a child node under the parent Log node, the shortcut menu is disabled.*

Menus & Toolbar Buttons

The Automated Application Converter user interface includes the following menus, commands, and toolbar icons:

Table 10-31 • Automated Application Converter Menus and Commands

Menu	Command	Icon	Description
File	**New**		Click to open a new Automated Application Converter project file.
	Open...		Click to open an existing Automated Application Converter project file.
	Save		Click to save the open Automated Application Converter project file.
	Save As...		Click to save the open Automated Application Converter project file in a new location or using a different name.
	Recently Opened Items		Lists the most recently used list of Automated Application Converter projects. Click to open.
	Exit		Click to exit the Automated Application Converter.
Edit	**Copy**		Copy selected text. You can then paste it in an external program such as Notepad or Microsoft Word.
	Select All		Select all of the Packages, Machines, or Results in the list.
	Select None		Unselect all selected items.
View	**Toolbars**		Select one of the following options: • **Standard**—Toggles the display of the toolbar. • **Customize** —Select to customize which menu commands are displayed on the toolbar.
	Windows		Toggle the display of the **Columns** selector area, the **Properties** window and the **Output** window.
	Status Bar		Toggles the display of the status bar at the bottom of the interface.

Table 10-31 • Automated Application Converter Menus and Commands (cont.)

Menu	Command	Icon	Description
Tools	**Project Wizard....**		Click to open the Application Conversion Project Wizard, which guides you step-by-step through the entire virtualization process: adding virtual machines, adding packages, and virtualizing applications.
	Application Conversion Wizard...		Click to open the Application Conversion Wizard, which you can use to select the virtualization format you want to convert to and to perform conversion of the selected packages on the selected virtual machines.
	Explore		Open the directory containing the selected package.
	View Report		After selecting the top level node of a conversion run log on the **Results** tab (`Log started Monday, April 01, 2010...`), click this to open the AdminStudio Automated Application Converter Log report. See AdminStudio Automated Application Converter Log Report for more information.
	Test Virtualization Readiness		Click to test the selected package for virtualization readiness. The possible results are: • —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • —Package must be repackaged before it can be successfully virtualized. • —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
	Cancel Virtualization		Cancel current conversion run.

Table 10-31 • Automated Application Converter Menus and Commands (cont.)

Menu	Command	Icon	Description
Help	**Contents**		Launches the Help Library, displaying the **Contents** tab.
	Index		Launches the Help Library, displaying the **Index** tab.
	Search		Launches the Help Library, displaying the **Search** tab.
	About the Automated Application Converter		Displays the **About the Automated Application Converter** dialog box.

Output Window

When a virtualization run is performed, the output messages and results are displayed in the Output Window. There is also additional progress indicators displayed near the Output window:

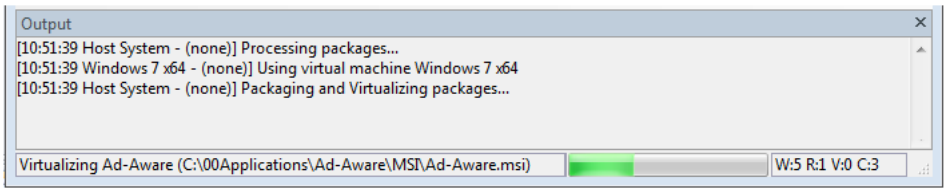


Figure 10-22: Output Window

The following information is listed:

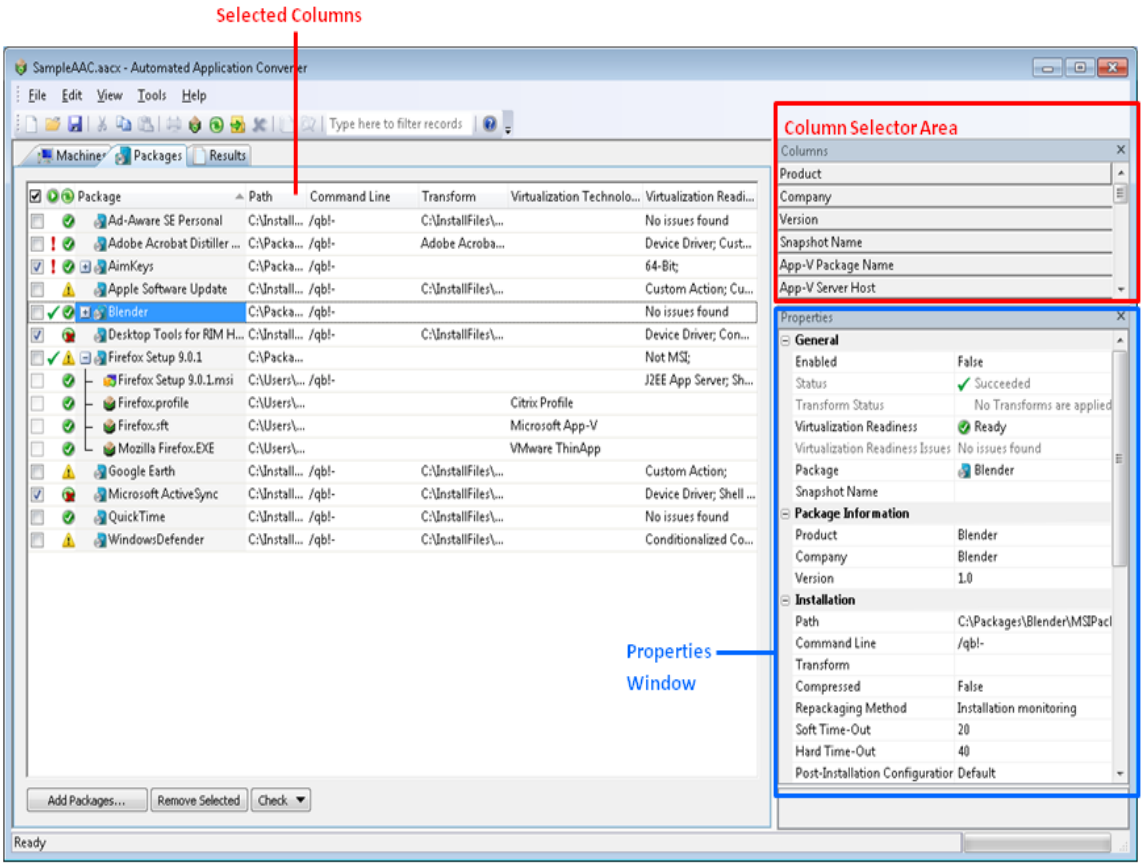
- **Messages**—Messages are listed in the **Output** window
- **Current file**—The name of the current file being processed is listed in the lower left.
- **Progress bar**—A progress bar is displayed at the bottom of the screen.
- **Count**—The count of packages in each of the following categories is displayed at the lower right:
  - **W:5**—Number of packages that are **W**aiting to be processed.
  - **R:1**—Number of packages that are being **R**epackaged.
  - **V:0**—Number of packages that are being converted or **V**irtualized into MSI and/or virtual packages.
  - **C:3**—Number of applications that have finished processing, including **C**ompleted and failed applications.

You can copy the results in the Output window and paste them in an outside location, such as Notepad or Microsoft Word.

All of the messages and results listed in the Output window can also be viewed in the AdminStudio Automated Application Converter Log report. See AdminStudio Automated Application Converter Log Report.

Column Selector and Properties Windows

By default, the **Packages** and **Machines** tabs list several columns of information. However, additional columns of information can be viewed by selecting one of the fields in the **Column Selector** area and dragging it onto the list. Also, the values for these fields for the selected package or machine can be viewed in the **Properties** window.



A description of each of these properties can be found in Packages Tab and Machines Tab.

Note • *You can sort the lists on the **Package** and **Machines** tabs, change the columns that are displayed, change the column order, resize the columns, and group the lists by a specific column. See Using List Features for more information.*

AdminStudio Automated Application Converter Log Report

The AdminStudio Automated Application Converter Log is an HTML report you can view that lists the following information for each conversion run:

- **Machines**—List of the virtual machines used in the conversion run.
- **Packages**—List of the packages that included in this conversion run. The packages are linked to the Package Conversion Messages section of the report for that package.
- **Log Results / General Messages**—Start time, number of errors and warnings generated, and general processing messages.

- **Log Results / Package Conversion Messages**—Conversion messages for each package that the Automated Application Converter attempted to convert.

The following is an example of an AdminStudio Automated Application Converter Log report:

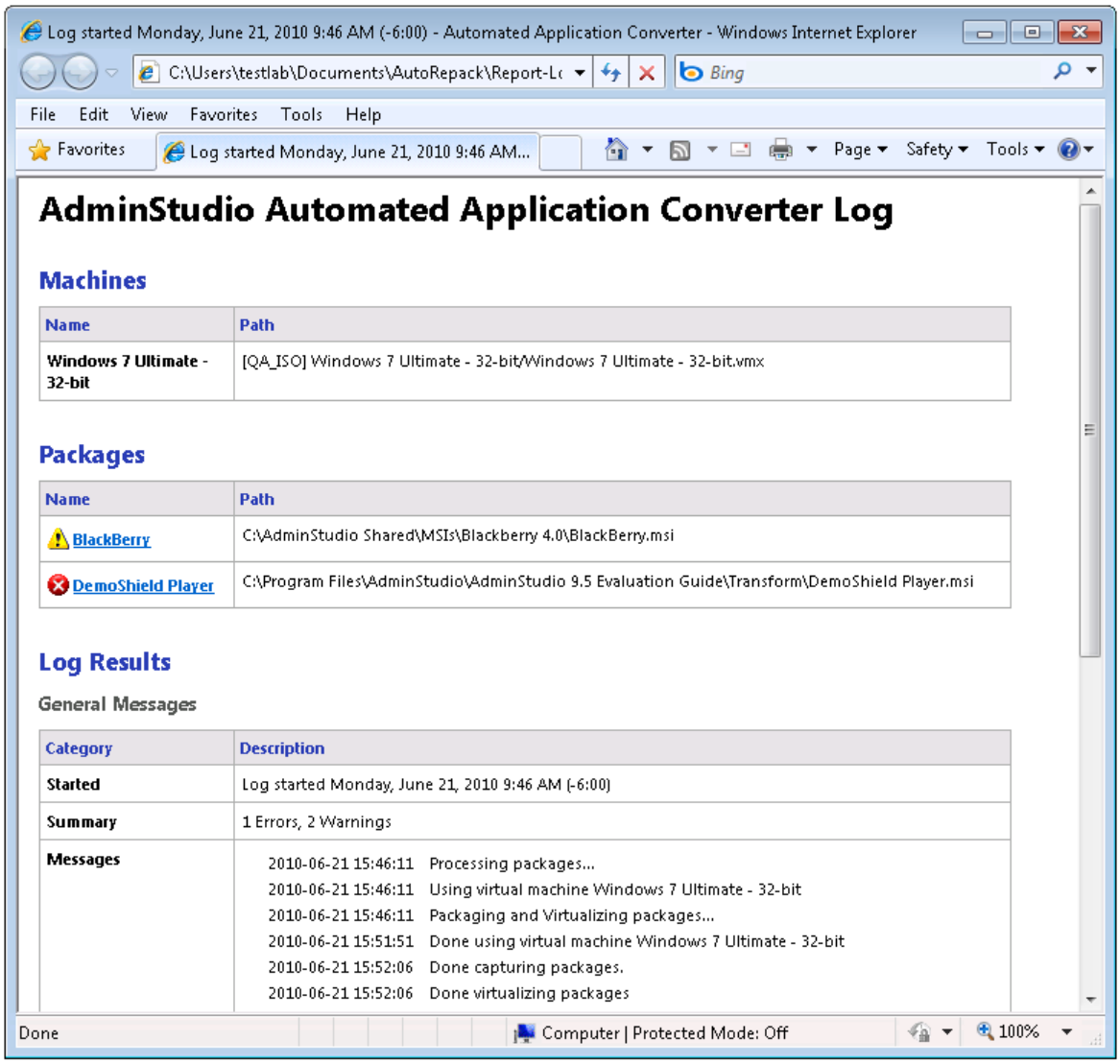


Figure 10-23: AdminStudio Automated Application Converter Log Report

Viewing an AdminStudio Automated Application Converter Log Report

To view an AdminStudio Automated Application Converter Log report, perform the following steps:

Task ***To view an AdminStudio Automated Application Converter Log report:***

1. Open the **Results** tab.
2. Select the top level node of a conversion run log (`Log started Monday, June 21, 2010...`), and do one of the following:

- Click **View Report** on the **Tools** menu.
- Select **View Report** from the shortcut menu.
- Click the **Reports** icon on the toolbar.
- Press Ctrl+R.

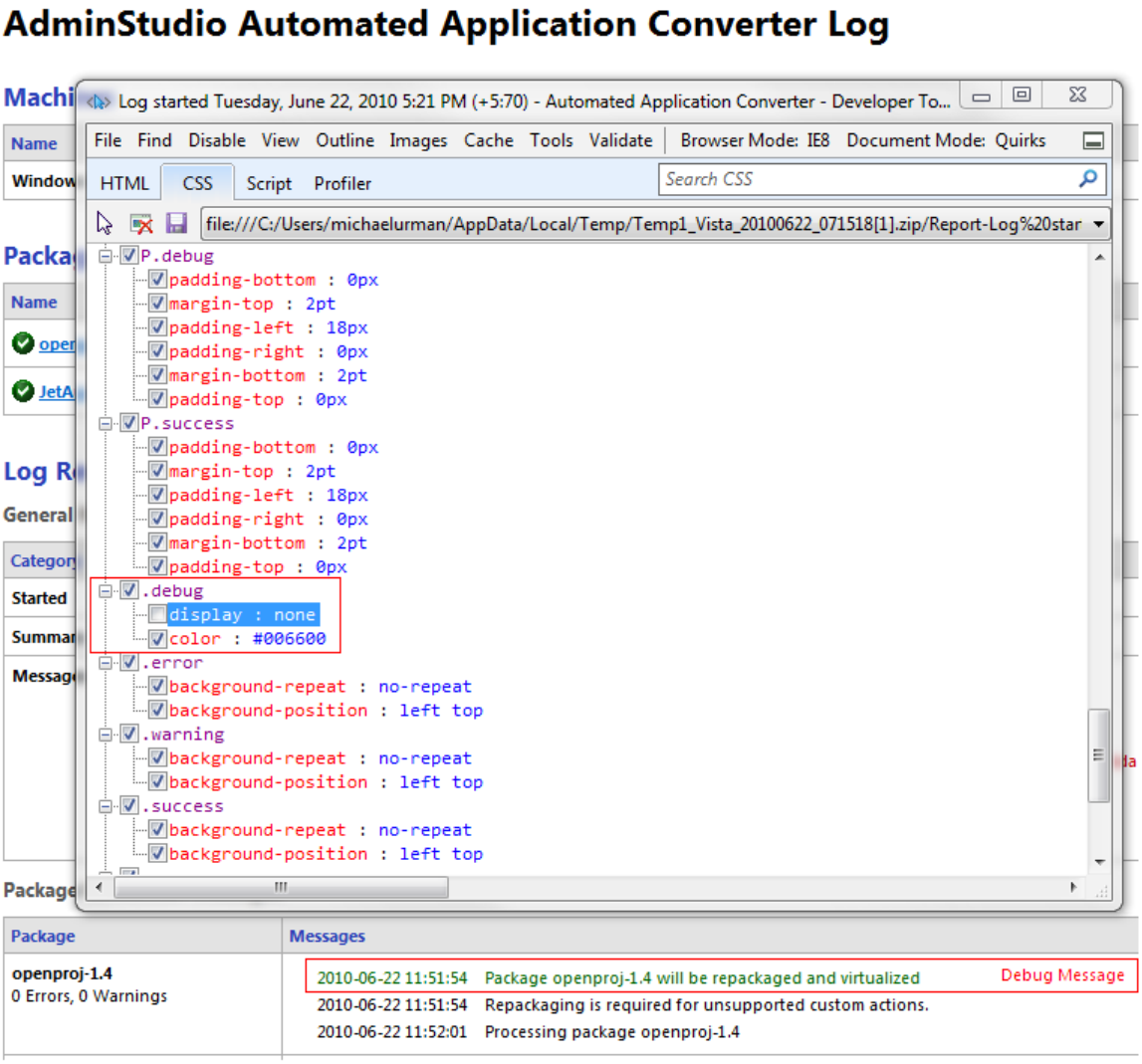
The report opens in a new browser window.

Viewing Debug Messages

By default, debug messages that occur during a conversion run are saved in the log report, but the display of those debug messages is turned off. However, if you are using Microsoft Internet Explorer 11 as your default browser, you can choose to view those debug messages by performing the following steps:

Task ***Viewing debug messages in an Automated Application Converter Log Report:***

1. Open an Automated Application Converter Log Report as described in Viewing an AdminStudio Automated Application Converter Log Report.
2. Press F12. The **Developer Tools** window opens.
3. Click on the **CSS** tab. The CSS file for the log report opens, displaying checkmarks next to each class and each property.
4. Scroll down to locate the following class: `.debug` (not `P.debug`).
5. Clear the selection of the `display : none` property, as shown below:



6. Return to the log report window. The debug messages are now displayed in green.

Using List Features

All of the lists displayed in the Automated Application Converter user interface—including the lists shown on Wizard panels—implement the same list features, which allow you to group a list by any column, sort a list by any column, resize list columns, change which columns are displayed, and change column order.

- Sorting Lists
- Changing Which List Columns Are Displayed
- Changing Column Order
- Resizing List Columns
- Grouping Lists

Sorting Lists

You can sort lists by any column by clicking on the header of the column you want to sort by or by right-clicking on a column header and making a selection from the shortcut menu.

Task ***To sort a list by a column heading:***

1. Open the **Machines**, **Packages**, or **Results** tab.
2. To sort by a column heading, click on the column heading to toggle through the three sort order states, which are identified by a visual indicator:
   - **Sorted in ascending order**—When the column is sorted in ascending order, an up arrow is displayed in the header row.
   - **Sorted in descending order**—When the column is sorted in descending order, a down arrow is displayed in the header row.
   - **Not sorted**—When the column is not sorted (meaning that the list is either sorted by another column or is just listed in the default order that the records appear in the database), no arrow is displayed.

   Tip • *Another way to do this is to right-click on the column header of the column you want to sort, and then select* ***Sort Ascending*** *or* ***Sort Descending*** *from the shortcut menu.*
3. To sort just the children of the top level items (not the top level items), right-click on the column header of the column you want to sort by, and then select **Sort Children** from the shortcut menu.

Changing Which List Columns Are Displayed

To improve readability or clarity, you can choose to remove a column from a list. When you remove a column from a list, you are just turning off the display of that column, not deleting the data that was in that column. You can restore a removed column to the list at any time.

Adding/Restoring a Column to a List

To restore the display of a hidden column to a list, perform the following steps.

Task ***To restore the display of a deleted column to a list:***

1. To restore the display of a deleted column to a list, right-click anywhere in the heading row.
2. Point to **Columns** in the shortcut menu. A list of all of the available columns for this list is displayed.
3. Select the name of the column that you want to restore to the list.

Tip • *To add a column to the list, you can also click and drag a column header from the* ***Column Selector*** *area to the header row of the list.*

Removing a Column from a List

To remove a column from a list, perform the following steps.

Task ***To remove a column from a list:***

1. To remove a column from a list, right-click anywhere in the heading row.
2. Point to **Columns** in the shortcut menu. A list of all of the available columns for this list is displayed, with those that are currently selected for display indicated by a check mark.
3. Select the name of the column to clear the selection.

   The column is now hidden.

   Note • *To remove a column to the list, you can also click and drag a column header from the header row of the list to the* ***Column Selector*** *area.*

Changing Column Order

To help compare the values of columns, you can click and drag to change the order of columns in a list.

Task ***To change column order:***

1. Click on the column header of the column you wish to relocate.
2. While holding the mouse button down, drag the column header on top of the rule between two columns.
3. When the red arrows appear, release the mouse button to perform the move.

Resizing List Columns

To improve the readability of the values in a column of a list, you can click and drag to resize a column.

Task ***To resize a column:***

1. Position your cursor at the right side of the column header of the column that you want to resize and click. After you click, the cursor turns into a double-arrow icon:
2. While holding the mouse button down, drag the edge of the column left or right until it is the desired width.

Grouping Lists

This section explains how to group a list by a column, ungroup a list, and create subgroupings.

- Grouping an Ungrouped List
- Ungrouping a List

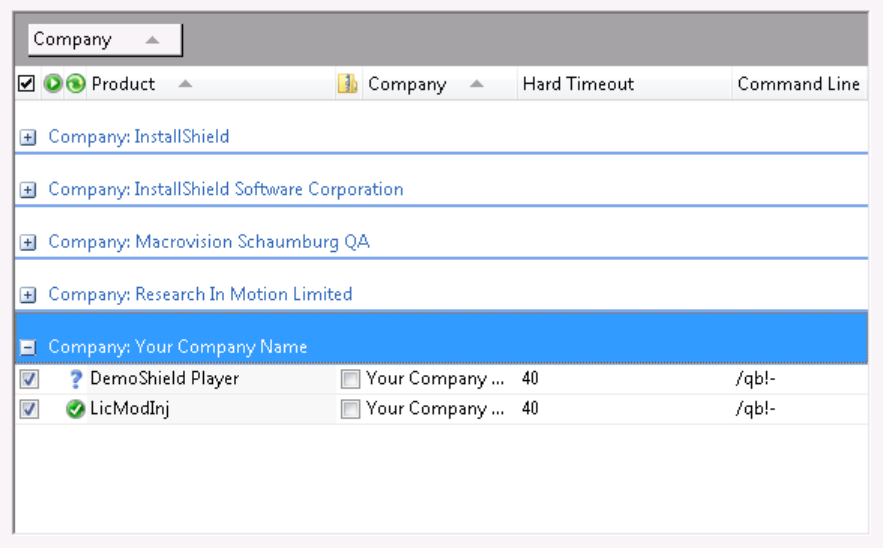
Grouping an Ungrouped List

If a list is not grouped by a column, no Group By Box is displayed.

To group by a column, perform the following steps.

Task ***To group a list by a column heading:***

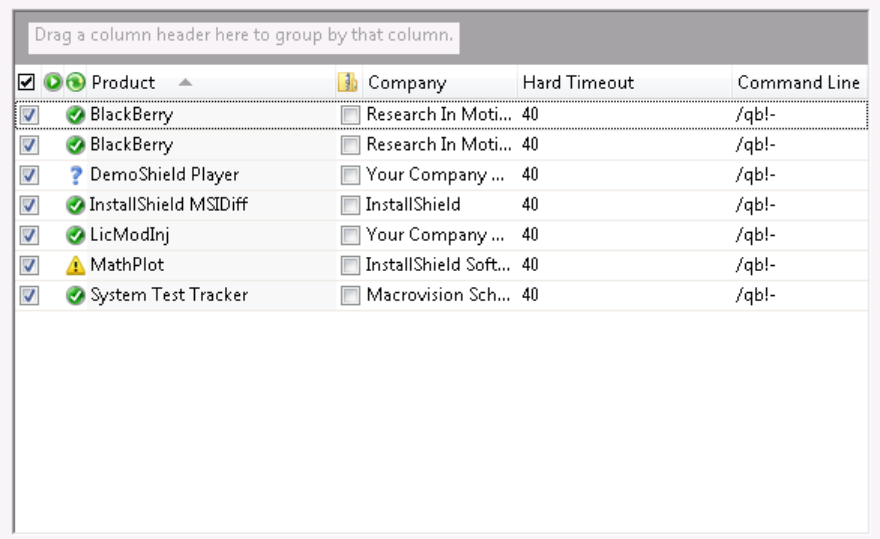
1. Right-click on the column header that you want to group by and select **Group By [Column Name]** from the shortcut menu.



When you group by a column header, the following occurs:

- The name of the column header that you have chosen now appears in the Group By Box.
- The name of the column header that you selected and each of its values (in the format of `Column Name: Value`) is now listed in a bar at the left of the list, with all records associated with that value of that column grouped underneath that bar.
- One group (bar) appears for each of the values of the selected column. Click the plus sign to expand the list.

2. To display the Group By box without performing any grouping, select **Show Group Header** from the shortcut menu. The text `Drag a column header here to group by that column.` is displayed in the **Group By** box:



Note • *When the Group By box is displayed, you can perform grouping by dragging a column header to the Group By box.*

Ungrouping a List

Perform the following steps to ungroup a list.

Task ***To ungroup a list or change a list's Group By column:***

1. Click on the name of the column header in the Group By Box and drag it back to the header row in the list to the location where you want the column to be displayed.

   Tip • *Another way to do this is to right-click on the column header name and clearing the* ***Group By [Column Name]*** *selection on the shortcut menu.*

   The list is now ungrouped.

2. If you want to choose another column to use to group the list by, follow the steps listed above under Grouping an Ungrouped List.

Wizards

The Application Conversion Project Wizard guides you step-by-step through the entire virtualization process: adding virtual machines, adding packages, and virtualizing packages. You can also choose to perform each of these tasks separately by using one of the other three wizards that are provided:

Table 10-32 • Automated Application Converter Wizards

If you want to ...	Use this wizard ...	Description and Purpose
Add packages	**Package Import Wizard**	Add packages from an AdminStudio Application Catalog or from a local or network file system.
Add virtual machines	**Virtual Machine Import Wizard**	Add virtual machines to use to perform automated repackaging of Windows Installer packages.
Virtualize packages	**Application Conversion Wizard**	Virtualize packages to the virtual formats you specify.

You also use wizards to add packages and virtual machines to Automated Application Converter:

- To add packages to the **Packages** tab, use the Package Import Wizard.
- To add virtual machines to the **Machines** tab, use the Virtual Machine Import Wizard.

Application Conversion Project Wizard

When using the Automated Application Converter to perform batch conversion to virtual packages, there are three main procedures that you perform:

- **Step 1: Select packages**—Select packages to virtualize and/or repackage.
- **Step 2: Select machines**—Select the virtual machines that you want to use during automated repackaging.
- **Step 3: Select formats and perform conversion**—Select the virtualization formats you want to convert to and perform the conversion.

You can use the **Application Conversion Project Wizard** to perform all three of these steps in one guided procedure.

You can launch the Application Conversion Project Wizard in one of two ways:

- **Creating a new project upon product launch**—The **Open Project** panel opens automatically when you launch the Automated Application Converter or when you select **New Project** on the **File** menu.
- **Creating a new project after product launch**—Select **Project Wizard** on the **Tools** menu or when you click the Project Wizard icon on the toolbar, or select **New Project** on the **File** menu.

The Application Conversion Project Wizard includes the following panels:

- Open Project Panel
- Application Conversion Project Wizard Welcome
- Select Package Source

- Connect to an AdminStudio Application Catalog
- Select Packages
- Selected Package List
- Select Virtual Machine Source
- Select Virtual Machines from a Microsoft Hyper-V Server
- Select Virtual Machines from a VMware ESX or ESXi Server
- Select Virtual Machines
- User Credentials
- Initial Configuration Complete
- Select Output Formats
- Automated Repackaging on Virtual Machines
- Application Conversion Project Wizard Complete Panel

Automated Application Converter's Other Wizards

Each of the Automated Application Converter's other three wizards—Virtual Machine Import Wizard, Package Import Wizard, and Application Conversion Wizard—consist of a subset of the panels included in the Application Conversion Project Wizard. The following table lists the panels in each of these three wizards.

Table 10-33 • Breakdown of Panels in the Automated Application Converter Wizards

Panel Name	Application Conversion Project Wizard	Package Import Wizard	Virtual Machine Import Wizard	Application Conversion Wizard
Select Package Source	X	X		
Connect to an AdminStudio Application Catalog	X	X		
Select Packages	X	X		
Selected Package List	X	X		
Select Virtual Machine Source	X		X	
Select Virtual Machines from a Microsoft Hyper-V Server	X		X	
Select Virtual Machines from a VMware ESX or ESXi Server	X		X	
Select Virtual Machines	X		X	
User Credentials	X		X	

Table 10-33 • Breakdown of Panels in the Automated Application Converter Wizards

Panel Name	Application Conversion Project Wizard	Package Import Wizard	Virtual Machine Import Wizard	Application Conversion Wizard
Initial Configuration Complete	X			
Select Output Formats	X			X
Automated Repackaging on Virtual Machines	X			X

Open Project Panel

The **Open Project** panel opens automatically when you launch the Automated Application Converter or when you select **New Project** on the **File** menu.

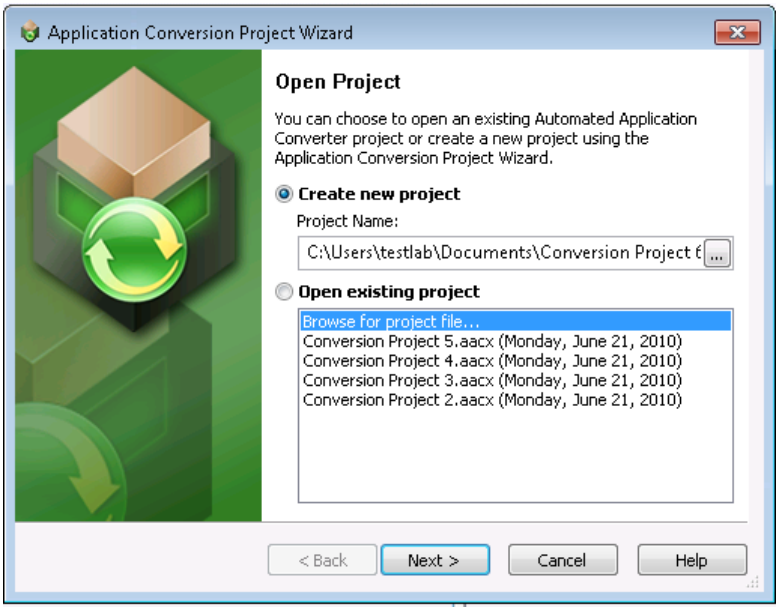


Figure 10-24: Open Project Panel

You have the following options:

- **Create a new project**—If you select the **Create a new project** option, click **Next** to continue with the wizard. You will be prompted to name and save the project when you begin conversion or exit the Automated Application Converter.
- **Select an existing project**—If you select an existing project from the list, click **Finish** to open the project.

Application Conversion Project Wizard Welcome

The **Application Conversion Project Wizard Welcome** panel opens when you select **Project Wizard** on the **Tools** menu or when you click the Project Wizard icon on the toolbar.

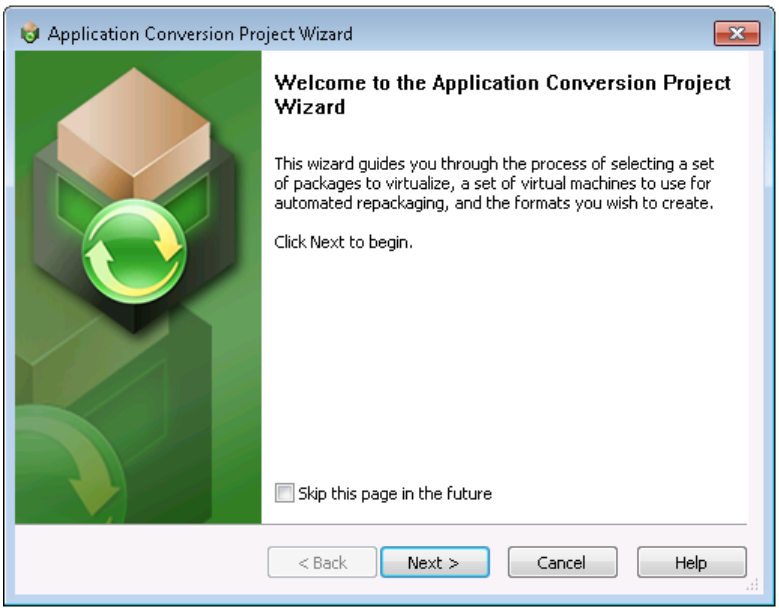


Figure 10-25: Application Conversion Project Wizard Welcome Panel

If you would prefer to perform each of these steps separately, you can instead use the following wizards:

Table 10-34 • Automated Application Converter Wizards

Wizard	Description
Virtual Machine Import Wizard	Use to add virtual machines to your project which can be used to perform automated repackaging into Windows Installer packages.
Package Import Wizard	Use to select packages from an AdminStudio Application Catalog or from a file system to virtualize.
Application Conversion Wizard	Use to select the virtualization format you want to convert to and to perform the conversion.

Select Package Source

On the **Select Package Source** panel, you select the source that contains the packages that you want to virtualize and/or repackage.

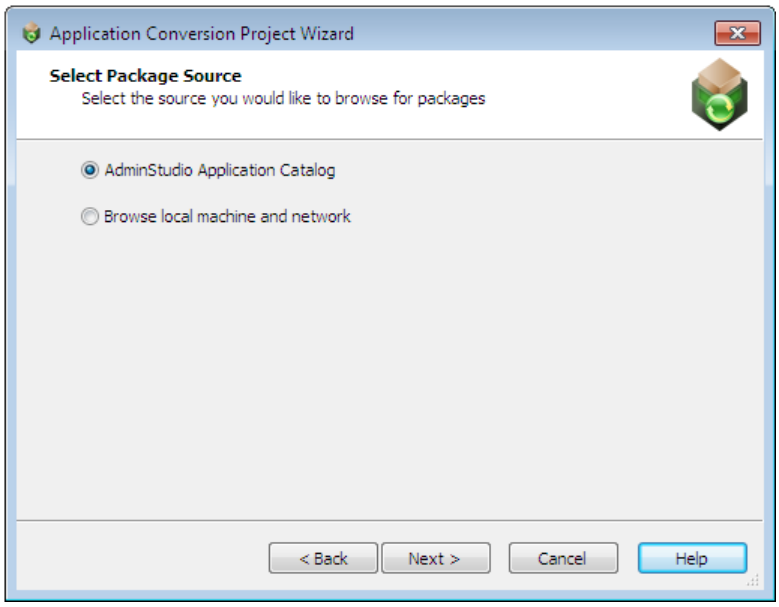


Figure 10-26: Select Package Source Panel

Tip • *To select packages from Microsoft Configuration Manager to convert to virtual applications, first import those packages into the Application Catalog, as described in Importing From ConfigMgr (Formerly called as System Center Configuration Manager).*

The **Select Package Source** panel includes the following options:

Table 10-35 • Select Package Source Panel

Option	Description
AdminStudio Application Catalog	Select this option to connect to an AdminStudio Application Catalog and add all of the installations in that catalog to the list of packages to convert. If you select this option, the Connect to an AdminStudio Application Catalog panel opens, prompting you to login to an Application Catalog.
Browse local machine and network	Select this option to browse a local or network machine to add installations to the list of packages to convert. If you select this option, the Selected Package List panel opens, where you are prompted to select an installation file or a directory of installation files to add to the list of packages to convert.

Connect to an AdminStudio Application Catalog

On the **Connect to an AdminStudio Application Catalog** panel, which opens if you select **AdminStudio Application Catalog** on the **Select Package Source** panel, you enter connection information to connect to an AdminStudio Application Catalog SQL database.

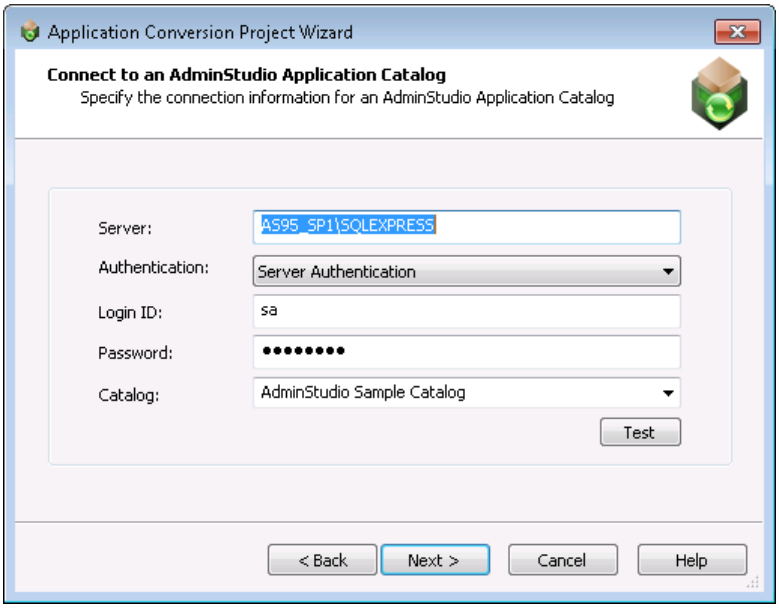


Figure 10-27: Connect to an AdminStudio Application Catalog Panel

On the **Connect to an AdminStudio Application Catalog** panel, enter the following information:

Table 10-36 • Connect to an AdminStudio Application Catalog Panel

Option	Description
Server	Enter the name of the SQL Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use SQL Server login identification to log into this Application Catalog. Then enter the appropriate **Login ID** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Application Catalog. ***Note*** *• After you successfully connect to an Application Catalog, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*
Catalog	Enter the name of the existing AdminStudio Application Catalog database that you want to connect to

Select Packages

The contents of the **Select Packages** panel depends upon the selection you made on the **Select Package Source** panel:

- **Browse local machine and network**—If you selected this option on the **Select Package Source** panel, there are no packages listed on the **Select Packages** panel. You need to click **Browse Folders** or **Browse Files** to select packages

to convert. The **Browse For Folder** or **Select Package Installation File** dialog box would open. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.

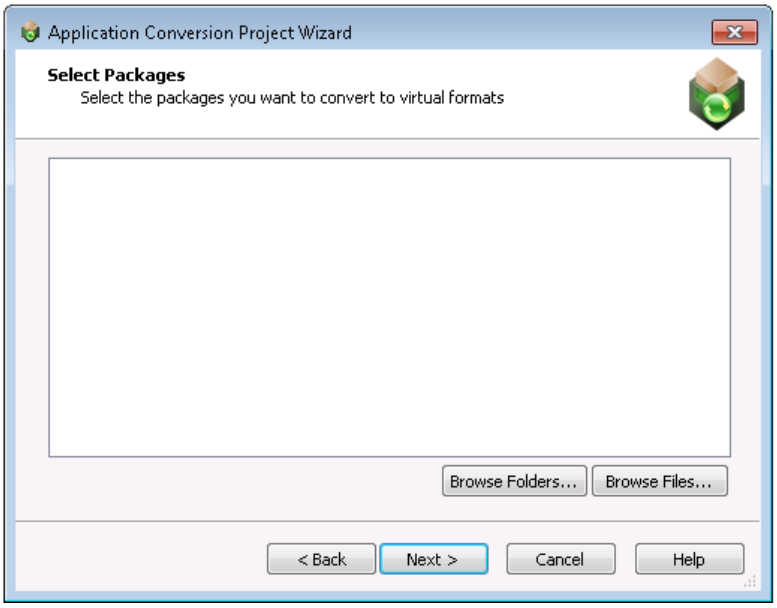


Figure 10-28: Select Packages Panel / No Packages Listed

- **AdminStudio Application Catalog**—If you selected the A**dminStudio Application Catalog** option on the **Select Package Source** panel, the **Select Packages** panel lists all of the packages in the connected package source, in a tree format.

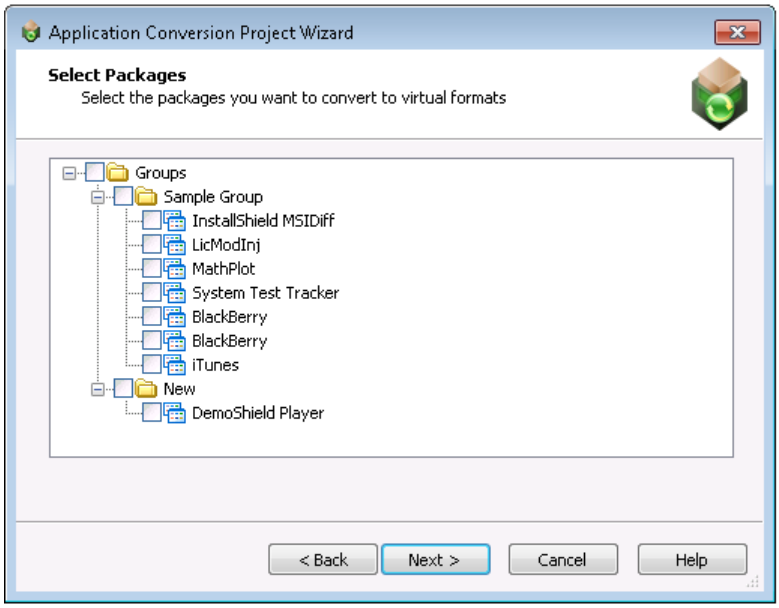


Figure 10-29: Select Packages Panel / Packages Listed

Select the packages that you want to convert and click **Next** to continue.

Selected Package List

The **Selected Package List** panel lists all of the packages you selected on the Select Packages panel.

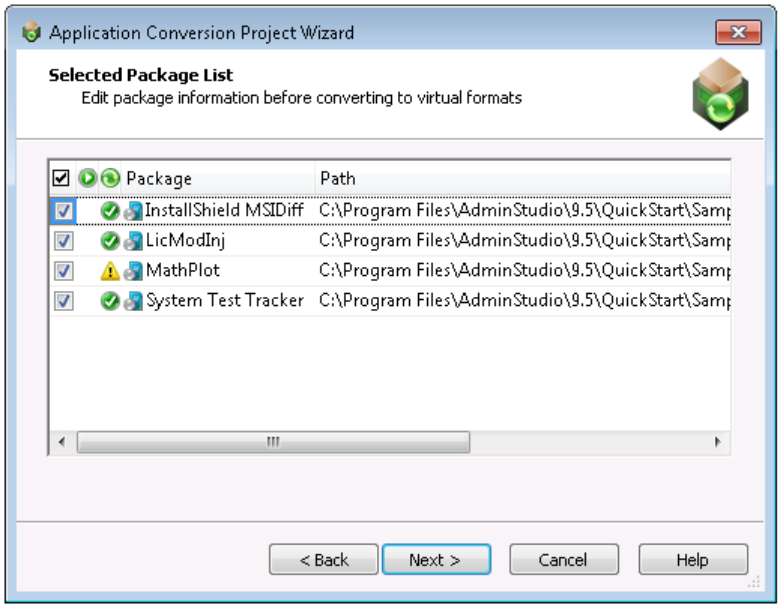


Figure 10-30: Selected Package List Panel

The **Selected Package List** panel includes the following options:

Table 10-37 • Selected Package List Panel

Option	Description
☑	Selection column. To select a package for conversion, click the check box in this column.
Status	Indicates the status of the package. On this panel, no status is indicated, but when this column is shown on the **Packages** tab, status will be indicated by an icon. See Packages Tab for more information.
Transforms	Indicates whether any transforms are associated with the listed Windows Installer package. Automated Application Converter automatically adds all of the `.mst` files located in the same directory as the selected `.msi` file. If a transform is associated with the selected package, one of the following two icons is displayed in this column: • One transform is being added with this package. • Multiple transforms are being added with this package. You may need to specify the order that you want these transforms to be applied. See Packages Tab and MST Dialog Box for more information.

Table 10-37 • Selected Package List Panel (cont.)

Option	Description
Virtualization Readiness	When you add a package to the **Selected Packages List** panel, the Automated Application Converter does a quick check to identify that package's virtualization readiness: whether the package can be virtualized directly or whether it requires repackaging before virtualization. An icon is displayed in this column to indicate the virtualization readiness: You can click on the icon in this column to override the Virtualization readiness status that was automatically been assigned to this package by the Automated Application Converter. The choices are: • **Ready** —Package is ready to virtualize; no repackaging is required. • **Requires repackaging** —Package must be repackaged before it can be successfully virtualized. • **Virtualization not supported** —Automated Application Converter has determined that virtualization is not supported. • **Virtualization not recommended** —Automated Application Converter has determined that this package is not recommended for virtualization. • **Unknown** —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.* ***Note*** • *You can click on the icon in this column to override the Virtualization readiness status that has automatically been assigned to this package.*
Package	Lists the name of the Windows Installer file or legacy installation file that you have added to the **Select Packages** panel.

Table 10-37 • Selected Package List Panel (cont.)

Option	Description
Path	Lists the path from where the package was selected locally or from where it was originally imported into the AdminStudio Application Catalog. ***Note*** • *It is recommended that you use UNC path when importing packages into the Application Catalog.* ***Note*** • *If you are adding packages from an AdminStudio Application Catalog installed on a machine other than the machine where the Automated Application Converter is installed, make sure that the package source path listed here is accessible to the Automated Application Converter machine.*
Command Line	Editable field that lists the command line parameters required to run this installation silently. ***Note*** • *For .exe packages the command line value is fetched from the Deployment Data Tab / Programs Subtab.*

For descriptions of additional columns that are viewable on the Selected Packages List panel, see Packages Tab Properties. (The information that is viewable in the various columns on the Selected Package List panel is same information that is viewable on the Packages tab.)

For information on how to view additional columns in the list, see Changing Which List Columns Are Displayed.

Automated Application Converter's Selection Rules When Adding Packages from a Directory

Instead of adding packages from an AdminStudio Application Catalog, you can choose to add a directory of packages from your local machine or network by doing the following:

- On the **Select Package Source** panel, select **Browse local machine and network** and click **Next**. The **Select Packages** panel opens.
- On the **Select Packages** panel, click **Browse Folders** and select a directory that contains multiple Windows Installer files (`.msi`), installation script files (`.vbs`, `.bat`, `.cmd`, `.ps1`), and/or legacy setups (`.exe`).

When adding packages from a directory, it is recommended that you organize the packages you want to convert in one root directory, with each package in its own first level subdirectory, such as:

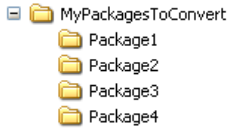


Figure 10-31: Recommended Directory Structure When Adding Packages from a Directory

When you click the **Browse Folders** button and select a folder (such as `MyPackagesToConvert`), the Automated Application Converter scans that folder's first-level subfolders (such as `Package1`, `Package2`, `Package3`, etc.) and uses specific rules to determine which packages it will add to the list on the **Select Packages** panel and which of those packages will be selected:

- **All .msi, .exe, and script files are added to the list**—All `.msi` files, `.exe` files, and script files in the first-level subfolders are added to the list.
- **Only some of the packages are selected**—The Automated Application Converter uses the following rules to determine which of the packages that it adds to the list are selected:
  - `.msi` files are always selected.
  - `.exe` files are only selected if there are no .msi files in that folder.
  - Script files are only selected if there are neither `.msi` files nor `.exe` files in that folder.
- **If a first-level subfolder does not contain any .msi, .exe, or script files, its subfolders are scanned** —If a first-level subfolder does not contain any `.msi`, `.exe`, or script files, the Automated Application Converter will scan its child subfolders to locate package files. However, if a first-level subfolder does contain an `.msi`, `.exe`, or script file, its subfolders are not scanned.

The following table demonstrates these rules:

Table 10-38 • Automated Application Converter's Selection Rules When Adding Packages from a Directory

If the root subdirectory contains...	What are added to the list? Which are selected?	Continue to search subdirectories?
MSI files only	☑ MSIs (added and selected)	No
MSI files and EXE or script files	☑ MSIs (added and selected) ☐ EXEs (added, not selected) ☐ Scripts (added, not selected)	No
EXE files only	☑ EXEs (added and selected)	No
EXE and script files	☑ EXEs (added and selected) ☐ Scripts (added, not selected)	No
Script files only	☑ Scripts (added and selected)	No
None of the above	None	Yes

The following diagram gives a visual representation of these rules in action:

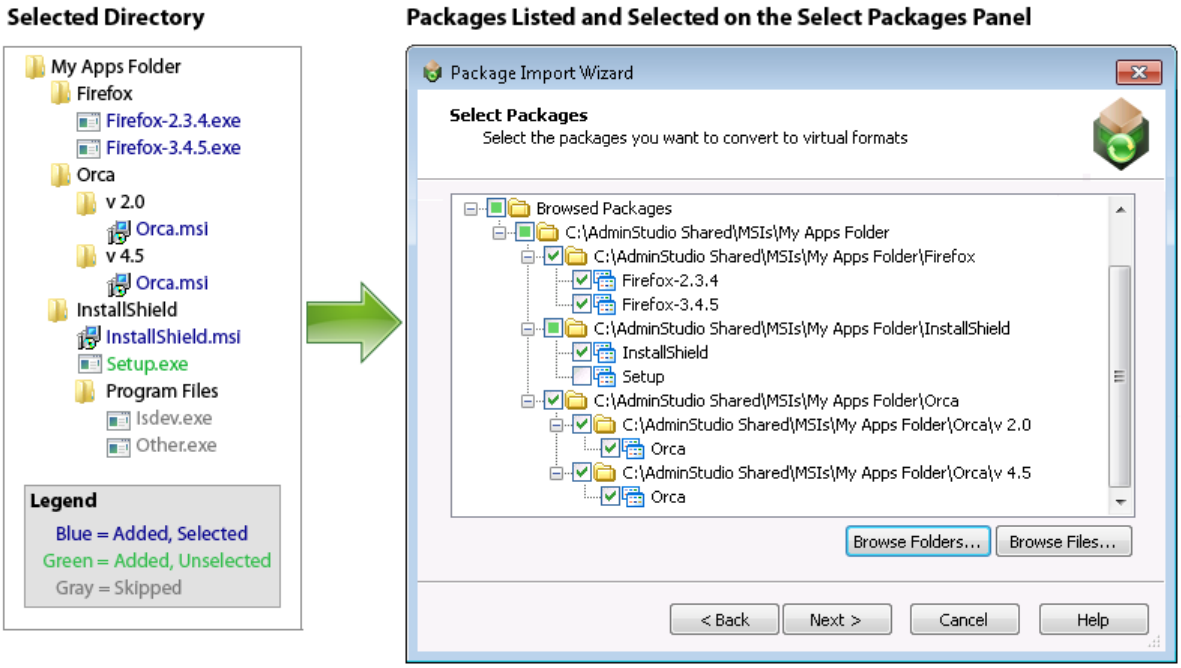


Figure 10-32: Example of How Packages are Added to List When a Directory is Selected

Select Virtual Machine Source

On the **Select Virtual Machine Source** panel, you select the source location of the virtual machines you want to use with the Automated Application Converter to perform automated repackaging.

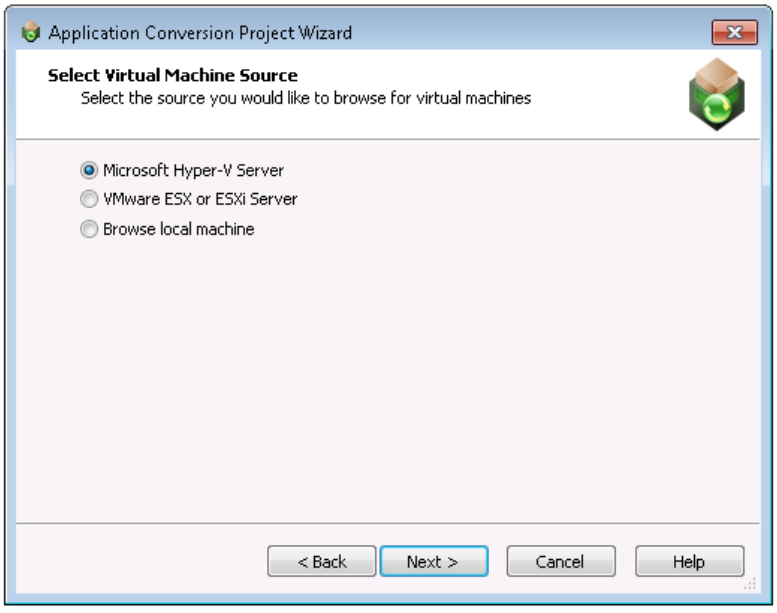


Figure 10-33: Select Virtual Machine Source Panel

Note • *If none of the packages selected on the* ***Selected Package List*** *panel require repackaging in order to be converted into a virtual package, the* ***Select Virtual Machine Source*** *panel will not be displayed. Instead, the* ***Initial Configuration Complete*** *panel will open.*

On the **Select Virtual Machine Source** panel, select one of the following options:

Table 10-39 • Select Virtual Machine Source Panel

Option	Description
Microsoft Hyper-V Server	Select this option to connect to a Microsoft Hyper-V Server. You will then be prompted for login information on the Select Virtual Machines from a Microsoft Hyper-V Server panel.
VMware ESX or ESXi Server	Select this option to connect to a VMware ESX or ESXi Server. You will then be prompted for login information on the Select Virtual Machines from a VMware ESX or ESXi Server panel.
Browse local machine	Select this option to connect to a VMware Workstation virtual image installed locally. The Select Virtual Machines opens, where will be prompted to select either a VMware Workstation image or directory of images.

Select Virtual Machines from a Microsoft Hyper-V Server

On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, you enter a server name and the login information to connect to a Microsoft Hyper-V Server.

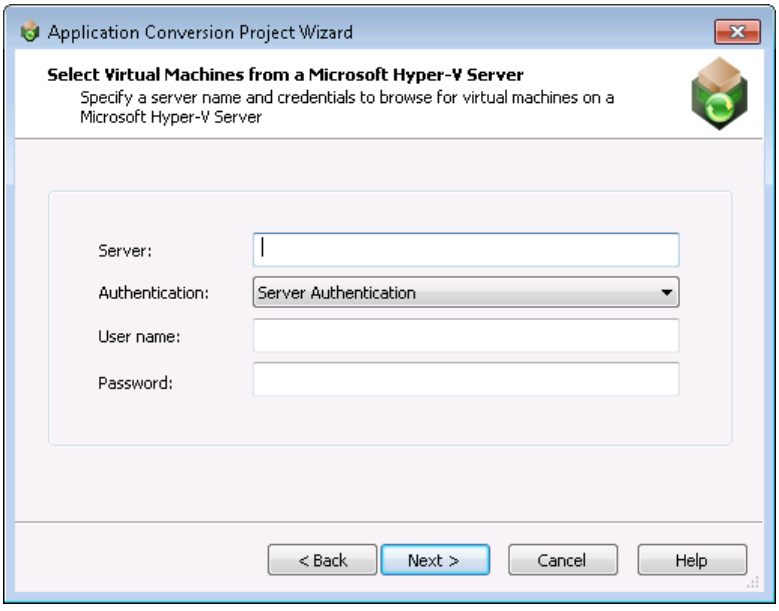


Figure 10-34: Select Virtual Machines from a Microsoft Hyper-V Server Panel

On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, enter the following information:

Table 10-40 • Select Virtual Machines from a Microsoft Hyper-V Server

Option	Description
Server Name	Enter the server name of the Microsoft Hyper-V Server that you want to connect to.
Authentication	Select one of the following options: • **Server Authentication**—Select this option if you want to connect to the Hyper-V Server using a **User name** and **Password** that you specify. • **Windows Authentication**—Select this option to use the credentials of the logged in user to login to the Hyper-V Server.

Select Virtual Machines from a VMware ESX or ESXi Server

On the **Select Virtual Machines from a VMware ESX or ESXi Server** panel, you enter a server name and the login information to connect to a VMware ESX or ESXi Server.

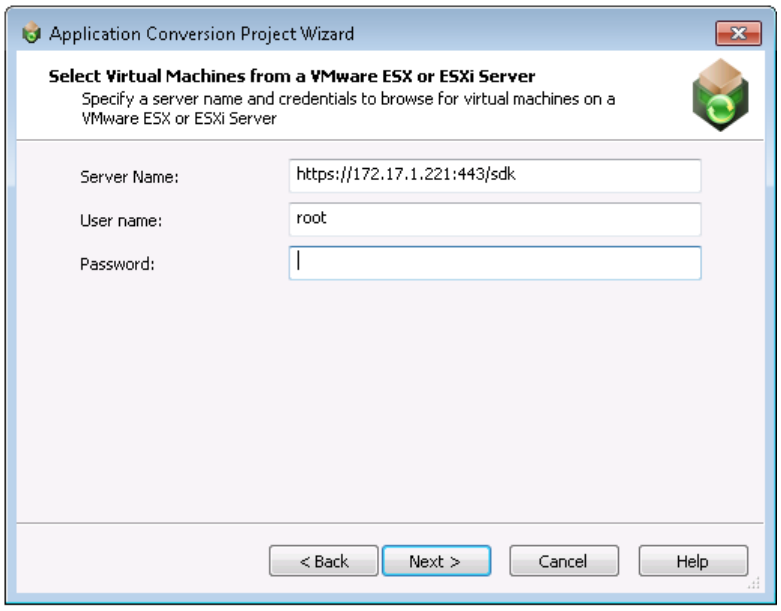


Figure 10-35: Select Virtual Machines from a VMware ESX or ESXi Server Panel

On the **Select Virtual Machines from a VMware ESX or ESXi Server** panel, enter the following information:

Table 10-41 • Select Virtual Machines from a VMware ESX or ESXi Server Panel

Option	Description
Server Name	Enter the name of the VMware ESX or ESXi server you want to connect to.

Table 10-41 • Select Virtual Machines from a VMware ESX or ESXi Server Panel

Option	Description
User name	Enter the login ID for the VMware ESX or ESXi server.
Password	Enter the password for the VMware ESX or ESXi server.

Select Virtual Machines

The contents of the **Select Virtual Machines** panel depends upon the selection you made on the **Select Virtual Machine Source** panel:

- **Browse local machine**—If you selected this option, there are no virtual machines listed on the **Select Virtual Machines** panel. You need to click **Browse Folders** or **Browse Files** to select virtual images. The Browse for Folder Dialog Box or Select Virtual Machine Image File Dialog Box would open.

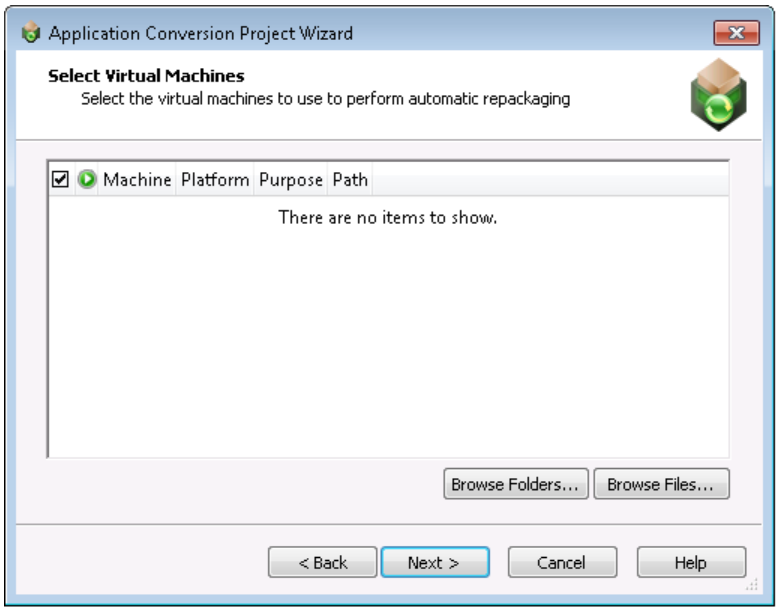


Figure 10-36: Select Virtual Machines Panel / No Machines Listed

- **Microsoft Hyper-V Server** or **VMware ESX or ESXi Server**—If you selected either of these options on the **Select Virtual Machine Source** panel, and have connected to the server, the **Select Virtual Machines** panel lists all of the virtual machines found on the selected server, but does not automatically select all of them.

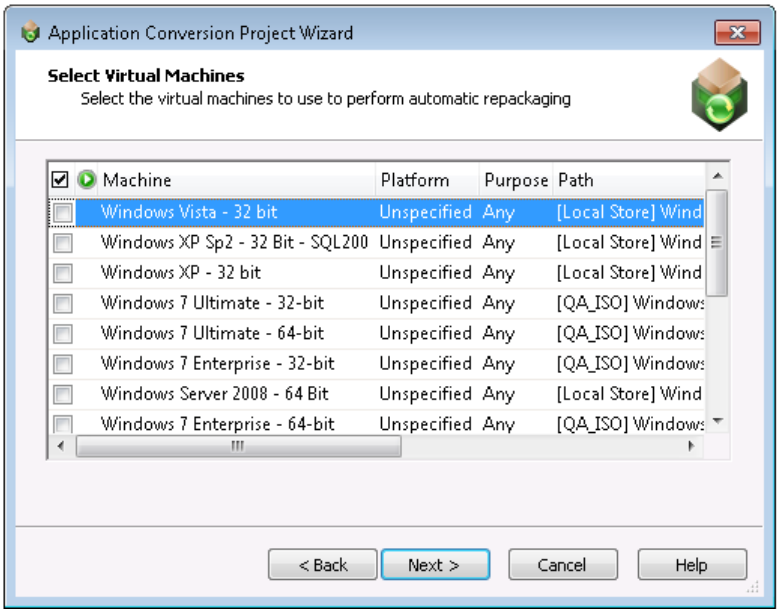


Figure 10-37: Select Virtual Machines Panel / Machines Listed

The **Select Virtual Machines** panel includes the following options:

Table 10-42 • Select Virtual Machines Panel

Option	Description
☑	Selection column. To select a virtual machine to use for automated repackaging, click the check box in this column.
Status	Indicates the status of the virtual machine. On the **Select Virtual Machines** panel, no status is indicated, but when this column is shown on the **Machines** tab, status will be indicated by an icon. See Machines Tab for more information.
Machine	Name of the virtual machine image.
Platform	Identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.
Path	Path to the virtual machine on the virtual machine server or your local machine.

User Credentials

On the **User Credentials** panel, enter the user name and password to use to access the virtual machines you selected on the **Select Virtual Machines** panel.

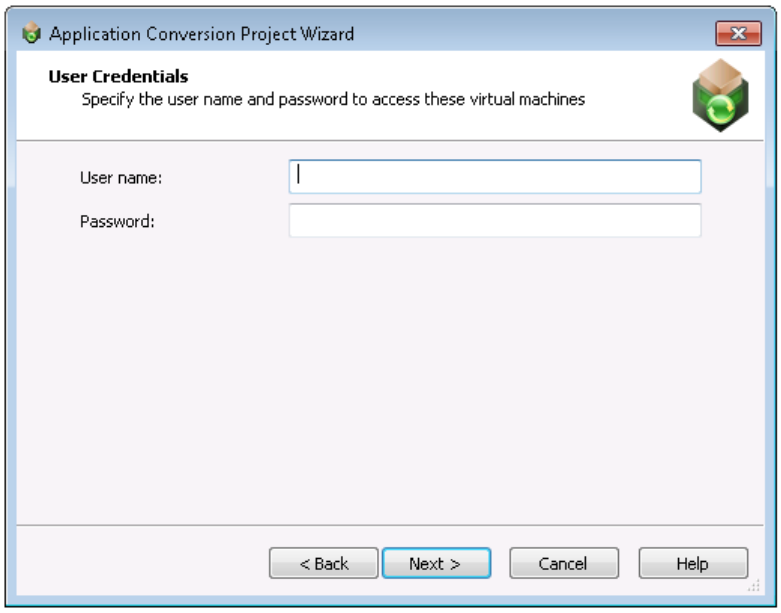


Figure 10-38: User Credentials Panel

The **User Credentials** panel includes the following options:

Table 10-43 • User Credentials Panel

Option	Description
User name	Enter the user name to use to access the virtual machines you selected on the **Select Virtual Machines** panel.
Password	Enter the password to use to access the virtual machines you selected on the **Select Virtual Machines** panel.

Important • *If the virtual machines you selected do not all use the same login credentials, you can add the appropriate credentials in the **Guest Username** and **Guest Password** properties on the Machines Tab after you have added the virtual machine.*

Initial Configuration Complete

The **Initial Configuration Complete** panel lists the packages and machines you have selected to add to your project. You can choose to either begin conversion or to close the wizard so that you can perform additional configuration of these packages and machines prior to conversion.

- **Virtualize packages with detected settings**—Select this option if you want to begin conversion of the selected packages using the selected virtual machines using the current settings.
- **Close wizard to configure packages and machines**—Select this option if you want to close this wizard and perform additional configuration of these packages and virtual machines on the **Packages** and **Machines** tabs prior to beginning conversion.

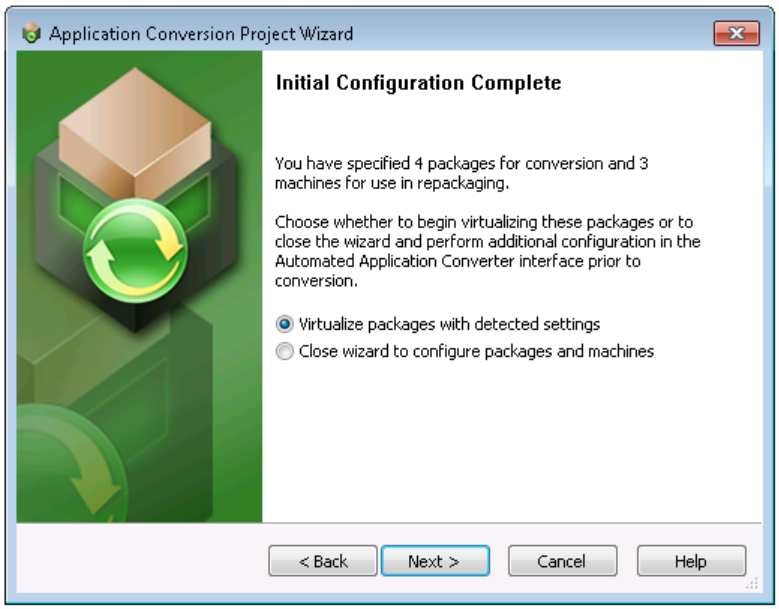


Figure 10-39: Initial Configuration Complete Panel

Select Output Formats

On the **Select Output Formats** panel, select the output formats you want to create and the output location for the packages.

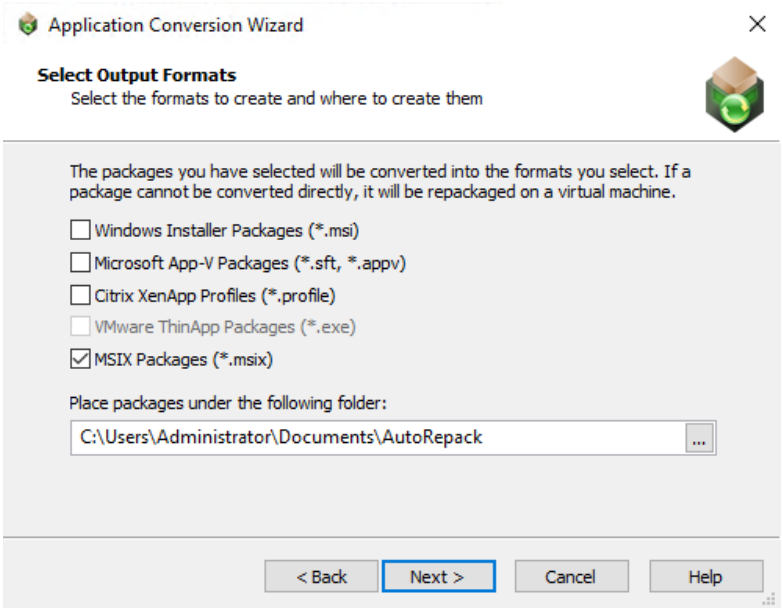


Figure 10-40: Select Output Formats Panel

The **Select Output Formats** panel includes the following options:

Table 10-44 • Select Output Formats Panel

Option	Description
Windows Installer Packages (*.msi)	Select this option to repackage the selected packages into Windows Installer packages (.msi).
Microsoft App-V Packages (*.sft)	Select this option to convert the selected packages to Microsoft App-V packages.
Citrix XenApp Profiles (*.profile)	Select this option to convert the selected packages to Citrix XenApp profiles.
VMware ThinApp Packages (*.exe)	Select this option to convert the selected packages to VMware ThinApp packages.
MSIX Package (.msix)	Select this option to convert the selected packages (MSI & EXE) to MSIX.
Place packages under the following folder	Select the location where you want to store the package output.

Automated Repackaging on Virtual Machines

On the **Automated Repackaging on Virtual Machines** panel, you specify which operating system platform you want to use to perform automated repackaging

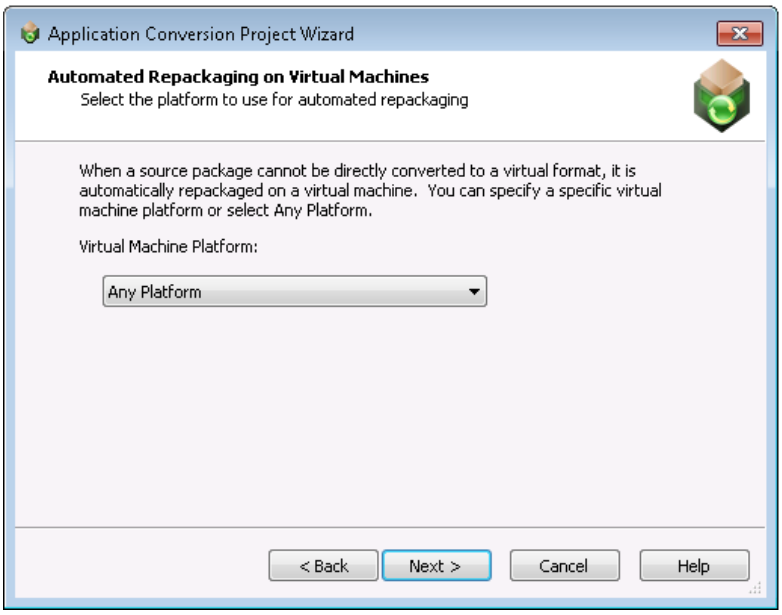


Figure 10-41: Automated Repacking on Virtual Machines Panel

On the **Automated Repackaging on Virtual Machines** panel, select one of the following options:

Table 10-45 •

Option	Description
Any Platform	The Automated Application Converter will use any of the virtual machines that you have selected on the **Machines** tab to perform automated repackaging, regardless of platform.
OS Platform	If you select a specific operating system, the Automated Application Converter will use only those virtual machines that you have selected on the **Machines** tab that are of the selected operating system to perform automated repackaging. ***Note*** *• When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the* ***Select Virtual Machines*** *panel or by clicking in the* ***Platform*** *field on the* ***Machines*** *tab and making a selection from the list.*

Application Conversion Project Wizard Complete Panel

The **Application Conversion Project Wizard Complete** panel lists the virtual formats that your selected packages will be converted to, and the operating system platform of the virtual machine that will be used to perform repackaging if repackaging is required during conversion.

Click **Finish** to close the wizard and begin converting your packages.

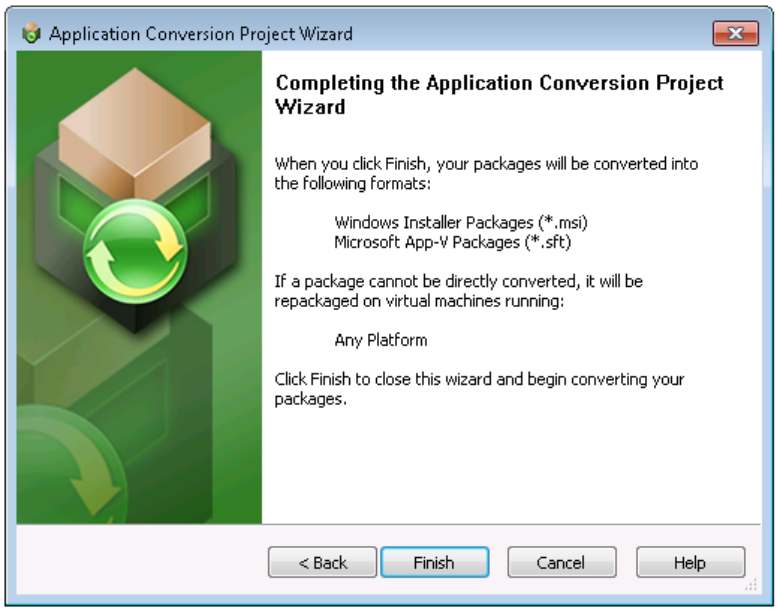


Figure 10-42: Application Conversion Project Wizard Complete Panel

Package Import Wizard

You can use the Package Import Wizard to select packages to convert to virtual formats. You can select packages from a specified AdminStudio Application Catalog or from the file system of a local or network machine.

The Package Import Wizard, which is launched by clicking the **Add Packages** button on the Automated Application Converter **Packages** tab, consists of the following panels:

- Package Import Wizard Welcome
- Select Package Source
- Connect to an AdminStudio Application Catalog
- Select Packages
- Selected Package List
- Package Import Wizard Complete

Note • *The main panels of the Package Import Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Package Import Wizard Welcome

You can use the Package Import Wizard to select packages to convert to virtual formats. You can select packages from a specified AdminStudio Application Catalog or from the file system of a local or network machine.

After you have added the packages, you can use the Virtual Machine Import Wizard to add virtual machines to use for automated repackaging, and then begin a conversion run using the Application Conversion Wizard.

Click **Next** to begin.

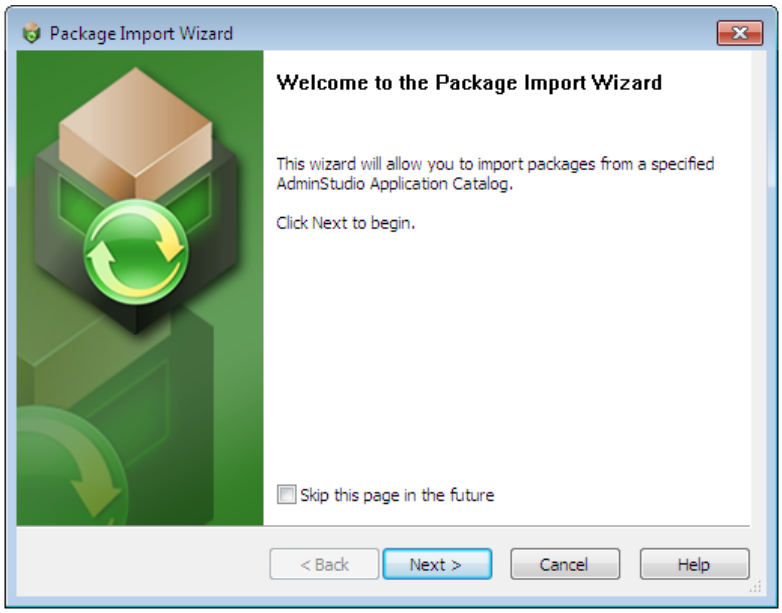


Figure 10-43: Package Import Wizard Welcome Panel

Package Import Wizard Complete

The **Package Import Wizard Complete** panel lists the number of packages you have added to your project for repackaging and virtualization. Click **Finish** to close the wizard and add these packages to your project.

To convert these packages to virtual applications, use the Application Conversion Wizard.

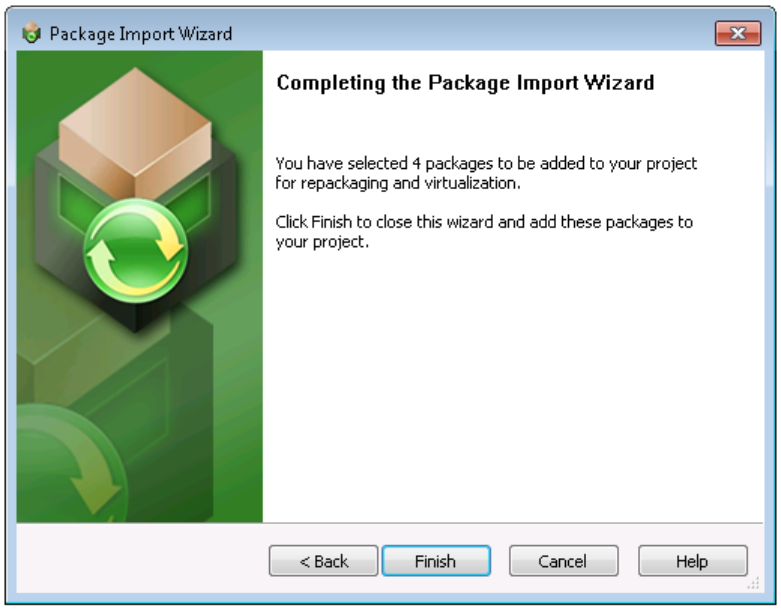


Figure 10-44: Package Import Wizard Complete Panel

Virtual Machine Import Wizard

The **Virtual Machine Import Wizard**, which you use to add virtual machines to the **Machines** tab, consists of the following panels:

- Virtual Machine Import Wizard Welcome
- Select Virtual Machine Source
- Select Virtual Machines from a Microsoft Hyper-V Server
- Select Virtual Machines from a VMware ESX or ESXi Server
- Select Virtual Machines
- User Credentials
- Virtual Machine Import Wizard Complete

Note • *The main panels of the Virtual Machine Import Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Virtual Machine Import Wizard Welcome

You can use the **Virtual Machine Import Wizard** to select virtual machines to use for automated repackaging. You can use virtual machines from a Microsoft Hyper-V Server, a VMware ESX or ESXi Server, or a local VMware Workstation.

After you have added the virtual machines, you can use the Package Import Wizard to add packages to convert, and then begin a conversion run using the Application Conversion Wizard.

Click **Next** to begin.

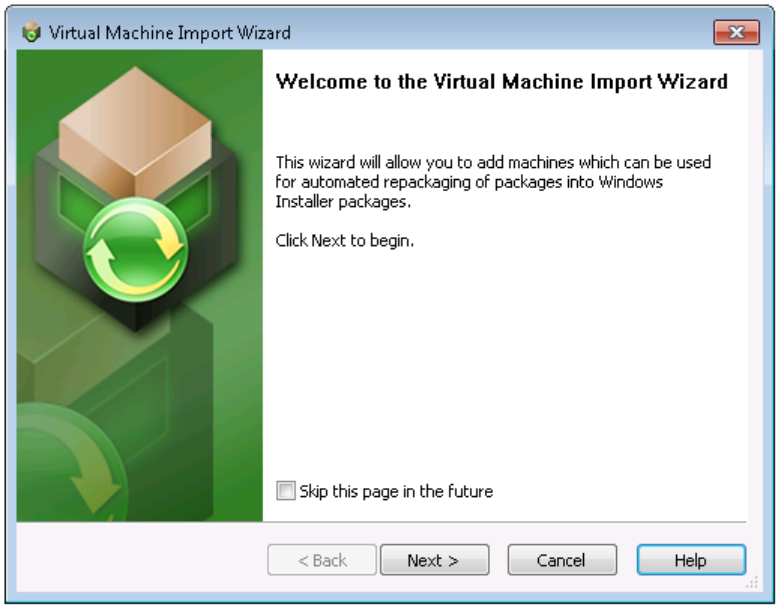


Figure 10-45: Virtual Machine Import Wizard Welcome Panel

Virtual Machine Import Wizard Complete

The **Virtual Machine Import Wizard Complete** panel lists the number of virtual machines you have added to your project for use in automated repackaging.

Click **Finish** to close the wizard and add these virtual machines.

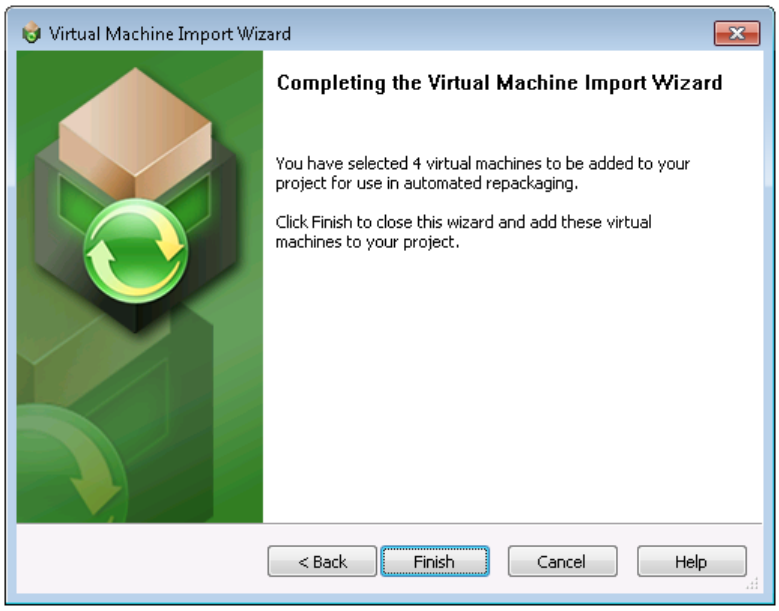


Figure 10-46: Virtual Machine Import Wizard Complete Panel

Application Conversion Wizard

The Application Conversion Wizard consists of the following panels:

- Application Conversion Wizard Welcome
- Select Output Formats
- Automated Repackaging on Virtual Machines
- Application Conversion Wizard Complete

Note • *The main panels of the Application Conversion Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Note • *Automated Application Converter (AAC) now support Windows Services for MSIX Packages. A Windows Service installed while converting a legacy package format (MSI/EXE) will be captured and packaged into the MSIX package upon conversion.*

Application Conversion Wizard Welcome

You can use the **Application Conversion Wizard** to convert selected packages to virtual applications after you have used the Virtual Machine Import Wizard to add virtual machines to the project to use for automated repackaging and used the Package Import Wizard to add the packages you want to convert.

Click **Next** to begin.

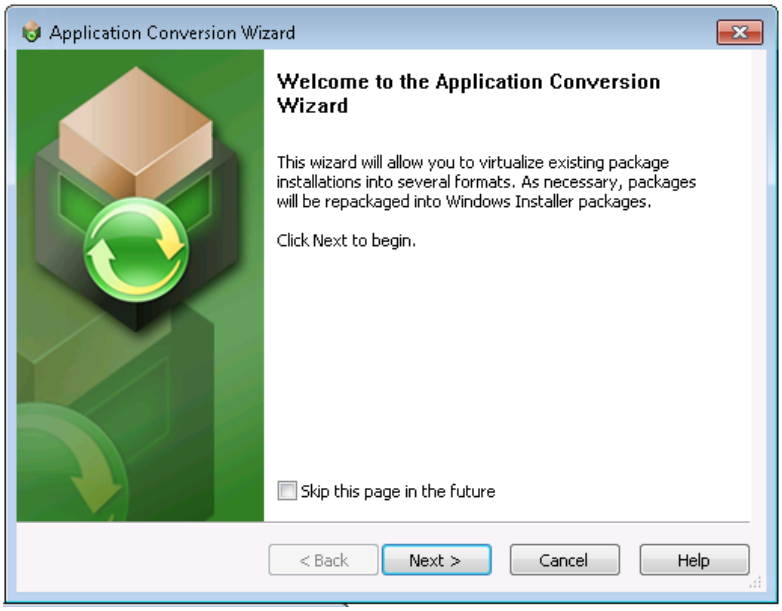


Figure 10-47: Application Conversion Wizard Welcome Panel

Application Conversion Wizard Complete

The **Application Conversion Wizard Complete** panel lists the virtual formats that your selected packages will be converted to, and the operating system platform of the virtual machine that will be used to perform repackaging if repackaging is required during conversion.

Click **Finish** to close the wizard and begin converting your packages.

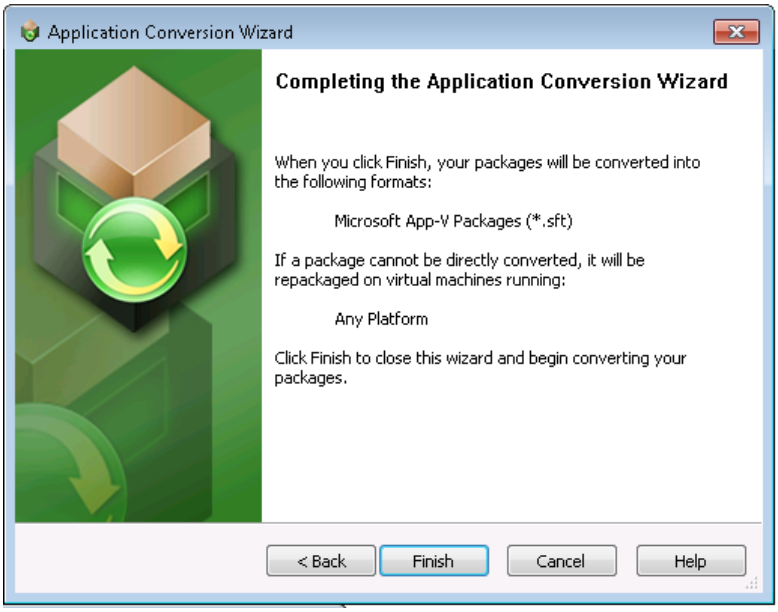


Figure 10-48: Application Conversion Wizard Complete Panel

Dialog Boxes

The Automated Application Converter provides the following dialog boxes:

- Browse for Folder Dialog Box
- Guest Agent
- Open Dialog Box
- Project Options Dialog Box
- Select Package Installation File Dialog Box
- Select Transform Dialog Box
- Select Virtual Machine Dialog Box
- Select Virtual Machine Image File Dialog Box

About Automated Application Converter

The About AdminStudio dialog box can be opened by selecting About AdminStudio from the Help menu. This dialog box displays information about the product, including the full version number (essential if you need technical support). If you have not registered AdminStudio, click the Register button to connect to the InstallShield website to begin the Product Registration process. Registering your product offers you expert technical support, new product announcements and special offers, plus notification of product upgrades.

App-V 5.x Application Launcher

You can use the **App-V 5.x Application Launcher** to launch App-V 5.x packages for testing.

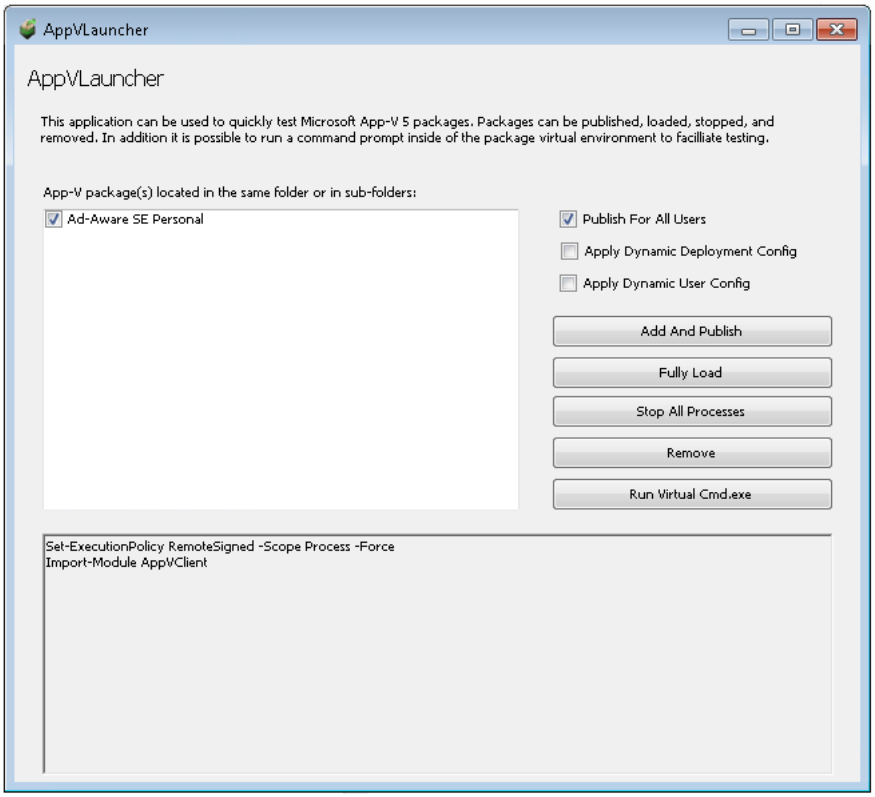


Figure 10-49: App-V 5.x Application Launcher

You open the App-V 5.x Application Launcher by double-clicking on the `AppVLauncher.exe` file located in the same directory as an App-V 5.x package. See Performing Manual Testing of an App-V 5.x Package.

The **App-V 5.x Launcher** includes the following options and buttons:

Table 10-46 • App-V 5.x Launcher Options and Buttons

Option	Description
App-V package list	List of all of the App-V packages located either in the same directory as the `AppVLauncher.exe` file or in a subdirectory of that directory. Select the App-V package(s) that you want to test.
Publish For All Users	Select this option to make the package available to all users on the system. If this option is not selected, then only the currently logged in user will have access to it.
Apply Dynamic Deployment Config	Select this option if you have made customizations to the deployment configuration file that is located next to the App-V package. Changes in this file apply to all users and therefore the **Publish For All Users** option should be used in conjunction.
Apply Dynamic User Config	Select this option if you have made customizations to the deployment configuration file that is located next to the App-V package. Changes in this file apply to all users and therefore the **Publish For All Users** option should be used in conjunction.

Table 10-46 • App-V 5.x Launcher Options and Buttons

Option	Description
Add And Publish	Click to publish this App-V package to the App-V Client so that it can be tested. After this is done, all of the entry points into the package will be published. This includes shortcuts and file type extensions, among others.
Fully Load	Click to load the entire package content into the App-V client. If this is not done, and the package has not been designed to fully load automatically, then the package files are loaded on demand.
Stop All Processes	Click to terminate any running processes originating from the selected App-V package(s). All processes must be stopped in order to perform some operations such as removal.
Remove	Click to un-publish the App-V package.
Run Virtual Cmd.exe	Click to open a command window within the virtual environment. This can be used to run other commands from within the virtual environment for testing purposes.
Output window	List of the informational, progress, and error messages. This includes the PowerShell commands run to perform the selected actions.

Browse for Folder Dialog Box

On the **Browse for Folder** dialog box, select the directory containing the packages to convert.

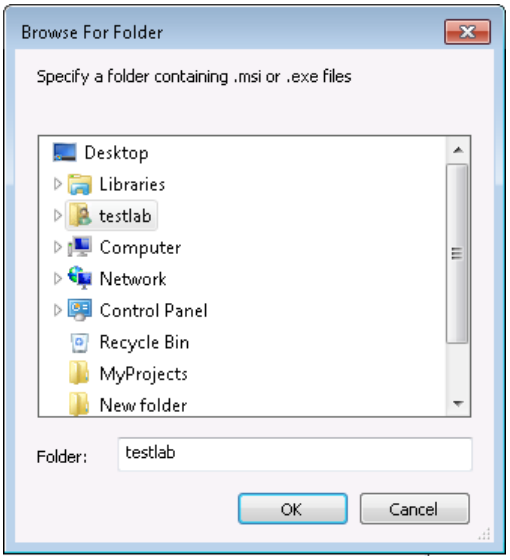


Figure 10-50: Browse for Folder Dialog Box

Select the directory that contains the Windows Installer files (`.msi`) and/or legacy package files (`.exe`) you want to convert and click **Open**. the Automated Application Converter searches the selected directory and its subdirectories to locate `.msi` and `.exe` files and adds them to the list on the **Select Packages** panel.

Important • *The Automated Application Converter uses specific rules to determine which packages in the selected directory and its subdirectories are added to the list on the* ***Select Packages*** *panel, and which of those files are automatically selected. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.*

Guest Agent

The Guest Agent (`GuestAgent.exe`) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the AdminStudio Repackager in an automated fashion.

Figure 10-51: Guest Agent Interface

Open Dialog Box

The **Open** dialog box opens when you select **Open** on the **File** menu or when you select the **Open existing project** option on the **Open Project** panel of the Application Conversion Project Wizard and then select **Browse for project file...** from the list.

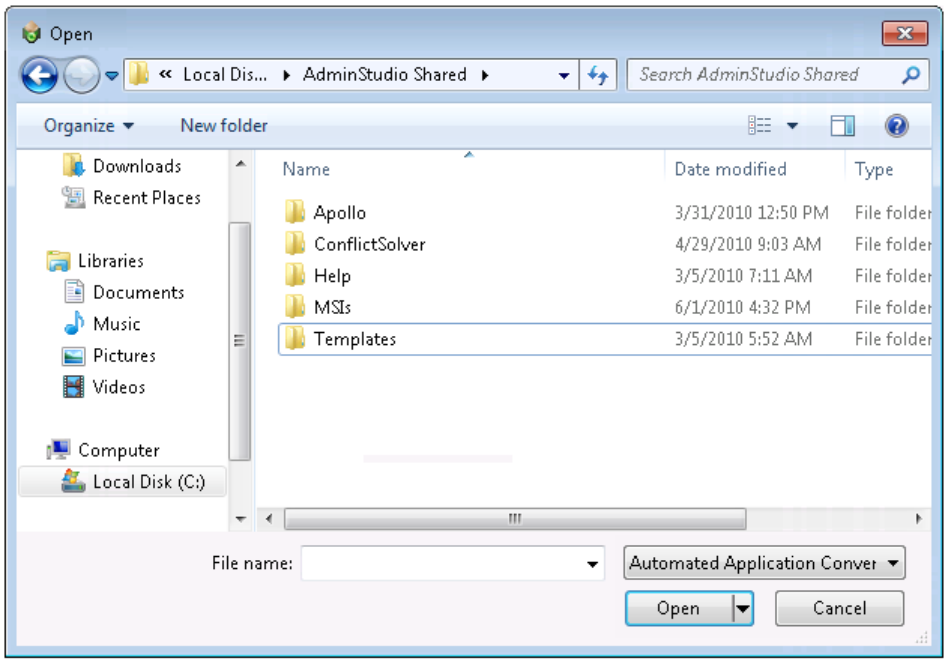


Figure 10-52: Open Dialog Box

Select an Automated Application Converter project file (`*.aacx`) and click **Open** to open the file.

MST Dialog Box

The **MST** dialog box, which opens when you click the Browse ... button in the **Transform** field in the **Properties** window of the **Packages** tab, lists the transforms that are associated with the selected Windows Installer package. Automated Application Converter automatically lists all of the `.mst` files located in the same directory as the selected `.msi` file.

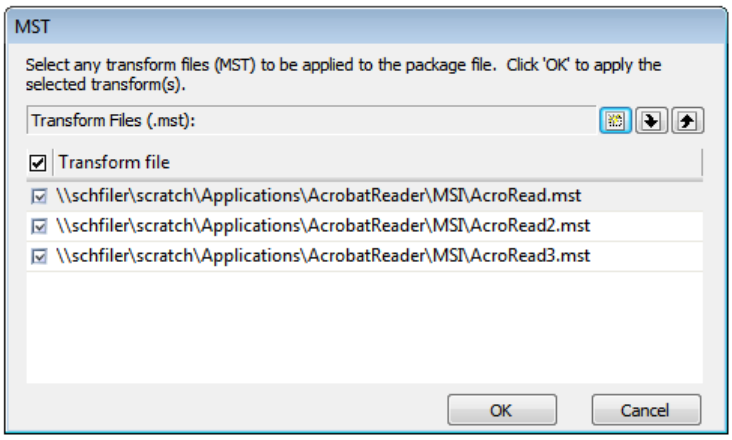


Figure 10-53: MST Dialog Box

Note • *You can also open the **MST** dialog box by clicking the Browse button in the **Transform** column of the package listing on the **Packages** tab, or by clicking the Browse button on the **Transform** column of the **Selected Package List** panel of the Package Import Wizard or the Application Conversion Project Wizard.*

On the **MST** dialog box, specify how transforms should be handled for the selected Windows Installer package:

- **Select transforms**—Select the transform (`.mst`) files that you want to import along with the Windows Installer package. If you do not want to import a selected `.mst` file, clear the selection.
- **Add additional transforms**—To add additional transforms that are not located in the same directory as the selected Windows Installer package, click the New button ( ) and browse to the location of the transform. If the package requires multiple transforms, you can repeat the procedure as necessary.
- **Order transforms**—If more than one transform is listed, use the up and down arrows to order the list of transforms in the order you want them applied.

Project Options Dialog Box

On the **Project Options** dialog box, which is opened by selecting **Options** on the **Tools** menu, you can specify project-wide default options.

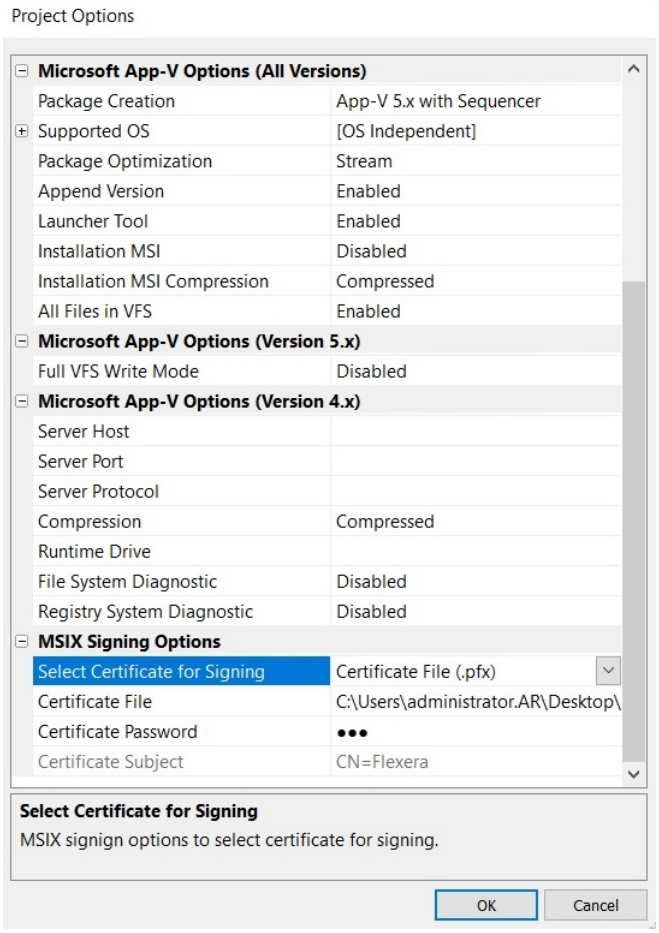


Figure 10-54: Project Options Dialog Box

The **Project Options** dialog box includes properties that are grouped into the following sections:

- Conversion Options
- Microsoft App-V Options (All Versions)
- Microsoft App-V Options (Version 4.x)
- Microsoft App-V Options (Version 5.x)
- MSIX Signing Options

Conversion Options

The **Conversion Options** group on the **Project Options** dialog box includes the following options:

Table 10-47 • Conversion Options Group on the Project Options Dialog Box

Option	Description
Output Path	Specify the default location that will populate the **Place packages under the following folder** field on the **Select Output Formats** panel of the Application Conversion Project Wizard and Application Conversion Wizard.
Build Windows Installer Packages **Build App-V Packages** **Build Citrix Profiles** **Build ThinApp Packages**	If this option is set to **True**, the output format will be selected by default on the **Select Output Formats** panel of the of the Application Conversion Project Wizard and Application Conversion Wizard. If it is set to **False**, the output format will not be selected by default.
Soft Time-Out Notification	Set this option to **True** if you want AdminStudio to automatically send you an email notification when a soft time-out is encountered while using Automated Application Converter to repackage an application on a virtual machine. ***Note*** • *To enable email notification, you need to configure your SMTP notification settings on the* ***Notification Settings*** *tab of the AdminStudio* ***Options*** *dialog box.*
Post-Installation Configuration	Indicate whether you want to enable configuration of the application after it is installed on the virtual machine but before it is converted into the target formats. Available options are: • **Disabled**—Disable post-installation configuration. The repackaging process does not pause after installing the product. This is the default value. • **Enabled**—Enable post-installation configuration. The repackaging process pauses after the installation of the product to allow you to launch the product and set up various application settings such as update settings and file associations. You can also perform other system configuration tasks. Once you are done with configuration, you can click a button to have the repackaging proceed with the capture and convert process. To override this behavior on an individual package basis, use the Post-Installation Configuration field in the Properties window for a specific package. ***Important*** • *If you select the Enabled option, ensure that the value that you enter for the Hard Time-Out setting for each individual package allows enough time to configure the application.*

Microsoft App-V Options (All Versions)

The **Microsoft App-V Options (All Versions)** group on the **Project Options** dialog box includes the following options:

Table 10-48 • Microsoft App-V Options (All Versions) Group on the Project Options Dialog Box

Option	Description
Package Creation	Select one of these options to identify the default App-V version (4.6 or 5.x) and conversion method (AdminStudio or App-V Sequencer) for App-V package conversion: • **App-V 4.6 with AdminStudio**—Convert to App-V 4.6 format using AdminStudio. • **App-V 5.x with AdminStudio**—Convert to App-V 5.0 format using AdminStudio. • **App-V 5.x with Sequencer**—Convert to App-V 5.0 format using the Microsoft App-V 5.0 Sequencer. ***Note*** • *The **App-V 5.x with Sequencer** option requires that the virtual machine have Microsoft Sequencer Version 5.x pre-installed. For more information, see Preparing a Snapshot for App-V 5.0 Conversion Using the App-V 5.0 Sequencer.* ***Note*** • *This setting can be overridden on a per-package basis by changing the **Package Creation** property for a package on the **Packages** tab.*

Table 10-48 • Microsoft App-V Options (All Versions) Group on the Project Options Dialog Box

Option	Description
Supported OS	Use the **Supported OS** setting and its subsettings to specify the default supported operating systems for App-V packages generated by Automated Application Converter: • **To specify that App-V packages are operating-system-dependent** (meaning that the App-V packages will only support some of the listed operating systems), select **True** next to the supported operating systems. If any of the listed operating systems are set to **True**, the value for **OS Independent** will automatically switch to **False**, and the selected operating systems will be listed in brackets next to **Supported OS**. • **To specify that App-V packages are operating-system-independent** (meaning that the App-V packages will support all listed operating systems), set **OS Independent** to **True**. When you make this selection, all operating systems will automatically switch to **False**. ***Important*** • *You can override these default settings for an individual App-V package by setting the **Supported OS** property on that package's **Packages** tab.* ***Important*** • *When setting the **Supported OS** property for App-V 5.0 packages, keep in mind that the packages are limited to the supported operating systems of the App-V 5.0 client:* • *Windows 7 and later* • *Windows Server 2008 R2 and later*

Table 10-48 • Microsoft App-V Options (All Versions) Group on the Project Options Dialog Box

Option	Description
Package Optimization	Specify how to optimize the package: • **Offline**—When the package is optimized for offline use, the entire package is included in feature block 1 and will be streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable users to use the App-V package when not connected to the App-V server and if you want to eliminate network traffic when the App-V package is being used. • **Stream**—When the package is optimized for streaming use, only the shortcut targets which are included in feature block 1 are streamed to the client at start up. Feature block 2 can contain additional functionality of the App-V package that is not necessary to launch the application. While the App-V package is being used, the files in feature block 2 are streamed in small packets on an as-needed basis. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** *• When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*
Append Version	Specify the default App-V versioning value. Available options are: • **Enabled**—Append the package version to the SFT file name. • **Disabled**—Leave the package version off of the SFT file name.
Launcher Tool	Set this option to **Enabled** to include the App-V Application Launcher when you build an App-V package. You can use the App-V Application Launcher to test a newly built App-V package before moving it to a deployment server. ***Note*** *• The default value for the* ***Launcher Tool*** *option is* ***Enabled****.* ***Note*** *• You can override this default setting for an individual package by setting the* ***Launcher Tool*** *property on the package's* ***Properties*** *window of the* ***Packages*** *tab.*
Installation MSI	Specify whether to create a Windows Installer package wrapper to install the App-V package: • **Enabled**—Create a Windows Installer package wrapper to install the App-V package. • **Disabled**—Do not create a wrapper.

Table 10-48 • Microsoft App-V Options (All Versions) Group on the Project Options Dialog Box

Option	Description
Installation MSI Compression	Specify whether to compress the Windows Installer package wrapper. • **Compressed**—Compress the Windows Installer package wrapper. • **Uncompressed**—Do not compress the wrapper. ***Note*** *• This option is not available when the* ***Package Conversion*** *option is set to* ***App-V 5.x with Sequencer****. In this case, the Windows Installer package wrapper is always uncompressed.*

Microsoft App-V Options (Version 4.x)

The **Microsoft App-V Options (Version 4.x)** group on the **Project Options** dialog box includes the following options:

Table 10-49 • Microsoft App-V Options (Version 4.x) Group on the Project Options Dialog Box

Option	Description
Server Host	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter **%SFT_SOFTGRIDSERVER%** to indicate an environment variable. ***Note*** *• If you enter* **%SFT_SOFTGRIDSERVER%***, you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Server Port	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.

Table 10-49 • Microsoft App-V Options (Version 4.x) Group on the Project Options Dialog Box

Option	Description
Server Protocol	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: • **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. • **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. • **FILE**—The App-V package are streamed from a file share. • **HTTP**—The hypertext transport protocol streams the App-V package. • **HTTPS**—The secure hypertext transport protocol streams the App-V package.
Compression	Specify whether to compress the App-V package: • **Compressed**—Compress the App-V package. • **Uncompressed**—Do not compress the App-V package.
Runtime Drive	Specify the App-V client runtime drive. If no value is set, the default value of `Q:\` will be used.
File System Diagnostic	Set this property to **Enabled** if you want to include the Windows Command Prompt application when you build an App-V packages so that you can browse the virtual file system at runtime from within the virtual environment. If this property is set to **Enabled**, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment. You can use `Virtual File System.osd` to view the existing files and folders on the computer plus the files and folders for the virtual package. ***Note*** *• The default value for the* ***File System Diagnostic*** *option is* ***Disabled****.* ***Note*** *• You can override this default setting for an individual package by setting the* ***File System Diagnostic*** *property on the package's* ***Properties*** *window of the* ***Packages*** *tab.*

Table 10-49 • Microsoft App-V Options (Version 4.x) Group on the Project Options Dialog Box

Option	Description
Registry System Diagnostic	Set this property to **Enabled** if you want to include the Registry Editor (`regedit.exe`) when you build an App-V package so that you can browse the registry at runtime from within the virtual environment. If this property is set to **Enabled**, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment. You can use `Virtual Registry.osd` to view the existing registry on the computer plus the registry for the virtual package. ***Note*** *• The default value for the* ***Registry System Diagnostic*** *option is* ***Disabled****.* ***Note*** *• You can override this default setting for an individual package by setting the* ***Registry System Diagnostic*** *property on the package's* ***Properties*** *window of the* ***Packages*** *tab.*

Microsoft App-V Options (Version 5.x)

The **Microsoft App-V Options (Version 5.x)** category includes the following properties:

Table 10-50 • Packages Tab / Microsoft App-V Options (Version 5.x) Category of Properties Window

Property	Description
Expand App-V Package	Use this option to expand an existing App-V 5.x package on the system before performing the sequencing. This is useful for specifying middleware and dependency App-V packages such as Java runtime. Click the browse button and select the App-V package to expand.
Named Objects Interaction	Specify **Enabled** to enable all named objects to interact with the local system. The default setting is **Disabled** to keep these objects isolated from the local system. Named objects include application's events and mutexes among other things. This is an advanced setting that typically does not need to be changed from the default. It is only compatible with the **App-V 5.x with AdminStudio** approach for package creation.
COM Objects Interaction	Specify **Enabled** to allow programs on the local system to interact with all COM objects present in the virtual package. The default setting is **Disabled** in order to isolate all COM objects of the virtual package. This is an advanced setting that typically does not need to be changed from the default. It is only compatible with the **App-V 5.x with AdminStudio** approach for package creation.

Table 10-50 • Packages Tab / Microsoft App-V Options (Version 5.x) Category of Properties Window

Property	Description
Full VFS Write Mode	Set this option to **Enabled** to give the virtual application full write permissions to its VFS (virtual file system) files and folders. ***Note*** • *The **Full VFS Write Mode** feature was introduced in App-V 5.0 SP2 HotFix 4.*

MSIX Signing Options

The **MSIX Signing Options** category includes the following properties:

Table 10-51 • Packages Tab / MSIX Signing Options Category of Properties Window

Property	Description
Select Certificate for Signing	Use to specify the signing option: • **Certificate File (.pfx)** Specify a path and password to a valid PFX certificate file. • **Certificate Store** Specify the following certificate store details: ***Note*** • *If you select the Certificate Store option, make sure that the Certificate has been imported. For more details on importing the certificate, see Import Certificate.* • **Certificate Store Location** Select either User or Machine. • **Certificate Store Name** Select the any one of the store name from the drop-down list. • **Certificate Subject** Specify the subject of the certificate. ***Note*** • *Certificate Subject populates with respect to the selected certificate.*

Select Package Installation File Dialog Box

On the **Select Package Installation File** dialog box, select the installation file (`.msi or .exe`) or installation script (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) that you want to add to your project for conversion to a virtual application.

Note • *You can use installation scripts to run more complex installation scenarios.*

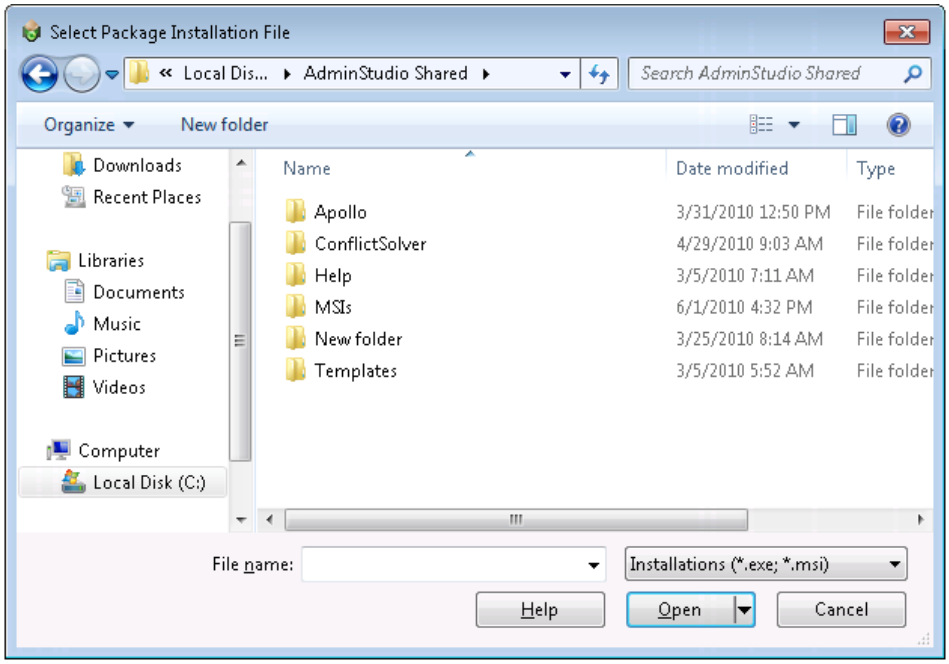


Figure 10-55: Select Package Installation File Dialog Box

Select Transform Dialog Box

On the **Select Transform** dialog box, which opens when you click in the **Transform** column/property on the **Packages** tab, you can select a transform file (`.mst`) to modify or install a Windows Installer package silently.

Note • *While the **Transform** property on the **Packages** tab can contain a semicolon-delimited list of transforms, when you browse to the transform file location using the **Select Transform** dialog box, you are only able to select one transform file. To include multiple transforms with a package, rather than browsing to the transform file location, you need to manually edit the **Transform** property on the **Packages** tab to enter multiple transform files, separated by a semicolon.*

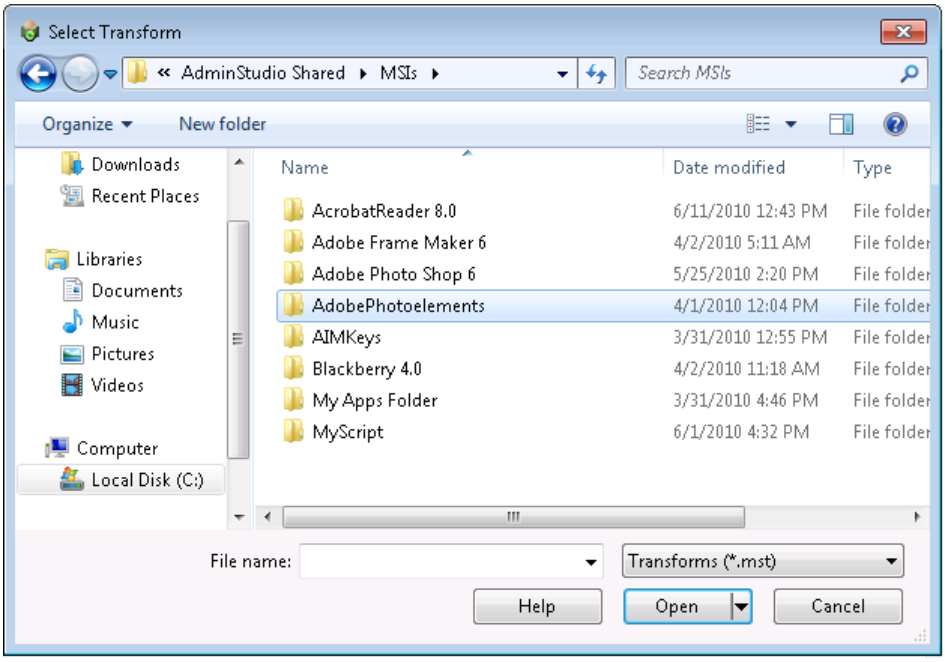


Figure 10-56: Select Transform Dialog Box

Select Virtual Machine Dialog Box

On the **Select Virtual Machine** dialog box, which opens when you right-click on a package on the **Packages** tab and then select **Launch Package for Testing** from the shortcut menu, you select the virtual machine that you want to use to test the selected package.

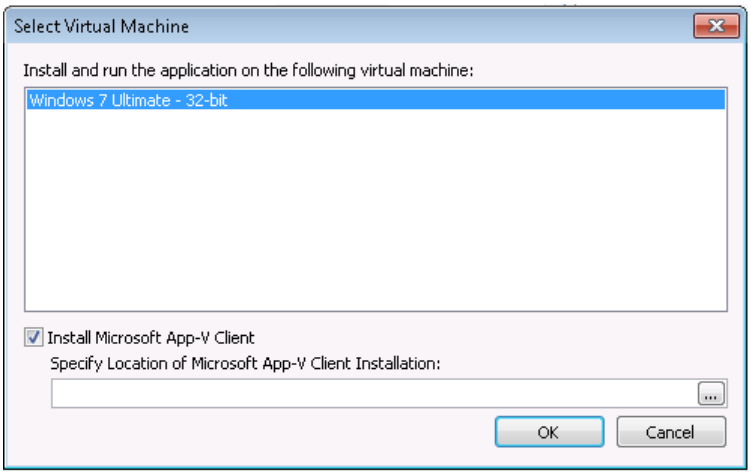


Figure 10-57: Select Virtual Machine Dialog Box

The following options are available:

Table 10-52 • Select Virtual Machine Dialog Box

Option	Description
Install and run the application on the following virtual machine	Select the virtual machine that you would like to use the test the selected package.
Install Microsoft App-V Client	If you are testing an App-V package, select this option to instruct the Automated Application Converter to install the App-V client on the selected virtual machine.
Specify Location of Microsoft App-V Client Installation	If you have selected the **Install Microsoft App-V Client** option, specify the location of the App-V client installation. Make sure that Automated Application Converter machine has access to the specified location.

Select Virtual Machine Image File Dialog Box

When you click **Browse Files** on the **Select Virtual Machines** panel, the **Select Virtual Machine Image File** dialog box opens, prompting you to select a VMware Workstation virtual machine image.

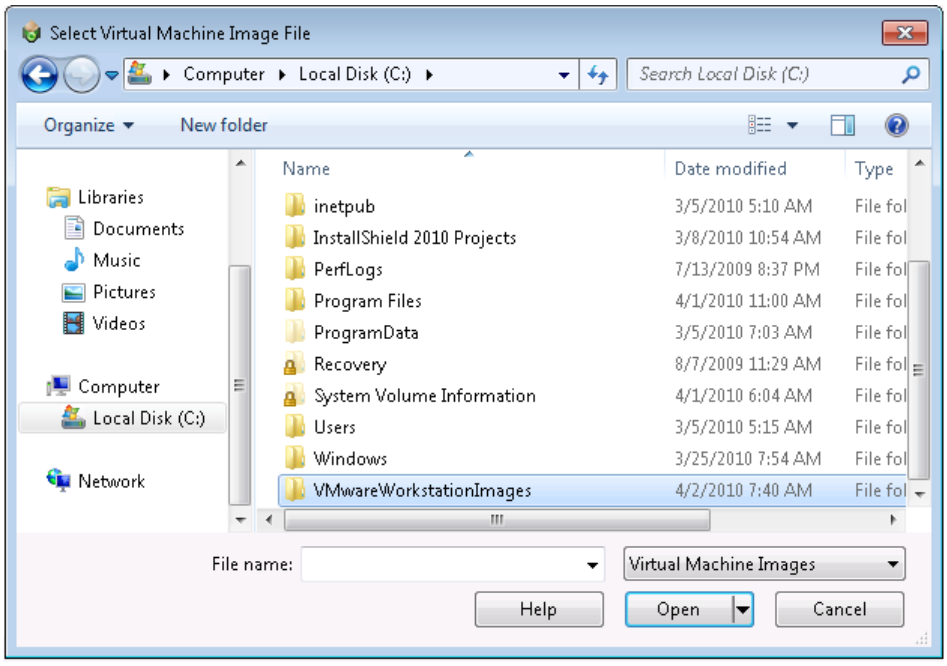


Figure 10-58: Select Virtual Machine Image File Dialog Box

Command Line Support

You can choose to run an Automated Application Converter project file via command line using the following command:

```
aacx.exe projectname.aacx
```

where `projectname.aacx` is the project file to load and execute. Results are displayed in console mode.

You can also use the following command line parameters to override project file settings:

Table 10-53 • Command Line Parameters

Parameter	Description

/create	Use the `/create` parameter to indicate the virtual formats to create, such as:
	`aacx.exe /create formattype projectname.aacx`
	where *formattype* can be any of the following (case-insensitive):
	`AppV`
	`Citrix`
	`ThinApp`
	`MSI`
	`xpf`
	and *projectname.aacx* is the project file to load and execute.
	For example, to create an App-V package, enter:
	`aacx.exe /create AppV myproject.aacx`
	Multiple create commands can be specified in the same command line. For example, to create all virtual formats, enter:
	`aacx.exe /create AppV /create Citrix /create ThinApp myproject.aacx`
	
	Note • *Settings made using the* `/create` *parameter override the selections you made on the* ***Select Output Formats*** *panel, which are saved in the project file.*

/help
/?
	To view command line help, enter either of the following:
	`aacx.exe /?`
	`aacx.exe /help`

/log	To create a unicode text file to contain output messages, enter the `/log` parameter followed by an output file name: `aacx.exe /log output.txt myproject.aacx` ***Note*** • *These are the same output messages that would appear in the Output window when using the Automated Application Converter interface,*

Table 10-53 • Command Line Parameters (cont.)

Parameter	Description
/options	To specify an alternate `options.ini` file for repackaging with Automated Application Converter, enter the `/options` parameter followed by the path to the `options.ini` file that you want to use, such as: `/options C:\options.ini` Using this option enables you to specify a different `options.ini` file when repackaging with Automated Application Converter than the `options.ini` file that you use when performing standard repackaging with Repackager (which could have custom options in it). The specified `options.ini` file will be copied to the guest image during repackaging and will overwrite the default Repackager `options.ini` file.
/outdir	To override the output directory for built and converted packages that was set in the project file on the **Select Output Formats** wizard panel, use the `/outdir` parameter: `aacx.exe /outdir "C:\output\aacxoutput" myproject.aacx` where `C:\output\aacxoutput` is the name of the directory that will contain the output.
/report	To specify the name of the HTML report that is generated after conversion, use the `/report` parameter: `aacx.exe /report reportname.html myproject.aacx`
/showreport	To specify the name of the HTML report that is generated after conversion and to automatically display that report, use the `/showreport` parameter: `aacx.exe /showreport reportname.html myproject.aacx`
/vmplatform	To specify the platform to use when performing automated repackaging, overriding the VMs selected in the project file, use the `/vmplatform` parameter: `aacx.exe /vmplatform platformvalue proj.aacx` where *platformvalue* is constructed from a version integer using the formula of `MajorVersion * 100 + MinorVersion` of the operating system (such as **600** for Windows Vista), followed optionally by **s** (for server) and/or **x64** (for 64-bit). Examples are below.

Windows Vista 32-bit	`aacx.exe /vmplatform 600 myproject.aacx`
Windows Vista 64-bit	`aacx.exe /vmplatform 600x64 myproject.aacx`
Windows Server 2008 R2 64-bit	`aacx.exe /vmplatform 601sx64 myproject.aacx`
All enabled machines	`aacx.exe /vmplatform any myproject.aacx`

Note *• The version integer described above is similar to the Windows Installer VersionNT property. See Operating System Property Values on the MSDN website.*

Specifying Global Default Virtual Conversion Settings

In addition to the settings that can be specified on the Project Options Dialog Box, a default value can be specified for any virtual conversion setting that would normally be stored in the `ISVirtualPackage` table by editing the `settings.xml` file. The global value is used if no project-specific value is found.

To configure these global default values, locate the `settings.xml` file installed with InstallShield Editor and AdminStudio Repackager, and then find the `<Properties>` subelement of the `<Virtualization>` element:

```
<Virtualization>
    <Properties>
        <Property Name="AppVRuntimeDrive" Value="G:" />
        <Property Name="AppVServerURLPath" Value="%PackageName%_v%PackageVersion%" />
    </Properties>
</Virtualization>
```

To define a default value for any of the properties in the `ISVirtualPackage` table, create a `<Property>` in the `<Properties>` element and set a value. In the examples above, the `AppVRunTimeDrive` property is set to a default value of `G:`, and the `AppVServerURLPath` property is set to a default value of `%PackageName%_v%PackageVersion%`.

The following three replaceable parameters are only valid for the `AppVServerURLPath` property:

- **%PackageName%**—Name of the virtual package (which normally corresponds to the MSI `ProductName`).
- **%PackageVersion%**—Version number. (Each new upgrade increments this number.)
- **%PackageVersionedName%**—This is the `%PackageName%` for version one packages, and `%PackageName%_v%PackageVersion%` otherwise.

Virtual Converter Table Documentation for Microsoft App-V and VMware ThinApp

The following documentation lists settings that you can use to customize your conversion process.

- **Per package**—You can use InstallShield to directly edit the `ISVirtualPackage` table to modify the settings referenced below. You could also use the App-V, ThinApp, or Citrix XenApp Assistants user interface to modify the settings.
- **Per Automated Application Converter project**—You can specify a limited set of options in Automated Application Converter's **Project Options** dialog box, but you cannot edit tables directly.
- **Globally for any conversion**—You could edit the `Settings.xml` file to specify default values for many of the settings that can be specified in the `ISVirtualPackage` table.

The table settings that you can edit to customize your conversion process are organized into the following sections:

- General Settings
- Microsoft App-V Settings
- VMWare ThinApp Settings

General Settings

The following settings are applicable to all virtual technologies.

- ISVirtualPackage Table

- ISVirtualRelease Table
- Miscellaneous Virtual Conversion Settings

ISVirtualPackage Table

The `ISVirtualPackage` table is the main table that stores package-wide conversion settings. To edit this table, open the package in InstallShield and open the Direct Editor view. Also, if you make selections in the InstallShield Assistants, it will modify the settings in this table.

If you want to modify these settings globally, you need to edit the `Settings.xml` file, as described in Editing the Settings.xml File.

Table 10-54 • General Settings in ISVirtualPackage Table

Setting Name	Setting Value	Meaning
Provider	Semicolon separated list of Thinstall, AppV, and Citrix	Indicates virtual technologies to which to convert MSI packages.
MSIFile0, MSIFile1, etc	Absolute path to MSI	Indicates other MSI packages to suite together with the current one into one package.
VirtualPackageBuildOutputFolder	Absolute path to a directory	Instead of creating the converted virtual applications in a folder next to the source MSI, put them in a new folder under this specified location - this overrides the global redirect option in settings.xml.

ISVirtualRelease Table

The `ISVirtualRelease` table stores the relationship between InstallShield project releases and the virtual package type you want to build. This table is only relevant when you are editing an InstallShield Basic MSI project (not when you are editing an MSI package in the DirectEdit mode). If you make the relevant selections in the Assistants, it will modify the settings in this table.

Note • *The settings in this table cannot be specified in the Settings.xml file.*

Table 10-55 • General Settings in ISVirtualRelease Table

ISRelease_	ISProductConfiguration_	Name	Value	Meaning
`Key to ISRelease`	Key to `ISProductConfiguration`	`BuildVirtualPackage`	1	Build virtual package when associated release is built

Table 10-55 • General Settings in ISVirtualRelease Table

ISRelease_	ISProductConfiguration_	Name	Value	Meaning
`Key to ISRelease`	Key to `ISProductConfiguration`	`Provider`	Semicolon separated list of Thinstall, AppV, and Citrix	Indicates virtual technologies to which to convert MSI packages

Miscellaneous Virtual Conversion Settings

You can edit the following XML file to modify global settings that also govern the creation of virtual packages.

Table 10-56 • Miscellaneous Settings

Location	Name	Value	Meaning
`System\Msi.xml`	`IgnoreTables`	MSI table names	Control whether an error or warning is flagged for certain tables during conversion
`System\Msi.xml`	`IgnoreCustomActions`	MSI custom action names	List of custom actions that can safely be ignored during virtual conversion
`System\Msi.xml`	`PropertyDefaults`	MSI property names with given values	Default values to use for certain MSI properties rather than flagging them as warnings
`Support\0409\settings.xml`	`GlobalBuildRedirectFolder`	Absolute directory path	Instead of creating the converted virtual applications in a folder next to the source MSI, put them in a new folder under this specified location

Microsoft App-V Settings

The `ISVirtualPackage` table is the main table that stores package-wide App-V conversion settings. To edit this table, open the package in InstallShield and open the Direct Editor view. Also, if you make selections in the InstallShield Assistants, it will modify the settings in this table.

Note *• If you want to modify the setting in the* `ISVirtualPackage` *table globally, you can edit the* `Settings.xml` *file, as described in* *Editing the Settings.xml File*

The other tables listed here (directory, file, registry, shortcut) store App-V conversion settings related to a particular item in the package, such as a particular shortcut, file, registry entry, or directory.

Note • *The settings in these four tables cannot be specified in the* `Settings.xml` *file.*

- *ISVirtualPackage Table*
- *ISVirtualDirectory Table*
- *ISVirtualFile Table*
- *ISVirtualRegistry Table*
- *ISVirtualShortcut Table*

ISVirtualPackage Table

The following are App-V settings in the ISVirtualPackage table.

Table 10-57 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
`AppVName`		Same as name of MSI	Specify package name
`AppVServerURLHost`			Server location of SFT file
`AppVServerURLPort`			Server location of SFT file
`AppVServerURLProtocol`	RTSP, RTSPS, FILE, HTTP, or HTTPS		Protocol to use to access SFT file location
`AppVRootFolderName`		8.3 name based on product name and version	Specify root folder name
`AppVComments`			SFT file comments
`AppVOS`	Bitwise or of flags representing OS	0	0 indicates OS independent. Otherwise, here is the OS list starting with bit 1: WinXP, WinXP64, Win2003Svr, Win2003TS, Win2003TS64, Win2008Svr, Win2008TS, Win2008TS64, WinVista, WinVista64, Win7, Win764, Win2008R2TS64
`AppVDSC0`, `AppVDSC1`, etc.	Absolute path to OSD or SFT file [: MANDATORY]		Dynamic Suite Composition settings

Table 10-57 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
AppVNoCompression	1	0	Compression setting - default is compressed
AppVPackageOptimization	Offline or Stream	Stream	Only the shortcut targets are put in feature block 1 (FB1) if Stream is selected. Otherwise the entire package is put in FB1.
AppVUpgrade	1	0	Enables creation of an upgrade package
AppVUpgradePreviousPackage			Absolute path to SFT from previous package that will be upgraded.
AppVUpgradeLatest	1	0	Will locate the most recently built App-V package based on modified timestamp on SFT files found in appropriately named sub-folders next to the MSI file.
AppVUpgradeAppendPackageVersion	1	1	Package version will be appended to the end of the SFT file name
AppVDiagFileSystem	1	0	Include File System Diagnostic tool - a shortcut is included to run cmd.exe from the physical System32 folder. This cmd.exe and any programs launched from it will have access to the virtual environment of the package
AppVDiagRegistry	1	0	Include Registry System Diagnostic tool - a shortcut is included to run regedit.exe from the physical Windows folder. It will have access to the virtual environment.
AppVTestLauncher	1	1	AppVLauncher.exe is copied next to the newly built App-V package. This tool can be used to easily test deploy App-V packages.

Table 10-57 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
BuildMSI	1	0	Create a wrapper MSI file that can be used to deploy the App-V package
AppVMsiWrapperCompress	1	0	Compression setting for wrapper MSI
AppVPrereq	1	0	Set this option to include App-V client setup as a setup prerequisite for the wrapper MSI. It will be necessary to obtain a redistributable copy of the App-V client setup to use this feature.
APPVLOADING	1	0	Set this option to not include the SFT file in the wrapper MSI. The SFT file will be streamed from the server location specified in the OSD and manifest files.
AppVNoSpacesInFileNames	1		Will replace spaces in the SFT, OSD, and Icon file names with '_'.
AppVSpaceReplacementString	Some string		Use together with setting AppVNoSpacesInFileNames property to 1. Any spaces in SFT, OSD, and Icon file names will be replaced by the string specified in the value of this property. If the string 'EMPTYSTRING' is used, then spaces will just be removed.
AppVRuntimeDrive	Drive letter such as M:	Q:	App-V client drive to use

ISVirtualDirectory Table

The following are App-V settings in the ISVirtualDirectory table.

Table 10-58 • App-V Settings in ISVirtualDirectory Table

Directory_	Name	Value	Meaning
Key into Directory table	AppVUserData	1	If set, then treat this directory as user data. If unspecified, then default algorithm is used to determine whether to mark directory as user data or application data.

Table 10-58 • App-V Settings in ISVirtualDirectory Table

Directory_	Name	Value	Meaning
Key into Directory table	AppVOverride	1	Override directory contents during upgrade

ISVirtualFile Table

The following are App-V settings in the ISVirtualFile table.

Table 10-59 • App-V Settings ISVirtualFile Table

File_	Name	Value	Meaning
Key into File table	AppVUserData	1	If set, then treat this file as user data. If unspecified, then default algorithm is used to determine whether to mark file as user data or application data.
Key into File table	AppVOverride	1	Override file during upgrade

ISVirtualRegistry Table

The following are App-V settings in the ISVirtualRegistry table.

Table 10-60 • App-V Settings in ISVirtualRegistry Table

Registry_	Name	Value	Meaning
Key into Registry table	AppVOverride	1	If set, virtual application will only see the registry key contents in the virtual package and no child keys that may be present on the physical machine. Otherwise, virtual application will see only values in the virtual package, but will see child keys present on the physical machine, if they are not also present in the virtual package.

ISVirtualShortcut Table

The following are App-V settings in the ISVirtualShortcut table.

Table 10-61 • App-V Settings in ISVirtualShortcut Table

Shortcut_	Name	Value	Meaning
Key into Shortcut table	AppVApplication	0	A value of zero indicates that this shortcut will not be included in the converted App-V package.

VMWare ThinApp Settings

The ISVirtualPackage table is the main table that stores package-wide ThinApp conversion settings. To edit this table, open the package in InstallShield and open the Direct Editor view. Also, if you make selections in the Assistants, it will modify the settings in this table.

Note • *If you want to modify the setting in the ISVirtualPackage table globally, you can edit the Settings.xml file, as described in Editing the Settings.xml File*

The other tables listed here (directory, file, registry, shortcut) store ThinApp conversion settings related to a particular item in the package, such as a particular shortcut, file, registry entry, or directory.

Note • *The settings in these four tables cannot be specified in the Settings.xml file.*

- ISVirtualPackage Table
- ISVirtualDirectory Table
- ISVirtualRegistry Table
- ISVirtualShortcut Table

ISVirtualPackage Table

The following are ThinApp settings in the ISVirtualPackage table.

Table 10-62 • ThinApp Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
ThinRemoveSandboxOnExit	1	0	Determines if sandbox is deleted when application exits
ThinSandboxRemovableDisk	1	0	Determines whether write operations to removable disks go to the disks or to sandbox
ThinSandboxNetworkDrives	1	0	Determines whether write operations to network drive go to network drive or to sandbox
ThinSandboxName			Name of directory that stores the sandbox
ThinActiveDirectory	1	0	Restrict access based on AD groups
ThinPermittedGroups	Semicolon separated AD group names		Only users belonging in the specified local or domain groups will be able to run the application

Table 10-62 • ThinApp Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
ThinAccessDeniedMsg			Message to display when access is denied
BuildMSI	1	0	Build a MSI that will install and register ThinApp package
ThinDisableTracing	1	0	Disable ThinApp tracing
ThinCompessionType	None/Fast	None	Compression setting
ThinDirectoryIsolationMode	Merged, WriteCopy, or Full	WriteCopy	Controls whether write operations are directed to the sandbox or physical drive
OptionalAppLinks	Semicolon separated list of absolute paths to ThinApp package EXE or DAT files		Used to establish optional AppLink relationship between ThinApp packages
RequiredAppLinks	Semicolon separated list of absolute paths to ThinApp package EXE or DAT files		Used to establish required AppLink relationship between ThinApp packages
AppSyncURL			HTTP, HTTPS, or File URL to web server that hosts application updates
AppSyncUpdatedMessage			Message to display to user upon update
AppSyncUpdateFrequency	Minutes (m), hours (h), days (d), and 0		The frequency with which to check for updates
AppSyncClearSandboxOnUpdate	1	0	Determines whether to clear sandbox after update
AppSyncExpirePeriod	Minutes (m), hours (h), days (d), and never		Number of days before application will stop working due to not being able to connect to server
AppSyncWarningPeriod	Minutes (m), hours (h), and days (d)		Start of warning period. After this ThinApp will check for an update each time the application starts
AppSyncWarningFrequency	Minutes (m), hours (h), and days (d)		Frequency of warnings to the user during the warning period

Table 10-62 • ThinApp Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
AppSyncExpireMessage			Message user sees when application cannot be run because it has failed to contact the server for the duration of the expiration period
AppSyncWarningMessage			Message user sees when application has not been able to contact the server for the duration of the warning period
ThinDiagCmd	1	0	Include File System Diagnostic tool
ThinDiagReg	1	0	Include Registry System Diagnostic tool

ISVirtualDirectory Table

The following are ThinApp settings in the ISVirtualDirectory table.

Table 10-63 • ThinApp Settings in ISVirtualDirectory Table

Directory_	Name	Value	Meaning
Key to Directory table	ThinIsolation	1 for default, 2 for Full, 4 for WriteCopy, and 8 for Merged	Sets isolation setting for a directory. Default setting is default (use overall package isolation setting)

ISVirtualRegistry Table

The following are ThinApp settings in the ISVirtualRegistry table.

Table 10-64 • ThinApp Settings in ISVirtualRegistry Table

Registry_	Name	Value	Meaning
Key to Registry table	ThinIsolation	1 for default, 2 for Full, 4 for WriteCopy, and 8 for Merged	Sets isolation setting for a registry key. Default setting is default (use overall package isolation setting)

ISVirtualShortcut Table

The following are ThinApp settings in the ISVirtualShortcut table.

Table 10-65 • ThinApp Settings in ISVirtualShortcut Table

Shortcut_	Name	Value	Meaning
Key into Shortcut table	`ThinIsolation`	0	A value of zero indicates that this shortcut will not be included in the converted ThinApp package.

Additional Settings

You can edit the following configuration files to further customize the conversion process of a ThinApp package.

Table 10-66 • Additional ThinApp Settings

Location	Name	Value	Meaning
`System\Thinstall.xml`	`AlwaysBuildDatFile`	true/false	Always put the application payload in a DAT file - even with only one shortcut
`System\Thinstall.xml`	`PreserveMostPackageIniSettings`	true/false	Start with existing package.ini file from previous build instead of from template package.ini. This has the potential of preserving many user customizations made directly to package.ini file.
`System\Thinstall.xml`	`CreateBuildBatFile`	true/false	Copy a template BAT file into the Interim folder that can be used to rebuild the ThinApp package without running the full MSI conversion virtual build process
`System\Package.ini`			Template package.ini file used in the virtual build process that can be customized somewhat
`System\Thinapp.bat`			Template BAT file that can be used to build ThinApp package directly without running full virtual build process

Editing the Settings.xml File

To edit the `Settings.xml` file, add a property element for each setting in the `Virtualization/Properties` section of the file. You can find the `Settings.xml` file in the following directory:

```
C:\Program Files\AdminStudio\2022 R2\Repackager\Support\0409
```

Edit the following section of the file:

```
<Virtualization>
...
    <Properties>
        <!--Use this section to provide a global default for any setting
        that is found in the ISVirtualPackage table-->
        <!--<Property Name="AppVRuntimeDrive" Value="G:"/>-->
        <!--<Property Name="AppVServerURLPath" Value="%PackageName%_v%PackageVersion%" />-->
    </Properties>
</Virtualization>
```

Troubleshooting

This section includes information to help you resolve typical problems that you might encounter when using the Automated Application Converter. The following sections are included:

- First Things to Check
- Problems and Solutions
- Best Practices for Optimal Performance
- How to Test a Virtual Machine
- Resolving Problems Connecting to a Hyper-V Image
- Automated Application Converter Error Messages
- Virtualization Conversion Error Messages

First Things to Check

If you encounter a problem when performing package conversion, first scan this table to review of list of the most likely causes for conversion failure.

Table 10-67 • Most Likely Causes of Errors

Cause of Error	Resolution
Did not run the Virtual Machine Preparation Tool	On each virtual machine that you are going to use to perform automated repackaging, you need to run the Virtual Machine Preparation Tool, an application that will enable automatic login. See Preparing Your Virtual Machines for Use With the Automated Application Converter. ***Important*** • *If you do not run the Virtual Machine Preparation Tool on the virtual machines you want to use, the Automated Application Converter will be unable to connect to them.*

Table 10-67 • Most Likely Causes of Errors

Cause of Error	Resolution
Did not install VMware vSphere API	If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation 6.5 or later), you need to have the VMware vSphere API installed on the same machine as the Automated Application Converter. See VMware vSphere API Requirement on the AdminStudio Machine.
Snapshot does not exist on the virtual image	After you run the Virtual Machine Preparation Tool on a virtual machine, you need to shut it down and create a snapshot. This enables the Automated Application Converter to revert the virtual image to a clean state after each repackaging run. See Taking a Snapshot. ***Important*** • *If a snapshot does not exist on the virtual machine, not only will repackaging on that virtual machine fail, but you will also be unable to use that virtual machine to perform testing (as described in Testing Packages).*
Name of Snapshot on virtual image is not identified properly	If your virtualization technology supports named snapshots, you should name the snapshot AutoRepack_Base, which is the default name that the Automated Application Converter will be looking for. If you assign a snapshot name other than AutoRepack_Base, after you add the virtual machine to the Automated Application Converter, you need to specify that snapshot name in the **Snapshot Name** property in the **Properties** window of the **Machines** tab for that machine. See Editing Virtual Machine Properties on the Machines Tab. ***Important*** • *If the snapshot on the virtual machine is not identified properly in the Automated Application Converter, not only will repackaging on that virtual machine fail, but you will also be unable to use that virtual machine to perform testing (as described in Testing Packages).*
ThinApp client is not installed	If you choose to build ThinApp applications, AdminStudio will convert the package installation into a format compatible with ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools. As a prerequisite to building a ThinApp application from AdminStudio, you must have installed ThinApp and accepted any and all license agreements. For more information, see ThinApp on the VMware website.
Password of the virtual image has changed	When you add a virtual image to a project, you are prompted for the user name and password to logon to that machine. If you entered an incorrect password or if the password has recently changed, you need to edit that machine's **Guest Password** property on the **Machines** tab. See Editing Virtual Machine Properties on the Machines Tab.
Virtual machine is corrupted, or cannot be launched	If the Automated Application Converter is attempting to connect to a virtual machine that is corrupt or cannot be launched, conversion will fail. To make sure that your virtual machines are in proper working order, attempt to launch them manually (outside of the Automated Application Converter) using the configuration tool of the virtual technology.

Table 10-67 • Most Likely Causes of Errors

Cause of Error	Resolution
Virtual machine does not have network connectivity	In order for the Automated Application Converter to use a virtual machine, the virtual machine must have connectivity to your network. From the host machine, try to manually browse to the C drive of the virtual machine by entering the following address: `\\virtual_machine_name\C$`
Repackaging is taking a very long period of time	If the repackaging of a package is taking a very long period of time, you may want to verify that the value for that package's **Compressed** property is correct. If a package is in a directory that contains many other applications, and its **Compressed** property is set to **True**, the Automated Application Converter knows that only that one file needs to be copied to the virtual machine for repackaging. However, if the **Compressed** property set is set to **False**, there is no way to determine which of the files in that directory belong to the package, so all of the files in the directory must be copied to the virtual machine before repackaging can start. See Editing Package Properties on the Packages Tab. ***Tip*** • *It is recommended that each package be placed in its own directory to avoid problems such as this one.*

Problems and Solutions

The following chart lists some typical problems that you might encounter when using the Automated Application Converter and some suggested solutions.

Table 10-68 • Solutions to Common Problems

Problem	Possible Causes	Solution
Cannot connect to a virtual machine	Virtual machine has not been prepared.	Verify that the Virtual Machine Preparation Setup (`VMCfg.exe`) was run on the virtual machine to enable automatic login. See Running the Virtual Machine Preparation Setup for instructions. ***Tip*** *• A quick way to determine if the Virtual Machine Preparation Tool has been run on a virtual image is the presence of the `GuestAgent.exe` file on the root of the `C:` drive.*
	Specified **Guest Username** or **Guest Password** property is not specified correctly.	When you add a virtual machine to the Automated Application Converter, you specify the **User name** and **Password** on the **User Credentials** panel. If you entered an incorrect value or if one of these values has changed, you will be unable to connect to the virtual machine. Open the **Machines** tab and verify that the values in the **Guest Username** and **Guest Password** properties for the virtual machine are correct. ***Note*** *• When using a domain account, do not include the domain name in the **Guest Username** property.*
Cannot connect to a Windows 7 or Windows Server 2008 virtual machine	User Account Control (UAC) settings on a Windows 7 or Windows Server 2008 virtual machine could be causing problems during auto-login.	Make sure that you run the Virtual Machine Preparation Setup on the virtual machine to disable UAC. See Running the Virtual Machine Preparation Setup.
Unable to add packages from an AdminStudio Application Catalog	User does not have required login and/or view permissions on the Application Catalog.	Try to manually view packages in an AdminStudio Application Catalog to see if you have the required view permissions. Use AdminStudio Application Catalog to view the packages in the Application Catalog. Also, if using an AdminStudio Application Catalog with the Software Repository, make sure you have view permission to the Software Repository location of that Application Catalog.

Table 10-68 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Cannot connect to an AdminStudio Application Catalog	There are connection issues between domains due to user name.	Try to manually connect to the Application Catalog to make sure that you are using the correct credentials or that there are no other networking issues.
Unable to publish packages to an AdminStudio Application Catalog	User does not have required permission to import/publish packages.	Try to manually publish and/or import a package to an AdminStudio Application Catalog to see if you have the required permissions: Use AdminStudio Application Catalog Import Wizard to manually import a package into the Application Catalog. This enables you to determine if you have import permission. Also, if using an AdminStudio Application Catalog with the Software Repository, make sure you have write permission to the Software Repository location of that Application Catalog.
Cannot connect to a Microsoft Hyper-V Server	DCOM configuration settings need to be adjusted.	Adjust the DCOM settings, as described in this MSDN article, Connecting to WMI on a Remote Computer: http://msdn.microsoft.com/en-us/library/aa389290%28VS.85%29.aspx
Copy errors are generated during package conversion	User does not have permission to the **Output Path** location specified in the Automated Application Converter.	To make sure that you have permission to the Automated Application Converter output directory, perform the following steps: ***To specify Output Path directory:*** 1. Open the Automated Application Converter. 2. Select **Options** on the **Tools** menu. The **Project Options** dialog box opens. 3. Locate the **Output Path** setting under **Conversion Options**. By default, the path is `C:\Users\[UserName]\Documents\AutoRepack`. 4. Browse to that location using Windows Explorer to and attempt to copy a file to that location. 5. If you are unable to copy a file to that location, change the **Output Path** to a location that you do have write permission on.

Table 10-68 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Package status changes to soft or hard timeout	The installation is requesting user input.	In order for the Automated Application Converter to perform automated repackaging, packages must support silent installation mode and the silent installation mode command line parameters must be specified in the **Command Line** property for each package. To resolve this problem, open the **Packages** tab and make sure that the package's **Command Line** property contains the command line parameters to run the installation silently. ***Note*** • *The Automated Application Converter automatically populates the* ***Command Line*** *property for Windows Installer (`.msi`) packages that you add to the project. However, for non-MSI packages, you must manually enter the* ***Command Line*** *property.* If the **Command Line** property is specified, but you are still receiving a timeout error, you may want to try to manually run the installation using the command line parameters to make sure that the installation is not requesting any user input.
App-V application does not launch	VM does not have App-V client installed	If you are able to connect to the virtual machine, but the App-V package will not launch, make sure that the App-V client is installed on the virtual machine. Also, make sure that App-V file streaming is enabled on the virtual machine. ***Note*** • *When you select* ***Launch Package for Testing****, you are prompted to* ***Install the Microsoft App-V Client*** *on the* ***Select Virtual Machine*** *panel. See Testing Packages.*
Cannot build a ThinApp package	ThinApp application is not installed.	To create a ThinApp application, you are required to have ThinApp installed on the same machine as the Automated Application Converter.
Cannot connect to Hyper-V server	Hyper-V configuration tools are not installed.	Make sure that the Hyper-V configuration tools are installed on the Hyper-V Server's Hyper-V Microsoft Management Console.
Virtualization Readiness status of a package is "Unknown" (?)	Location of source files is no longer accessible.	When you originally added the package to your Automated Application Converter project, you were able to access the source files, but now the source files are either no longer there or you no longer have permission to access them.

Table 10-68 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Repackaging is taking a long time	It is taking a long time to copy the files required for repackaging to the virtual machine and back to the host machine.	Try to copy the files manually from the host machine to the virtual machine to attempt to identify the cause of the delay. For VMware, try switching the **Network connection** setting between **Bridged** and **NAT** on the virtual machine to see if it helps to improve the copy speed.
	The package's **Compressed** property setting is incorrect.	If a package is in a directory that contains many other applications, and its **Compressed** property is set to **True**, the Automated Application Converter knows that only that one file needs to be copied to the virtual machine for repackaging. However, if the **Compressed** property set is set to **False**, there is no way to determine which of the files in that directory belong to the package, so all of the files in the directory must be copied to the virtual machine before repackaging can start. See Editing Package Properties on the Packages Tab. ***Tip*** *• It is recommended that each package be placed in its own directory to avoid problems such as this one.*

Table 10-68 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Cannot connect to a VMware virtual machine	The latest version of the VMware vSphere API is not installed on the same machine as the Automated Application Converter.	If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation 6.5 or later), you need to have the VMware vSphere API installed on the same machine as the Automated Application Converter. See VMware vSphere API Requirement on the AdminStudio Machine.

	You are using VMware Workstation 7.01, but the `wrapper-config.txt` file does not identify Workstation 7.0.1	If running VMware Workstation 7.0.1, verify that the vSphere `wrapper-config.txt` contains the following line that identifies a Workstation 7.0.1. For example:
		`# Workstation 7.0.1`
		`ws 9vmdb 7.0.1 Workstation-7.0.0`
		`player 9 vmdb 3.0.1 workstation-7.0.0`

	You are attempting to connect to a VMware ESX and a VMware Workstation image at the same time.	When performing repackaging on a VMware virtual machine using the Automated Application Converter, you should connect to either a VMware ESX/ESXi image or a VMware Workstation image, but not both at the same time.
	vSphere API is not working properly on virtual machine.	If the vSphere API is not working, verify that `vmrun.exe` works. You can enter `vmrun.exe /?` to get the parameters for this tool. For example, to power up a virtual image on a VMware ESX server, use this command: `C:\Program Files (x86)\VMware\VMware vSphere>vmrun -h https://172.17.1.221/sdk -T esx -u root -p hostpassword -gu` *`UserName`* `-gp guestpassword start "[QA_ISO] Windows 7 Ultimate - 32-bit/ Windows 7 Ultimate - 32-bit.vmx"`

Best Practices for Optimal Performance

When setting up the Automated Application Converter to perform automated repackaging and conversion to virtual packages, you should following these best practices:

Table 10-69 • Best Practices

Practice	Description
Machine containing virtual images should be dedicated for use with the Automated Application Converter	Machines that contain the virtual images that you will use with the Automated Application Converter should be dedicated for use with this tool only.
Virtual machine needs adequate resources	Make sure that the machine containing the virtual images has adequate resources.

Table 10-69 • Best Practices

Practice	Description
Each package should be in its own folder	Each package that you add to an Automated Application Converter project should be in an individual folder, especially if they are uncompressed packages.
Manually test a virtual machine before adding it to an Automated Application Converter project	Before adding a virtual machine to your Automated Application Converter project, boot it up manually to make sure it is running properly. Make sure that no message boxes open (such as a `Failure to Start Service` error message) that would require user input.
Windows 7 is recommended	It is recommended that you install the Automated Application Converter on a machine running Windows 7, either 32-bit or 64-bit.
Disable unnecessary devices	On each virtual machine, disable devices that are not required, such as: • CD/DVD drives • USB Controller • Floppy drives • Sound cards
Turn off Windows Update on the virtual machines	To avoid problems with repackaging, turn off Windows Update on the virtual machines.

How to Test a Virtual Machine

If you are having trouble using a virtual machine with the Automated Application Converter, you may want to perform the following steps to manually test that virtual machine to see if it is in working order.

Task ***To manually test a virtual machine:***

1. Connect to the technology provider: Microsoft Hyper-V Server, VMware ESX or ESXi Server, or VMware Workstation 6.5 or later.
2. Use Remote Desktop to connect to the virtual machine you want to test.
3. Launch the virtual machine to make sure that it boots up properly and that you can login using the user name and password you specified when you added the virtual machine to your Automated Application Converter project.

   Tip • *If you discover that you specified an incorrect user name or password, update the* ***Guest Username*** *and* ***Guest Password*** *properties on the* ***Machines*** *tab for this machine.*
4. Check to see if a snapshot exists and that snapshot name is either `AutoRepack_Base` or that you have specified an alternate name in the **Snapshot Name** property on the **Packages** tab for that virtual machine.

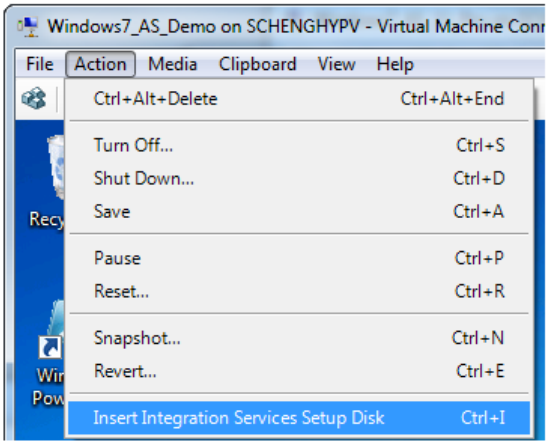
5. Manually copy the Repackager folder in the AdminStudio installation directory to the virtual machine to determine if you have write permission on the virtual machine.
6. Manually copy an application to the virtual machine.
7. Launch the Repackaging Wizard and repackage that application.
8. Copy the captured data from the virtual machine to the host machine in the output location specified for that package in the **Path** property on the **Packages** tab to test if you have permission to write to that location.
9. Repeat these steps for each of the virtual machines you are using with your project.

Resolving Problems Connecting to a Hyper-V Image

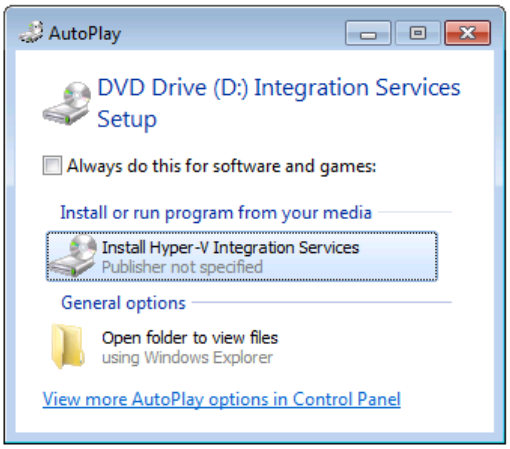
In some instances, Automated Application Converter may encounter a problem connecting to a Hyper-V image because it is failing to obtain an IP address from the Hyper-V Server. To resolve this issue, perform these steps:

Task ***To obtain IP address from Hyper-V Server:***

1. Open the image using the Hyper-V Console.
2. On the **Action** menu, select **Insert Integration Services Setup Disk**.



You will then be prompted to install the latest version of the Hyper-V Integration Services:



3. Click **Install Hyper-V Integration Services**.
4. When the installation of the Hyper-V Integration Services has completed, create a new snapshot of the Hyper-V image and name the snapshot AutoRepack_Base, which is the default name that the Automated Application Converter will be looking for.
5. Restart the Automated Application Converter process and attempt to connect to this Hyper-V image.

Automated Application Converter Error Messages

This section includes information on how to resolve the following error messages that could be generated by the Automated Application Converter:

- Error -4308: VM failed to start up
- Error -4309: VM failed to shut down
- Error -4310: Failed to connect to VM
- Error -4312: Failed to prepare Repackager
- Error -4313: Failed to access the package
- Error -4314: Failed to copy repackaged output from virtual machine
- Error -4315: Failed to send command to VM
- Error -4316: Failed getting response from VM
- Error -4317: Failed running pre-snapshot
- Error -4318: Failed running post-snapshot
- Error -4319: Failed running package installation
- Error -4320: Failed creating folder on VM
- Error -4333: Preparing command-line...
- Error -4380: Failed to prepare AppV
- Error -4388: Failed preparing for pre-snapshot

- Error -4389: Failed connecting to server
- Error -4390: Failed connecting to image
- Error -4391: Failed to reboot
- Error -4395: Failed to create VM directory
- Error -4409: Failed to delete package cache folder

Debug Messages in the Automated Application Converter Log Report

By default, debug messages that occur during a conversion run are saved in the AdminStudio Automated Application Converter Log report, but the display of those debug messages is turned off. However, if you are using Microsoft Internet Explorer 11 as your default browser, you can choose to view those debug messages. See Viewing Debug Messages for instructions.

Error -4308: VM failed to start up

The following table documents this message:

Table 10-70 • Error -4308: VM failed to start up

Category	Description
Message:	`Error -4308 controlling virtual machine: VM failed to start up`
Cause:	Automated Application Converter is unable to access this VMware virtual machine due to a failure to login to the virtual machine server or into the guest virtual machine.
Resolution:	Open the **Machines** tab and verify that the following properties are set correctly for this virtual machine: • **Machine Settings**—Verify the **Guest Username** and **Guest Password** properties. • **Virtual Machine Server**—Verify the **Server Username** and **Server Password** properties.

Error -4309: VM failed to shut down

The following table documents this message:

Table 10-71 • Error -4309: VM failed to shut down

Category	Description
Message:	`Error -4309 controlling virtual machine: VM failed to shut down`
Cause:	Automated Application Converter is unable to access this machine virtual machine in order to shut it down.

Table 10-71 • Error -4309: VM failed to shut down

Category	Description
Resolution:	Open the **Machines** tab and verify that the following properties are set correctly for this virtual machine: • **Machine Settings**—Verify the **Guest Username** and **Guest Password** properties. • **Virtual Machine Server**—Verify the **Server Username** and **Server Password** properties.

Error -4310: Failed to connect to VM

The following table documents this message:

Table 10-72 • Error -4310: Failed to connect to VM

Category	Description
Message:	`Error -4310 controlling virtual machine: Failed to connect to VM`
Cause:	This error could be caused by the following reasons: • Virtual machine was unexpectedly shut down early. • Operating system on the virtual machine does not launch. • The Guest Agent is not running on the virtual machine. • Permissions to the virtual machine are incorrect. • Virtual machine `AutoRepack_Base` snapshot was taken before the machine was powered off.
Resolution:	To resolve this error, try the following: • After you have run the Virtual Machine Preparation Tool on this virtual machine, verify that the snapshot launches, automatically logs into the virtual machine, and that the Guest Agent opens. • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Do not shut down the virtual machine in the middle of using it. • Make sure that you power off the virtual machine before taking the `AutoRepack_Base` snapshot.

Error -4312: Failed to prepare Repackager

The following table documents this message:

Table 10-73 • Error -4312: Failed to prepare Repackager

Category	Description
Message:	`Error -4312 controlling virtual machine: Failed to prepare Repackager`

Table 10-73 • Error -4312: Failed to prepare Repackager

Category	Description
Cause:	The Automated Application Converter could not read the Repackager source location or could not write to the Repackager cache location.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties under **Machine Settings** on the **Machines** tab for this virtual machine are set correctly. • Verify that the virtual machine's **Repackager Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4313: Failed to access the package

The following table documents this message:

Table 10-74 • Error -4313: Failed to access the package

Category	Description
Message:	`Error -4313 processing package: Failed to access the package`
Cause:	The Automated Application Converter could not read from the package source location or could not write to the package cache location.
Resolution:	To resolve this error, try the following: • Ensure that the host machine has access to the network and is visible to/accessible from the virtual machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • If the package is on a network share, verify that both the host machine and virtual machine have access to that network share. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Verify that the package does not already exist as read-only.

Error -4314: Failed to copy repackaged output from virtual machine

The following table documents this message:

Table 10-75 • Error -4314: Failed to copy repackaged output from virtual machine

Category	Description
Message:	`Error -4314 controlling virtual machine: Failed to copy repackaged output from virtual machine`
Cause:	The virtual machine could not read the output cache location, or could not write to the project output location.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • If the project **Output Path** is set to a network share, verify that the host machine has access to that network share.

Error -4315: Failed to send command to VM

The following table documents this message:

Table 10-76 • Error -4315: Failed to send command to VM

Category	Description
Message:	`Error -4315 controlling virtual machine: Failed to send command to VM`
Cause:	There was a network error sending a command from the host machine to the Guest Agent on the virtual machine.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the Guest Agent running on the virtual machine has not crashed. • Update the Guest Agent on the virtual machine by running the latest version of the Virtual Machine Preparation Tool (`VMCfg.exe`) and taking a new snapshot.

Error -4316: Failed getting response from VM

The following table documents this message:

Table 10-77 • Error -4316: Failed getting response from VM

Category	Description
Message:	`Error -4316 controlling virtual machine: Failed getting response from VM`
Cause:	The Automated Application Converter did not receive a response from the virtual machine.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the Guest Agent running on the virtual machine has not crashed. • Update the Guest Agent on the virtual machine by running the latest version of the Virtual Machine Preparation Tool (`VMCfg.exe`) and taking a new snapshot.

Error -4317: Failed running pre-snapshot

The following table documents this message:

Table 10-78 • Error -4317: Failed running pre-snapshot

Category	Description
Message:	`Error -4317 processing package: Failed running pre-snapshot`
Cause:	The virtual machine could not create the output cache location or could not run Repackager.
Resolution:	To resolve this error, try the following: • Ensure that the host machine has access to the network and is visible to/accessible from the virtual machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** and **Repackager Cache Path** properties on the **Machines** tab for this virtual machine are set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Consider changing the **Repackaging Method** property for the package to **Installation monitoring**, which would eliminate the need for a pre-snapshot.

Error -4318: Failed running post-snapshot

The following table documents this message:

Table 10-79 • Error -4318: Failed running post-snapshot

Category	Description
Message:	`Error -4318 processing package: Failed running post-snapshot`
Cause:	The virtual machine could not run Repackager.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** and **Repackager Cache Path** properties on the **Machines** tab for this virtual machine are set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Consider changing the **Repackaging Method** property for the package to **Installation monitoring**, which would eliminate the need for a post-snapshot.

Error -4319: Failed running package installation

The following table documents this message:

Table 10-80 • Error -4319: Failed running package installation

Category	Description
Message:	`Error -4319 processing package: Failed running package installation`
Cause:	The virtual machine could not run Repackager or the virtual machine could not launch the application.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4320: Failed creating folder on VM

The following table documents this message:

Table 10-81 • Error -4320: Failed creating folder on VM

Category	Description
Message:	`Error -4320 processing package: Failed creating folder on VM`
Cause:	Virtual machine was unable to create the Setup Cache folder. ***Note*** • *This error may be related to Error -4313: Failed to access the package.*
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4333: Preparing command-line...

The following table documents this message:

Table 10-82 • Error -4333: Preparing command-line...

Category	Description
Message:	`Error -4333 processing package: Preparing command-line...`
Cause:	The virtual machine could not query the associated program for the provided extension.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the package's source file (as specified in the **Setup Cache Path** property on the **Machines** tab) can run when double-clicked on the virtual machine. • If any special tools are required (script engines, etc.) to run the command line, install them on the virtual machine and retake the snapshot.

Error -4380: Failed to prepare AppV

The following table documents this message:

Table 10-83 • Error -4380: Failed to prepare AppV

Category	Description
Message:	`Error -4380 controlling virtual machine: Failed to prepare AppV`
Cause:	The virtual machine was unable to read the App-V client installation sources, or was unable to write to the Setup Cache Path.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify App-V Client installation folder was specified correctly on the Select Virtual Machine Dialog Box. • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4388: Failed preparing for pre-snapshot

The following table documents this message:

Table 10-84 • Error -4388: Failed preparing for pre-snapshot

Category	Description
Message:	`Error -4388 processing package: Failed preparing for pre-snapshot`
Cause:	The virtual machine was unable to write to the Output Cache Path.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4389: Failed connecting to server

The following table documents this message:

Table 10-85 • Error -4389: Failed connecting to server

Category	Description
Message:	`Error -4389 controlling virtual machine: Failed connecting to server`
Cause:	The Automated Application Converter was unable to connect to the virtual machine server. The server machine may be unavailable.
Resolution:	To resolve this error, try the following: • Verify that the **Server Address**, **Server Username**, and **Server Password** properties under **Virtual Machine Server** on the **Machines** tab are set correctly for this virtual machine. • Verify that the Hyper-V or VMware ESX/ESXi server is running. • Ensure that the virtual server has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • If you are using VMware and you received XE values 22002 (55F2) or 22003 (55F3), verify that the VMware vSphere API is installed. See VMware vSphere API Requirement on the AdminStudio Machine.

Error -4390: Failed connecting to image

The following table documents this message:

Table 10-86 • Error -4390: Failed connecting to image

Category	Description
Message:	`Error -4390 controlling virtual machine: Failed connecting to image`
Cause:	The Automated Application Converter was unable to connect to the virtual machine. The virtual machine may be unavailable or the virtual machine credentials may be incorrect.
Resolution:	To resolve this error, try the following: • Verify that the virtual machine has not been deleted. • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • If you are using VMware ESXi Server and you received an XE value of 0020 (corresponding to a `vSphere_E_LICENSE` error), this indicates that you may need to purchase a license for your VMware ESXi Server in order to use it with the Automated Application Converter.

Error -4391: Failed to reboot

The following table documents this message:

Table 10-87 • Error -4391: Failed to reboot

Category	Description
Message:	`Error -4391 controlling virtual machine: Failed to reboot`
Cause:	The Automated Application Converter could not access the virtual machine.
Resolution:	To resolve this error, try the following: • Verify that the **Server Address**, **Server Username**, and **Server Password** properties under **Virtual Machine Server** on the **Machines** tab are set correctly for this virtual machine. • Attempt to launch the virtual machine manually to ensure that it has not become corrupted.

Error -4395: Failed to create VM directory

The following table documents this message:

Table 10-88 • Error -4395: Failed to create VM directory

Category	Description
Message:	`Error -4395 controlling virtual machine: Failed to create VM directory`
Cause:	Virtual machine could not create a directory.
Resolution:	To resolve this error, try the following: • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Verify that the package's **Package** field does not include any characters that are invalid for a file name.

Error -4409: Failed to delete package cache folder

The following table documents this message:

Table 10-89 • Error -4409: Failed to delete package cache folder

Category	Description
Message:	`Error -4409 processing package: Failed to delete package cache folder`
Cause:	Virtual machine could not delete a directory. ***Note*** • *This error may be related to Error -4313: Failed to access the package.*
Resolution:	To resolve this error, try the following: • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the file or its containing folder is not locked due to being open in Windows Explorer, the Command Window, etc.

Virtualization Conversion Error Messages

When converting a Windows Installer package to a virtual application, error and warning messages are generated. Some of these messages are generic to package virtualization, and others are specific to the virtualization solution you are preparing packages for.

This section includes information on how to resolve error messages that could be generated by during virtualization using the Automated Application Converter, App-V Assistant, VMware ThinApp Assistant, and Citrix Assistant.

Error -9000: Unknown Exception

The following table documents this message:

Table 10-90 • Error -9000: Unknown Exception

Category	Description
Type:	Error
Message:	`An unknown exception occurred.`
Cause:	This is an unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9001: Unknown COM

The following table documents this message:

Table 10-91 • Error -9001: Unknown COM

Category	Description
Type:	Error
Message:	`Internal error.`
Resolution:	Contact AdminStudio Technical Support.

Error -9002: Error Opening Package

The following table documents this message:

Table 10-92 • Error -9002: Error Opening Package

Category	Description
Type:	Error
Message:	`An error occurred when opening the package.`
Cause:	This is an unexpected internal error when reading the Windows Installer package.
Resolution:	Check to make sure that the package is accessible to the user. If the error persists and the package is on a network share, copy the package locally (to avoid any network connection issues) and try again. If this does not solve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9003: Error Saving Package

The following table documents this message:

Table 10-93 • Error -9003: Error Saving Package

Category	Description
Type:	Error
Message:	`An error occurred when saving the package.`
Cause:	This is an unexpected internal error when trying to save the Citrix profile.

Table 10-93 • Error -9003: Error Saving Package

Category	Description
Resolution:	Check to see if the user has proper access to the location the profile is being built to. If this does not solve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9004: Process Cancelled By User

The following table documents this message:

Table 10-94 • Error -9004: Process Cancelled By User

Category	Description
Type:	Error
Message:	`Process cancelled by user.`
Cause:	The user clicked the **Cancel** button to stop the build.
Resolution:	Restart the build process.

Error -9005: Error Creating Temporary Folder

The following table documents this message:

Table 10-95 • Error -9005: Error Creating Temporary Folder

Category	Description
Type:	Error
Message:	`An error occurred while creating a temporary folder`
Cause:	You encounter this error when the user does not have permission to write to `C:\TMP`, or the drive is out of disk space.
Resolution:	Obtain write access to `C:\TMP`, and free some disk space on the drive, and then rebuild the profile.

Error -9006: Error Decompressing Package

The following table documents this message:

Table 10-96 • Error -9006: Error Decompressing Package

Category	Description
Type:	Error
Message:	`An error occurred while decompressing the package 'PackageName'.`
Cause:	You encounter this error when the package is a compressed Windows Installer package (`.msi`) and errors were generated when AdminStudio attempted to perform an administrative installation to extract the files.
Resolution:	When this error occurred, you should have also received a return error code from Windows Installer. Look up that error code in the Windows Installer Help Library to determine the cause of the problem. If you did not receive a return error code from Windows Installer, this error could have been caused by the package not being authored properly. In the Windows Installer package, check to see if the `AdminExecuteSequence` table was defined. If that table is missing, the package cannot be decompressed.

Error -9007: File With Extension Not Found

The following table documents this message:

Table 10-97 • Error -9007: File With Extension Not Found

Category	Description
Type:	Error
Message:	`No file found with the extension 'ComponentKeyName'.`
Cause:	This is an unexpected error that occurred when file extensions were being processed.
Resolution:	Check to make sure that the executable for the file extension exists and that it is set as the key file in its component.

Error -9008: Error Extracting Icon

The following table documents this message:

Table 10-98 • Error -9008: Error Extracting Icon

Category	Description
Type:	Error
Message:	`An error occurred while extracting the icon 'IconKeyName'`
Cause:	This is an unexpected error that occurred when an icon listed in the `Icon` table was being extracted.
Resolution:	Verify that the `Icon` entry in the `Icon` table is valid. If necessary, replace it with a valid icon.

Error -9009: Unknown Provider

The following table documents this message:

Table 10-99 • Error -9009: Unknown Provider

Category	Description
Type:	Error
Message:	`The specified provider is unknown 'ProviderName'.`
Cause:	This is an unexpected internal error.
Resolution:	Invalid data may have been modified via the Direct Editor causing this error. Delete the **Release** you are building, and then create a new one and rebuild.

Error -9010: Invalid Target File Name

The following table documents this message:

Table 10-100 • Error -9010: Invalid Target File Name

Category	Description
Type:	Error
Message:	`The target file name is invalid. 'FileName'`
Cause:	This is an unexpected internal error.

Table 10-100 • Error -9010: Invalid Target File Name

Category	Description
Resolution:	Invalid data may have been modified via the Direct Editor causing this error. Verify the **Name** field on the Citrix Assistant / ThinApp Assistant **Profile Information** page and make sure the name does not contain any invalid file name characters.

Error -9011: Error Reading MSI Table

The following table documents this message:

Table 10-101 • Error -9011: Error Reading MSI Table

Category	Description
Type:	Error
Message:	`Unexpected error reading MSI table 'TableName'`
Cause:	This is an unexpected error that occurred when the specified Windows Installer table was being processed.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9012: Unexpected Error in Method

The following table documents this message:

Table 10-102 • Error -9012: Unexpected Error in Method

Category	Description
Type:	Error
Message:	`Unexpected error in method 'MethodName'`
Cause:	This is an unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9013: Type Library Not Found

The following table documents this message:

Table 10-103 • Error -9013: Type Library Not Found

Category	Description
Type:	Error
Message:	`Type library not found: 'TypeLibraryName'`
Cause:	You encounter this error when a type library file was not found when trying to extract COM information.
Resolution:	Check to see if the type library file exists in the proper location when building the profile. If this does not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9014: ShellExecute Failed

The following table documents this message:

Table 10-104 • Error -9014: ShellExecute Failed

Category	Description
Type:	Error
Message:	`ShellExecute failed: 'CommandLine'`
Cause:	You encounter this error when the specified command line failed to launch a process.
Resolution:	Check to see if the executable file name shown is a valid file and that the user has the proper access rights to run it. If this does not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9015: Unable to Determine Full Path for Driver

The following table documents this message:

Table 10-105 • Error -9015: Unable to Determine Full Path for Driver

Category	Description
Type:	Warning

Table 10-105 • Error -9015: Unable to Determine Full Path for Driver

Category	Description
Message:	`Unable to determine the full path for driver 'DriverName'`
Cause:	You encounter this error when a driver referenced in the `ODBCDataSource` table is not being installed by the package.
Resolution:	This error can be resolved in one of two ways: **Editing the Windows Installer Package** 1. Edit the package using InstallShield Direct Edit Mode. 2. Navigate to the `ISVirtualPackage` table. 3. Create an entry as follows to identify the full path of the missing driver: **Name**: `<DriverName> Description` **Value**: `Path to Driver` **Manually Installing the Driver** Install the missing driver on your machine and then rebuild the Citrix profile.

Warning -9016: Contents of Table Ignored

The following table documents this message:

Table 10-106 • Warning -9016: Contents of Table Ignored

Category	Description
Type:	Warning
Message:	`Contents of table 'TableName' will be ignored`
Cause:	This error message identifies a known limitation of Citrix conversion.
Resolution:	If the contents of the table is deemed critical, repackage the application, and then rebuild the Citrix profile.

Warning -9017: .NET 1.x Assembly Not Supported

The following table documents this message:

Table 10-107 • Warning -9017: .NET 1.x Assembly Not Supported

Category	Description
Type:	Warning
Message:	`Assembly 'AssemblyName' is a .NET 1.x assembly and will not be converted correctly. Only .NET 2.0/3.0 assemblies are currently supported. You may wish to repackage this package first.`
Cause:	You encounter this error when attempting to convert a package containing a .NET 1.x assembly. Only .NET 2.0/3.0 assemblies are currently supported.
Resolution:	Repackage the application, and then rebuild the Citrix profile.

Warning -9018: Custom Actions Ignored

The following table documents this message:

Table 10-108 • Warning -9018: Custom Actions Ignored

Category	Description
Type:	Warning
Message:	`Custom action 'CustomActionName' will be ignored.`
Cause:	When converting a Windows Installer package to a Citrix profile, all custom actions are ignored. Any modifications to a target machine that a custom action in this Windows Installer package may create will not be propagated into the Citrix profile. ***Note*** • *When a custom action that does not modify the system or perform any part of the installation (such as an InstallShield Editor predefined custom action or a Type 19 custom action) is encountered, no message is generated. If a Type 51 custom action is encountered (which sets a property from a formatted text string), it is automatically resolved. If a Type 35 custom action is encountered, it is only resolved if it is referenced in the `Directory` table.*

Table 10-108 • Warning -9018: Custom Actions Ignored

Category	Description
Resolution:	The resolution that you should perform depends upon the purpose of the custom action: • If the custom action merely automatically enters a value or makes some other kind of minor modification, you can ignore this warning. • If the custom action does something which could change the behavior of the installation (such as setting a Property), you may need to resolve this issue. To resolve this issue, first attempt to launch the converted package on Citrix XenApp. If you receive any application errors, you need to repackage this application. ***To repackage a Windows Installer package to capture custom action functionality:*** 1. Use the Repackaging Wizard to repackage this application. The Repackaging Wizard monitors system changes as an application is installed, and then that data is converted into a Repackager project. 2. Build the Repackager project to generate a revised Windows Installer package. This new Windows Installer package does not contain any custom actions, but (as a result of being repackaged) it will include the functionality performed by the original custom action.

Warning -9019: Conditionalized Components

The following table documents this message:

Table 10-109 • Warning -9019: Conditionalized Components

Category	Description
Type:	Warning
Message:	`There exist one or more conditionalized components which may not be converted correctly`
Cause:	This warning is generated when attempting to convert conditionalized components because conditions on components are not evaluated.
Resolution:	Repackage the application on a machine that has a similar environment to the machines where the profile will be deployed. Then rebuild the Citrix profile. You can also evaluate the conditions on the listed components and remove the components you know are not needed for your target machines. Then rebuild the Citrix profile.

Error -9020: Directory With Null Parent

The following table documents this message:

Table 10-110 • Error -9020: Directory With Null Parent

Category	Description
Type:	Error
Message:	`Directory 'DirectoryName' has a null parent and will be ignored.`
Cause:	This error occurs if a directory table entry (other than `TARGETDIR`) is null.
Resolution:	Evaluate the ThinApp application to see if it works. If it does not work properly, you may want to consider repackaging the package.

Error -9021: Unable to Extract COM Data

The following table documents this message:

Table 10-111 • Error -9021: Unable to Extract COM Data

Category	Description
Type:	Error
Message:	`Unable to extract COM data for 'FileName'`
Cause:	This Windows Installer package has an entry in the `TypeLib` or `SelfReg` table that contains COM data that AdminStudio cannot convert to application data. Depending upon which file cannot be COM extracted, it is possible that this application will still work properly in Citrix XenApp isolation environment if you repackage this Windows Installer package with COM table mapping turned off. COM data is stored in the Windows Registry. So, if you repackage this Windows Installer package, the capture process will be able to obtain all of this data because it captures all modifications to the Registry and does not have to rely on COM extraction.
Resolution:	To resolve this issue, you need to repackage your Windows Installer package with COM table mapping turned off.

Error -9022: Complus Table

The following table documents this message:

Table 10-112 • Error -9022: Complus Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'Complus'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a `Complus` table. During the conversion process, the `Complus` table is not read.
Resolution:	The `Complus` table contains information needed to install COM+ applications. While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, this application cannot be deployed on Citrix XenApp.

Error -9024: FileSFPCatalog

The following table documents this message:

Table 10-113 • Error -9024: FileSFPCatalog

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'FileSFPCatalog'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **FileSFPCatalog** table. During the conversion process, the **FileSFPCatalog** table is not read.
Resolution:	The **FileSFPCatalog** table associates specified files with the catalog files used by system file protection. If this file is necessary for the function of the application, you need to use Repackager to repackage the application.

Warning -9026: LaunchCondition Table

The following table documents this message:

Table 10-114 • Warning -9026: LaunchCondition Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'LaunchCondition'.`

Table 10-114 • Warning -9026: LaunchCondition Table

Category	Description
Cause:	You encounter this warning when the Windows Installer package that you are converting includes a `LaunchCondition` table. During the conversion process, the **LaunchCondition** table is not read.
Resolution:	The **LaunchCondition** table contains a list of conditions that all must be satisfied for the installation to begin. For example, if an application requires Windows 7 to run, Windows 7 is listed in the **LaunchCondition** table. Because this table is not read, no check is performed. Therefore, when a user on an operating system other than Windows 7 launches this Citrix profile, the application may not function properly. To resolve this issue, perform one of the following tasks: • **Option 1: Set Requirements on the Profile Requirements Page**—If the launch conditions only include operating system, service pack, and language requirements, open this package in the InstallShield Editor Citrix Assistant / ThinApp Assistant, and set those Operating System and Language requirements on the **Profile Requirements** page. Then deploy this application as a Citrix profile. • **Option 2: Determine if Launch Conditions are Met**—Review the launch conditions listed in the table, and determine if the desktops in your enterprise meet those requirements. If all of the desktops meet those requirements, you can deploy this application as a Citrix profile. If the desktops do not meet those requirements, upgrade those desktops to meet these requirements, and then deploy this application as a Citrix profile.

Warning -9027: LockPermissions Table

The following table documents this message:

Table 10-115 • Warning -9027: LockPermissions Table

Category	Description
Type	Warning
Message:	`The conversion process does not support data in the MSI table 'LockPermissions'.`
Cause:	You encounter this warning when the Windows Installer package that you are converting includes a `LockPermissions` table. During the conversion process, the `LockPermissions` table is not read.
Resolution:	The **LockPermissions** table is used to secure individual portions of your application (files, registry keys, and created folders) in a locked-down environment. Citrix does not support permissions on files, registry keys, or created folders. You cannot modify permissions on any of the application's ACLs (access control lists). Because users will have full permissions when running this application in the isolation environment, this warning will not result in any problems.

Error -9028: MoveFile Table

The following table documents this message.

Table 10-116 • Error -9028: MoveFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'MoveFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a `MoveFile` table. During the conversion process, the `MoveFile` table is not read.
Resolution:	This `MoveFile` table contains a list of files to be moved or copied from a specified source directory to a specified destination directory. Because this table is not read, you need to do one of the following to resolve this issue: • **Option 1: Edit the Windows Installer Package**—Open the Windows Installer package in InstallShield Editor and modify it to eliminate the use of the **MoveFile** table by installing additional files in the specified directories. • **Option 2: Repackage the Application**—Use the Repackaging Wizard to repackage this application, and then build the Repackager project to generate a revised Windows Installer package. • **Option 3: Write a Pre-Launch Script**—Write a pre-launch script that performs the file moving operations identified in the **MoveFile** table upon application launch.

Error -9029: MsiDriverPackages Table

The following table documents this message:

Table 10-117 • Error -9029: MsiDriverPackages Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'MsiDriverPackages'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **MsiDriverPackages** table. During the conversion process, the **MsiDriverPackages** table is not read.

Table 10-117 • Error -9029: MsiDriverPackages Table

Category	Description
Resolution:	The **MsiDriverPackages** table includes one record for each driver package component in the application. Citrix XenApp does not support any type of driver. For example, when installing a printer, you can install the printer software within the isolation environment, but not the printer drivers. Therefore, to resolve this issue, you need to install any required drivers outside of the isolation environment on the user desktop machines.

Warning -9030: ODBCTranslator Table

The following table documents this message:

Table 10-118 • Warning -9030: ODBCTranslator Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'ODBCTranslator'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ODBCTranslator** table. During the conversion process, the **ODBCTranslator** table is not read.
Resolution:	The **ODBCTranslator** table lists the ODBC translators belonging to the installation. ODBC translators translate one form of raw data into another form that can be used with a specific database type. Ignoring the **ODBCTranslator** table should not prevent your application from functioning properly. However, if it does, you need to use Repackager to repackage the application.

Warning -9031: RemoveFile Table

The following table documents this message:

Table 10-119 • Warning -9031: RemoveFile Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveFile** table. During the conversion process, the **RemoveFile** table is not read. This warning is displayed only if the application installation removes files during install (not uninstall).

Table 10-119 • Warning -9031: RemoveFile Table

Category	Description
Resolution:	The **RemoveFile** table contains a list of files to be removed. If this file removal requirement is just a clean-up step, that does not impact the function of the application, you do not need to address this issue. However, if the existence of the files listed in the **RemoveFile** table prevents the application from functioning, you need to set the isolation option of the files to **Ignore** so that they are not visible to the isolation environment. The **Ignore** option directs the isolation environment to always look for the file on the system (ignoring the one inside the isolation environment).

Warning -9032: RemoveIniFile Table

The following table documents this message:

Table 10-120 • Warning -9032: RemoveIniFile Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveIniFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveIniFile** table. During the conversion process, the **RemoveIniFile** table is not read.
Resolution:	The **RemoveIniFile** table contains the information an application needs to delete from a `.ini` file. If the removal of this entry is necessary for the function of the application, you need to use Repackager to repackage the application.

Warning -9033: RemoveRegistry Table

The following table documents this message:

Table 10-121 • Warning -9033: RemoveRegistry Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveRegistry'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveRegistry** table. During the conversion process, the **RemoveRegistry** table is not read.

Table 10-121 • Warning -9033: RemoveRegistry Table

Category	Description
Resolution:	The **RemoveRegistry** table contains the registry information the application needs to delete from the system registry. If this removal requirement is just a clean-up step, that does not impact the function of the application, you do not need to address this issue. However, if the existence of the registry keys listed in the **RemoveRegistry** table prevents the application from functioning, you need to set the isolation option of the registry keys to **Ignore** so that they are not visible to the isolation environment. The **Ignore** option directs the isolation environment to always look for the registry key on the system (ignoring the one inside the isolation environment).

Error -9036: ISCEInstall Table

The following table documents this message:

Table 10-122 • Error -9036: ISCEInstall Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISCEInstall'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISCEInstall** table. During the conversion process, the **ISCEInstall** table is not read.
Resolution:	The **ISCEInstall** table is used to install Windows Store mobile apps. The conversion of mobile apps to Citrix XenApp profiles is not supported.

Error -9037: ISComPlusApplication Table

The following table documents this message:

Table 10-123 • Error -9037: ISComPlusApplication Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISComPlusApplication'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISComPlusApplication** table. During the conversion process, the **ISComPlusApplication** table is not read.

Table 10-123 • Error -9037: ISComPlusApplication Table

Category	Description
Resolution:	The **ISComPlusApplication** table contains information about COM+ applications. While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, this application cannot be deployed on Citrix XenApp.

Error -9038: ISPalmApp Table

The following table documents this message:

Table 10-124 • Error -9038: ISPalmApp Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISPalmApp'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISPalmApp** table. During the conversion process, the **ISPalmApp** table is not read.
Resolution:	The **ISPalmApp** table is used to install Palm mobile apps. The conversion of mobile apps to Citrix XenApp profiles is not supported.

Error -9039: ISSQLScriptFile Table

The following table documents this message:

Table 10-125 • Error -9039: ISSQLScriptFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISSQLScriptFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISSQLScriptFile** table. During the conversion process, the **ISSQLScriptFile** table is not read.

Table 10-125 • Error -9039: ISSQLScriptFile Table

Category	Description
Resolution:	The **ISSQLScriptFile** table lists SQL scripts. When a Windows Installer package is installed, it can run an SQL script to update a database. An application running as a Citrix profile cannot update a database. To resolve this issue, you need to update the database prior to using the converted Citrix profile using one of the following methods: • Use scripts to update the database manually. • Update it by running the Windows Installer installation on one of the machines in your network that has access to that database.

Error -9040: ISVRoot Table

The following table documents this message:

Table 10-126 • Error -9040: ISVRoot Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISVRoot'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISVRoot** table. During the conversion process, the **ISVRoot** table is not read.
Resolution:	The **ISVRoot** table installs a website. An application running as a Citrix profile in an isolation environment cannot create a website. Therefore, creating Citrix profiles for applications that create websites during installation is not supported.

Error -9041: ISXmlFile Table

The following table documents this message:

Table 10-127 • Error -9041: ISXmlFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISXmlFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISXmlFile** table. During the conversion process, the **ISXmlFile** table is not read.

Table 10-127 • Error -9041: ISXmlFile Table

Category	Description
Resolution:	The **ISXmlFile** table modifies XML files. If the modification of XML files is required in order for this application to operate properly, you need to use Repackager to repackage this application.

Error -9051: Package Decompression Canceled

The following table documents this message:

Table 10-128 • Error -9051: Package Decompression Canceled

Category	Description
Type:	Error
Message:	`Package decompression canceled by the user`
Cause:	The user cancelled the process of decompression of compressed MSI packages.

Error -9100: CreateInstance of Package Object Failed

The following table documents this message:

Table 10-129 • Error -9100: CreateInstance of Package Object Failed

Category	Description
Type:	Error
Message:	`CreateInstance of the Citrix package object failed.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9101: Create Operation of Package Object Failed

The following table documents this message:

Table 10-130 • Error -9101: Create Operation of Package Object Failed

Category	Description
Type:	Error

Table 10-130 • Error -9101: Create Operation of Package Object Failed

Category	Description
Message:	`Create operation on Citrix package object failed 'ObjectName'.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9102: Failed to Write Header Information

The following table documents this message:

Table 10-131 • Error -9102: Failed to Write Header Information

Category	Description
Type:	Error
Message:	`Failed to write package header information.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9103: Citrix Finalization Failed

The following table documents this message:

Table 10-132 • Error -9103: Citrix Finalization Failed

Category	Description
Type:	Error
Message:	`Citrix Finalization Failed`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9104: Citrix Save Failed

The following table documents this message:

Table 10-133 • Error -9104: Citrix Save Failed

Category	Description
Type:	Error
Message:	`Citrix Save Failed`
Cause:	Unexpected internal error. This error may sometimes occur when the profile is to be digitally signed.
Resolution:	Deselect the option to digitally sign the Citrix profile and then rebuild it.

Error -9105: Error Initializing Citrix Writer

The following table documents this message:

Table 10-134 • Error -9105: Error Initializing Citrix Writer

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix writer`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9106: Error Initializing Citrix Package

The following table documents this message:

Table 10-135 • Error -9106: Error Initializing Citrix Package

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix package`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9107: Error Writing Citrix File Entries

The following table documents this message:

Table 10-136 • Error -9107: Error Writing Citrix File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix file entries.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9108: Error Determining Source File Path

The following table documents this message:

Table 10-137 • Error -9108: Error Determining Source File Path

Category	Description
Type:	Error
Message:	`Unexpected error determining source file path for 'FileName'`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9109: Error Writing Citrix Folder Entries

The following table documents this message:

Table 10-138 • Error -9109: Error Writing Citrix Folder Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix folder entries`
Cause:	Unexpected internal error.

Table 10-138 • Error -9109: Error Writing Citrix Folder Entries

Category	Description
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9110: Error Writing Citrix Registry Entries

The following table documents this message:

Table 10-139 • Error -9110: Error Writing Citrix Registry Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix registry entries`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9113: Error Writing Citrix INI File Entries

The following table documents this message:

Table 10-140 • Error -9113: Error Writing Citrix INI File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix INI file entries`
Cause:	Unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9114: Error Writing Citrix Shortcuts

The following table documents this message:

Table 10-141 • Error -9114: Error Writing Citrix Shortcuts

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix shortcuts`
Cause:	A catastrophic error has occurred while writing shortcuts to the profile.
Resolution:	Verify that shortcuts point to a valid file. Try to narrow down issue by only keeping one shortcut and then try to rebuild.

Error -9115: Error Saving Citrix Profile

The following table documents this message:

Table 10-142 • Error -9115: Error Saving Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error saving Citrix profile`
Cause:	A catastrophic error has occurred while saving the Citrix profile.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9116: Error Creating Empty Citrix Profile

The following table documents this message:

Table 10-143 • Error -9116: Error Creating Empty Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error creating empty Citrix profile`
Cause:	AdminStudio is unable to create a new internal instance of a Citrix profile.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9117: Error Creating Intermediate Folder

The following table documents this message:

Table 10-144 • Error -9117: Error Creating Intermediate Folder

Category	Description
Type:	Error
Message:	`Unexpected error creating intermediate folder`
Cause:	AdminStudio is unable to create the intermediate folder used for the build. This error could occur if the user does not have permission to write to `C:\TMP`.
Resolution:	Obtain write access to `C:\TMP` and then rebuild the profile.

Error -9118: Error Initializing Citrix Profile

The following table documents this message:

Table 10-145 • Error -9118: Error Initializing Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix profile.`
Cause:	The initial values on the new profile could not be set.
Resolution:	Check the package name, description, version, and security settings for any possible causes.

Error -9119: Error Creating Default Target in Citrix Profile

The following table documents this message:

Table 10-146 • Error -9119: Error Creating Default Target in Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error creating default target in Citrix profile`
Cause:	Initial target in the new profile could not be created.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9120: Error Deleting File From Profile

The following table documents this message:

Table 10-147 • Error -9120: Error Deleting File From Profile

Category	Description
Type:	Error
Message:	`Unexpected error deleting file 'FileName' from profile`
Cause:	Specified file could not be deleted from profile.
Resolution:	Check to see if the file exists and if the user has access rights to the file. If that did not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9121: Failed to Copy File into Citrix Profile

The following table documents this message:

Table 10-148 • Error -9121: Failed to Copy File into Citrix Profile

Category	Description
Type:	Error
Message:	`Failed to copy file into Citrix profile. Error: 'Name' File: 'Name'`
Cause:	Specified file could not be copied into profile.
Resolution:	Check to see if the file exists and if the user has access rights to the file. Also, when this error occurred, you should have also received a return error code from Windows Installer. Look up that error code in the Windows Installer Help Library to determine the cause of the problem.

Error -9122: Target Does Not Exist in Citrix Profile

The following table documents this message:

Table 10-149 • Error -9122: Target Does Not Exist in Citrix Profile

Category	Description
Type:	Warning

Table 10-149 • Error -9122: Target Does Not Exist in Citrix Profile

Category	Description
Message:	`The target for shortcut 'ShortcutName' does not exist in the Citrix profile. Excluding shortcut.`
Cause:	Actual file that shortcut points to is not included in the package.
Resolution:	Exclude the shortcut by clearing the selection on the Citrix Assistant **Profile Shortcuts** page, and then rebuild the profile.

Error -9124: No Shortcuts Created for this Profile

The following table documents this message:

Table 10-150 • Error -9124: No Shortcuts Created for this Profile

Category	Description
Type:	Error
Message:	`No shortcuts were created in the XenApp profile during the conversion because none were detected in the source package.`
Cause:	A XenApp profile must include at least one valid shortcut.
Resolution:	Add a shortcut on the Citrix Assistant **Profile Shortcuts** page, and then rebuild the profile.

Error -9125: Error Writing Citrix File Type Associations

The following table documents this message:

Table 10-151 • Error -9125: Error Writing Citrix File Type Associations

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix file type associations`
Cause:	Unable to write file type associations.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9126: Failed to Sign Profile Using Certificate

The following table documents this message:

Table 10-152 • Error -9126: Failed to Sign Profile Using Certificate

Category	Description
Type:	Error
Message:	`Failed to sign the profile using the supplied certificate`
Cause:	The certificate that is being used in invalid.
Resolution:	Obtain a valid certificate and rebuild the profile.

Error -9127: Could Not Create Script Execution

The following table documents this message:

Table 10-153 • Error -9127: Could Not Create Script Execution

Category	Description
Type:	Error
Message:	`Could not create script execution for 'ScriptName'`
Cause:	The specified script contains invalid data.
Resolution:	On the Citrix Assistant **Build Settings** page, delete the script from the profile, re-add it, and then rebuild the profile. If you are still having problems, perform these additional investigational steps and then contact AdminStudio Technical Support.

Warning -9128: Duplicate Shortcut

The following table documents this message:

Table 10-154 • Warning -9128: Duplicate Shortcut

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile. Excluding duplicate shortcut.`

Table 10-154 • Warning -9128: Duplicate Shortcut

Category	Description
Cause:	There are multiple shortcuts defined in this profile that refer to different Start Menu locations or to other locations in the package.
Resolution:	These shortcuts are not needed. On the Citrix Assistant **Profile Shortcuts** page, unselect these shortcuts, and then rebuild the profile.

Warning -9129: Duplicate Shortcut Names

The following table documents this message:

Table 10-155 • Warning -9129: Duplicate Shortcut Names

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile, but with different command line parameters. A new unique shortcut 'NewShortcutName(1)' will be created in the profile.`
Cause:	There are two shortcuts defined in this profile that have the same name, even though they have different command line parameters.
Resolution:	On the Citrix Assistant **Profile Shortcuts** page, rename one of these shortcuts and then rebuild the profile.

Warning -9130: Duplicate Shortcut Targets

The following table documents this message:

Table 10-156 • Warning -9130: Duplicate Shortcut Targets

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile, but with different target. A new unique shortcut 'NewShortcutName(1)' will be created in the profile.`
Cause:	There are two shortcuts defined in this profile that have the same name, even though they have different targets.
Resolution:	On the Citrix Assistant **Profile Shortcuts** page, rename one of these shortcuts and then rebuild the profile.

Warning -9131: Unable to Resolve Installer Variable

The following table documents this message:

Table 10-157 • Warning -9131: Unable to Resolve Installer Variable

Category	Description
Type:	Warning
Message:	`Unable to resolve an installer variable in the string 'StringName'`
Cause:	Not all Windows Installer variables can be resolved at build time. This can result in errors if your application requires a specific value.
Resolution:	Repackage this application and rebuild the profile, or use a constant value in the Windows Installer package.

Warning -9132: 16 Color Shortcut Icon Not Found

The following table documents this message:

Table 10-158 • Warning -9132: 16 Color Shortcut Icon Not Found

Category	Description
Type:	Warning
Message:	`No 16 color icon found for 'ShortcutName' shortcut. Shortcut icon may look distorted when published.`
Cause:	The icon used for this shortcut does not contain a 16-color image. Since Citrix currently does not support images with a higher number of colors, this icon may look distorted when shown and published in Citrix XenApp.
Resolution:	You can modify the shortcut to use a different icon or add a 16-color image to the one currently used.

Warning -9133: Shortcut Icon Not Found

The following table documents this message:

Table 10-159 • Warning -9133: Shortcut Icon Not Found

Category	Description
Type:	Warning
Message:	`No icon found for 'ShortcutName' shortcut. Using generic Windows application icon instead.`

Table 10-159 • Warning -9133: Shortcut Icon Not Found

Category	Description
Cause:	If no icon can be loaded for this shortcut, the generic Windows application icon is used. This can happen if the file used is corrupt or if extracting an image from it is not possible.
Resolution:	Modify the shortcut to use a different icon.

Warning -9134: Failure to Extract Icon from Executable

The following table documents this message:

Table 10-160 • Warning -9134: Failure to Extract Icon from Executable

Category	Description
Type:	Warning
Message:	`Failed to extract icon from executable 'filename'. Make sure the executable is not corrupt.`
Cause:	A corrupt icon file may cause this warning.
Resolution:	Modify the shortcut to use a different icon.

Error -9135: Shortcut Target is 16-Bit

The following table documents this message:

Table 10-161 • Error -9135: Shortcut Target is 16-Bit

Category	Description
Type:	Error
Message:	`The target for shortcut 'ShortcutName' is 16-bit. This application may not function properly in the Citrix Isolation Environment.`
Cause:	The file this shortcut points to is a 16-bit application.
Resolution:	Replace file with a newer 32-bit version. Can also test and see if the application works properly in the Citrix environment.

Warning -9136: Some Files May Not Be Decompressed

The following table documents this message:

Table 10-162 • Warning -9136: Some Files May Not Be Decompressed

Category	Description
Type:	Warning
Message:	`Some files may not be decompressed from this package because it contains features with a default install level of 0.`
Cause:	When installing a compressed Windows Installer package, the build engine runs an administrative installation of it to decompress it. One limitation of an administrative installation is that it does not decompress a file if the feature it is contained in has a default install level of **0**. If there are any files in any components contained within those features, AdminStudio will generate an error when it attempts to copy those files into the Citrix profile, because they will not exist in the source location.
Resolution:	To resolve this issue, edit the Windows Installer package and set the default install level of that feature to something other than **0**.

Warning -9137: Destination Directory Cannot Be Found

The following table documents this message:

Table 10-163 • Warning -9137: Destination Directory Cannot Be Found

Category	Description
Type:	Warning
Message:	`The destination directory for the 'FileName' file cannot be found. You should consider Repackaging this application before proceeding with the conversion process.`
Cause:	This is an internal error.
Resolution:	Contact AdminStudio Technical Support.

Warning -9138: Ignoring a DuplicateFile Table Entry

The following table documents this message:

Table 10-164 • Warning -9138: Ignoring a DuplicateFile Table Entry

Category	Description
Type:	Warning
Message:	`Ignoring a DuplicateFile table entry because unable to resolve the property used for the DestFolder: 'INVALIDPATH'`
Cause:	You might encounter this error when the Windows Installer package that you are converting includes one or more entries in the `DuplicateFile` table, and a property that is used in the DestFolder column for one of those entries in the `DuplicateFile` table cannot be resolved. For example, if the destination for a duplicate file is set by a custom action, that destination cannot be resolved during the conversion.
Resolution:	The `DuplicateFile` table contains a list of files that need to be duplicated during installation, either to a different directory than the original file or to the same directory but with a different name. Because a destination in this table cannot be resolved, you need to do one of the following to resolve this issue: • **Option 1: Edit the Windows Installer Package**—Open the Windows Installer package in InstallShield and modify it to eliminate the use of the problematic entry in the `DuplicateFile` table by including any additional copies of that file. • **Option 2: Repackage the Application**—Use the Repackaging Wizard to repackage this application, and then build the Repackager project to generate a revised Windows Installer package. • **Option 3: Write a Pre-Launch Script**—Write a pre-launch script that—upon application launch—performs the file copy operations for the problematic entry in the `DuplicateFile` table.

Error -9139: 64-Bit Executables (XenApp)

The following table documents this message:

Table 10-165 • Error -9139: 64-Bit Executables (XenApp)

Category	Description
Type:	Error
Message:	`The target for shortcut '[SHORTCUT_NAME]' is a 64-bit executable. XenApp does not support 64-bit applications.`
Cause:	XenApp does not support 64-bit applications.

Table 10-165 • Error -9139: 64-Bit Executables (XenApp) (cont.)

Category	Description
Resolution:	This 64-bit application cannot be converted to XenApp format.

Error -9200: ThinApp Must Be Installed

The following table documents this message:

Table 10-166 • Error -9200: ThinApp Must Be Installed

Category	Description
Type:	Error
Message:	`A licensed or demo version of ThinApp must be installed on this machine in order to successfully build ThinApp applications. (www.vmware.com)`
Cause:	ThinApp is not installed.
Resolution:	Install ThinApp.

Warning -9201: Extension for Shortcut Files Must Be ".exe"

The following table documents this message:

Table 10-167 • Warning -9201: Extension for Shortcut Files Must Be ".exe"

Category	Description
Type:	Warning
Message:	`The extension for the target for shortcut 'ShortcutName' is not '.exe'. Excluding shortcut.`
Cause:	Shortcuts that do not have a filename extension of `.exe` are excluded.
Resolution:	No action is required.

Error -9202: No Applications Were Created

The following table documents this message:

Table 10-168 • Error -9202: No Applications Were Created

Category	Description
Type:	Error
Message:	`No applications were created during the ThinApp conversion because no shortcuts were detected in the source package.`
Cause:	Either the Windows Installer package had no shortcuts or all shortcuts were excluded.
Resolution:	Make sure that the source Windows Installer `.msi` package has at least one shortcut to an `.exe` file.

Error -9203: ThinApp Tool is Missing

The following table documents this message:

Table 10-169 • Error -9203: ThinApp Tool is Missing

Category	Description
Type:	Error
Message:	`ThinApp: 'ToolName' was not found`
Cause:	One of the ThinApp tools required to build a ThinApp application was not found.
Resolution:	Reinstall ThinApp.

Error -9204: Duplicate Shortcut

The following table documents this message:

Table 10-170 • Error -9204: Duplicate Shortcut

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists. Excluding duplicate shortcut.`
Cause:	The source package has two shortcuts that both point to the same `.exe` target.
Resolution:	No action is required.

Error -9205: Identically-Named Shortcut Already Exists, But With Different Parameters

The following table documents this message:

Table 10-171 • Error -9205: Identically-Named Shortcut Already Exists, But With Different Command Line Parameters

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists, but with different command line parameters. A new, unique shortcut will be created.`
Cause:	Two shortcuts in the package differed in arguments only.
Resolution:	No action is required.

Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target

The following table documents this message:

Table 10-172 • Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists, but with a different target. A new, unique shortcut will be created.`
Cause:	Two shortcuts differed in the target pointed to only.
Resolution:	No action is required.

Error -9207: Error During Build Process (vregtool.exe)

The following table documents this message:

Table 10-173 • Error -9207: Error During Build Process (vregtool.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (vregtool.exe).`
Cause:	An unexpected error occurred while running the vregtool.exe step of the ThinApp build process.

Table 10-173 • Error -9207: Error During Build Process (vregtool.exe)

Category	Description
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders in the build folder hierarchy are locked.

Error -9208: Error Occurred During Build Process (vftool.exe)

The following table documents this message:

Table 10-174 • Error -9208: Error Occurred During Build Process (vftool.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (vftool.exe)`
Cause:	An unexpected error occurred while running the `vftool.exe` step of the ThinApp build process.
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders in the build folder hierarchy are locked.

Error -9209: Error Occurred During ThinApp Build Process (tlink.exe)

The following table documents this message:

Table 10-175 • Error -9209: Error Occurred During ThinApp Build Process (tlink.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (tlink.exe)`
Cause:	An unexpected error occurred while running the tlink.exe step of the ThinApp build process.
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders of the build folder hierarchy are locked.

Error -9210: 64-Bit Executables (ThinApp)

The following table documents this message:

Table 10-176 • Error -9210: 64-Bit Executables (ThinApp)

Category	Description
Type:	Error
Message:	`The target for shortcut '[SHORTCUT_NAME]' is a 64-bit executable. ThinApp does not support 64-bit applications.`
Cause:	ThinApp does not support 64-bit applications.
Resolution:	This 64-bit application cannot be converted to ThinApp format.

Error -9300: Unhandled Exception During AdviseFile Operation

The following table documents this message:

Table 10-177 • Error -9300: Unhandled Exception During AdviseFile Operation

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the AdviseFile operation for rule 'RuleName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9301: Unhandled Exception During AdviseRegistry Operation

The following table documents this message:

Table 10-178 • Error -9301: Unhandled Exception During AdviseRegistry Operation

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the AdviseRegistry operation for rule 'RuleName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9302: Unhandled Exception During Command Action

The following table documents this message:

Table 10-179 • Error -9302: Unhandled Exception During Command Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the command action with the description 'CommandActionName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9303: Unhandled Exception During Alter File Action

The following table documents this message:

Table 10-180 • Error -9303: Unhandled Exception During Alter File Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the alter file action with the description 'FileName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9304: Unhandled Exception During Alter Registry Action

The following table documents this message:

Table 10-181 • Error -9304: Unhandled Exception During Alter Registry Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the alter registry action with the description 'RegistryName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9305: Unhandled Exception During Create Action

The following table documents this message:

Table 10-182 • Error -9305: Unhandled Exception During Create Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the create action with the description 'CreateName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9306: Unhandled Exception During Execution of Rules Engine

The following table documents this message:

Table 10-183 • Error -9306: Unhandled Exception During Execution of Rules Engine

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the execution of the rules engine.`
Resolution:	Contact AdminStudio Technical Support.

Error -9401: Error Initializing App-V Writer

The following table documents this message:

Table 10-184 • Error -9401: Error Initializing App-V Writer

Category	Description
Type:	Error
Message:	`Unexpected error initializing App-V writer.`
Resolution:	Contact AdminStudio Technical Support.

Error -9402: Error Initializing App-V Package

The following table documents this message:

Table 10-185 • Error -9402: Error Initializing App-V Package

Category	Description
Type:	Error
Message:	`Unexpected error initializing App-V package.`
Resolution:	Contact AdminStudio Technical Support.

Error -9403: Error Writing App-V File Entries

The following table documents this message:

Table 10-186 • Error -9403: Error Writing App-V File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V file entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9404: Error Writing App-V Folder Entries

The following table documents this message:

Table 10-187 • Error -9404: Error Writing App-V Folder Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V folder entries`
Resolution:	Contact AdminStudio Technical Support.

Error -9405: Error Writing App-V Registry Entries

The following table documents this message:

Table 10-188 • Error -9405: Error Writing App-V Registry Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V registry entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9406: Error Writing App-V INI File Entries

The following table documents this message:

Table 10-189 • Error -9406: Error Writing App-V INI File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V INI file entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9407: Error Writing App-V Shortcuts

The following table documents this message:

Table 10-190 • Error -9407: Error Writing App-V Shortcuts

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V shortcuts.`
Resolution:	Contact AdminStudio Technical Support.

Error -9408: Error Writing App-V File Type Data

The following table documents this message:

Table 10-191 • Error -9408: Error Writing App-V File Type Data

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V file type data.`
Resolution:	Contact AdminStudio Technical Support.

Error -9409: Error Saving App-V Data

The following table documents this message:

Table 10-192 • Error -9409: Error Saving App-V Data

Category	Description
Type:	Error
Message:	`Unexpected error saving App-V data.`
Resolution:	Contact AdminStudio Technical Support.

Error -9410: Error Determining Source File Path

The following table documents this message:

Table 10-193 • Error -9410: Error Determining Source File Path

Category	Description
Type:	Error
Message:	`Unexpected error determining source file path for 'FileName'.`
Cause:	The installation location of a file, which is determined by some run time property, cannot be determined by the App-V virtual converter.
Resolution:	Locate the file in InstallShield and provide a known directory.

Error -9411: OSD File Template Could Not Be Extracted

The following table documents this message:

Table 10-194 • Error -9411: OSD File Template Could Not Be Extracted

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file template could not be extracted. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Error -9412: OSD File Could Not Be Saved

The following table documents this message:

Table 10-195 • Error -9412: OSD File Could Not Be Saved

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file could not be saved. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Error -9413: App-V OSD Save

The following table documents this message:

Table 10-196 • Error -4313: App-V OSD Save

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file could not be saved. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application

The following table documents this message:

Table 10-197 • Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application

Category	Description
Type:	Warning
Message:	`A local App-V application was specified as a dependency of the primary application. The primary application may not run correctly if it is relocated to a different location.`
Cause:	The user specified a dependent App-V application that is either on the local drive or on a mapped drive. This is determined by examining the HREF attribute of the CODEBASE tag in the dependency application's OSD file.
Resolution:	Dependency applications should be referenced by a portable mechanism using either a non-FILE protocol or by using a network URL.

Error -9415: Dependency Application Was Not Found

The following table documents this message:

Table 10-198 • Error -9415: Dependency Application Was Not Found

Category	Description
Type:	Error
Message:	`Dependency application was not found: 'ApplicationName'.`
Cause:	A specified App-V dependency application file was not found.
Resolution:	Check the path of the specified App-V dependency application.

Warning -9416: Invalid Primary Application Directory

The following table documents this message:

Table 10-199 • Warning -9416: Invalid Primary Application Directory

Category	Description
Type:	Error
Message:	`The specified Primary Application Directory, 'DirectoryName', does not exist.`

Table 10-199 • Warning -9416: Invalid Primary Application Directory

Category	Description
Cause:	This may be caused if the directories specified in the Windows Installer package have changed after a valid Primary Application Directory was specified.
Resolution:	Specify a valid Primary Application Directory using the supplied browse folder in InstallShield.

Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value

The following table documents this message:

Table 10-200 • Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value

Category	Description
Type:	Error
Message:	`Dependency application OSD file contains an invalid value for the HREF field of the CODEBASE tag: 'HREF_Field_Value'`
Cause:	The CODEBASE tag of the dependency application's OSD file may have an empty or non-existent HREF attribute.
Resolution:	Make sure that the CODEBASE tag of the dependency application's OSD file has a valid HREF attribute.

Error -9418: Error While Privatizing Side-By-Side Assemblies

The following table documents this message:

Table 10-201 • Error -9418: Error While Privatizing Side-By-Side Assemblies

Category	Description
Type:	Error
Message:	`An error occurred while privatizing Side-By-Side assemblies.`
Cause:	When converting to an App-V package, files installed to the win32 Sxs assembly cache need to be privatized so that the App-V runtime can find them. This error occurs if there was an unexpected failure in that process.
Resolution:	Contact AdminStudio Technical Support.

Error -9419: Error Inserting Watermark

The following table documents this message:

Table 10-202 • Error -9419: Error Inserting Watermark

Category	Description
Type:	Error
Message:	`An error has occurred inserting the evaluation watermark into the App-V Package.`
Resolution:	Contact AdminStudio Technical Support.

Error -9420: Error During App-V Package Upgrade

The following table documents this message:

Table 10-203 • Error -9420: Error During App-V Package Upgrade

Category	Description
Type:	Error
Message:	`An error occurred while configuring App-V package upgrade.`
Cause:	This is a general failure to configure the new App-V package that is being built to be an upgrade for the previous one.
Resolution:	Verify that the previous App-V package is present and accessible at the location specified. Verify that the previous App-V package is not corrupt.

Warning -9421: 16-Bit Application

The following table documents this message:

Table 10-204 • Warning -9421: 16-Bit Application

Category	Description
Type:	Warning
Message:	`The target for shortcut '[SHORTCUT_NAME]' is 16-bit. This application will not function in 64-bit environments.`
Cause:	This warning helps to identify 16-bit applications. This is important for enterprises looking to deploy to 64-bit environments.

Table 10-204 • Warning -9421: 16-Bit Application

Category	Description
Resolution:	This warning can be ignored if the application will not be deployed to a 64-bit environment. Otherwise, it will be necessary to get a newer version of this application that is 32 or 64-bit.

Error -9422: Package Cannot Be Opened

The following table documents this message:

Table 10-205 • Error -9422: Package Cannot Be Opened

Category	Description
Type:	Error
Message:	`The previous package '[SFT_FILE_NAME]' could not be opened. Please verify that the package is a valid SFT file.`
Cause:	This error indicates that there was a problem reading information from the previous .sft file which is necessary to configure an upgrade. This could mean that the previous .sft file is corrupt, missing, or is not really an .sft file.
Resolution:	Verify that the file is present and accessible. Verify that the file is not corrupt. One way this could be done is by trying to deploy the App-V package on a machine with the App-V client.

Warning -9423: No Shortcuts Detected

The following table documents this message:

Table 10-206 • Warning -9423: No Shortcuts Detected

Category	Description
Type	Warning

Message	For App-V packages:
	`No shortcuts were detected in the source package. This can be a valid scenario for packages with file type association entry points and/or those that are meant to be dependencies of other App-V packages through the use of Dynamic Suite Composition functionality.`
	For XenApp and ThinApp packages:
	`This package contains no shortcuts. Shortcuts are necessary to define the entry point into the virtual application.`

Table 10-206 • Warning -9423: No Shortcuts Detected

Category	Description
Cause	Package contains no shortcuts. • **App-V**—For conversion to App-V packages, this issue is acceptable in some scenarios, such as packages which provide dependencies to others which dynamically suite it. However, if this package merely provides a plug-in to another application, it must contain a shortcut to launch that application in this package's virtual context. • **ThinApp and XenApp**—For conversion to ThinApp and XenApp formats, shortcuts are necessary to define the entry point into the virtual application.
Resolution:	One potential resolution to this issue is to use InstallShield Editor to add shortcut(s) to the Windows Installer package.

Error -9424: Windows 8 or Windows 2012 OS Required

The following table documents this message:

Table 10-207 • Error -9424: Windows 8 or Windows 2012 OS Required

Category	Description
Type	Error
Message	`Skipping conversion to Microsoft App-V version 5 because it is only supported on Windows 8 or Windows 2012 operating systems.`
Cause	Conversion of packages to App-V 5.x format using AdminStudio is only supported on Windows 8 or Windows 2012 operating systems.
Resolution:	Install AdminStudio on a machine with a Windows 8 or Windows 2012 operating system and try the conversion again.

Warning -9500: Shortcut Missing

The following table documents this message:

Table 10-208 • Warning -9500: Shortcut Missing

Category	Description
Type:	Warning
Message:	`The target for shortcut 'FileName' does not exist. Excluding shortcut.`
Cause:	The target file of a shortcut in the project does not exist.

Table 10-208 • Warning -9500: Shortcut Missing

Category	Description
Resolution:	Repackage this application and then rebuild the virtual package.

Error -9600: Error Initializing Symantec Writer

The following table documents this message:

Table 10-209 • Error -9600: Error Initializing Symantec Writer

Category	Description
Type:	Error
Message:	`Unexpected error initializing Symantec writer.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9601: Error Writing Symantec Folder Entries

The following table documents this message:

Table 10-210 • Error -9601: Error Writing Symantec Folder Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Symantec folder entries.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9602: Error Writing Symantec Shortcuts

The following table documents this message:

Table 10-211 • Error -9602: Error Writing Symantec Shortcuts

Category	Description
Type:	Error
Message:	`Unexpected error writing Symantec shortcuts.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9603: Error Creating Target File for Symantec Package

The following table documents this message:

Table 10-212 • Error -9603: XXXX

Category	Description
Type:	Error
Message:	`Unexpected error creating target file for Symantec package.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9604: Error Writing Symantec File Entries

The following table documents this message:

Table 10-213 • Error -9604: Error Writing Symantec File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Symantec file entries.`
Cause:	Unexpected internal error.

Table 10-213 • Error -9604: Error Writing Symantec File Entries

Category	Description
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9605: Error Writing Symantec Registry Entries

The following table documents this message:

Table 10-214 • Error -9605: Error Writing Symantec Registry Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Symantec registry entries.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -9606: Error Writing Symantec INI File Entries

The following table documents this message:

Table 10-215 • Error -9606: Error Writing Symantec INI File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Symantec INI file entries.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps, as described in Steps to Take Before Calling Technical Support, and contact AdminStudio Technical Support.

Error -10000: Process Cancelled By User

The following table documents this message:

Table 10-216 • Error -10000: Process Cancelled By User

Category	Description
Type:	Error
Message:	`Process cancelled by user.`
Cause:	User clicked **Cancel** to cancel the profile conversion process.

Warning -10001: Suite File Missing

The following table documents this message:

Table 10-217 • Warning -10001: Suite File Missing

Category	Description
Type:	Warning
Message:	`The suite MSI file 'FileName' is missing and will be excluded from the conversion.`
Cause:	An MSI file that is part of a suite conversion was not found.
Resolution:	Make sure the input file for the suite conversion process exists.

Warning -10002: Suite File is Duplicate

The following table documents this message:

Table 10-218 • Warning -10002: Suite File is Duplicate

Category	Description
Type:	Warning
Message:	`The suite MSI file 'FileName' appears to be the same as the main MSI file and we will exclude this file from the conversion process.`
Cause:	A suite conversion was attempted where the main Windows Installer file (.msi) and one of the additional Windows Installer files specified were the same.
Resolution:	Specify unique Windows Installer files as part of the suite conversion process.

Warning -10003: Application File Missing

The following table documents this message:

Table 10-219 • Warning -10003: Application File Missing

Category	Description
Type:	Error
Message:	`Application file not found 'ApplicationName'`
Cause:	A file referenced by the installation was not found by the App-V virtual converter. It is likely that the file reference is broken in the installation.
Resolution:	Use InstallShield to locate the file in the installation and either fix the link or delete it.

Warning -10004: INI File Missing

The following table documents this message:

Table 10-220 • Warning: -10004: INI File Missing

Category	Description
Type:	Error
Message:	`INI file not found 'INI_File_Name'.`
Resolution:	Contact AdminStudio Technical Support.

Fix 11000: Excluding TCPIP Registry Entries

The following table documents this message:

Table 10-221 • Fix 11000: Excluding TCPIP Registry Entries

Category	Description
Type:	Fix
Message:	`Excluding TCPIP registry entries from the Citrix profile.`
Action:	Automated Application Converter will exclude all TCPIP registry entries from the Citrix profile.

Fatal Error 11001: Fail on VMware

The following table documents this message:

Table 10-222 • Fatal Error 11001: Fail on VMware

Category	Description
Type:	Fatal
Message:	`VMware cannot be virtualized.`
Cause:	Conversion will fail when the application being virtualized is VMware.
Action:	This error message is displayed: `VMware cannot be virtualized.`

Warning 11003: Control Panel Applet - Citrix

The following table documents this message:

Table 10-223 • Warning 11003: Control Panel Applet - Citrix

Category	Description
Type:	Warning
Message:	`The Control Panel Applet [AppletName] will not appear in Control Panel.`
Action:	Automated Application Converter will display a warning when the application contains a control panel applet.

Fix 11004: Control Panel Applet - ThinApp

The following table documents this message:

Table 10-224 • Fix 11004: Control Panel Applet - ThinApp

Category	Description
Type:	Fix
Message:	`Generating shortcut for the Control Panel Applet located at 'DirectoryPath'`
Action:	Automated Application Converter will create a default shortcut for ThinApp Control Panel applets.

Fatal Error 11005: QuickTime 7.4.1 Causes Fatal Error

The following table documents this message:

Table 10-225 • Error 11005: QuickTime 7.4.1 Causes Fatal Error

Category	Description
Type:	Fatal Error
Message:	`QuickTime 7.4.1 is known to have errors when running from a virtual package. Use QuickTime 7.4.5 instead.`
Cause:	QuickTime 7.4.1 cannot be virtualized correctly.
Resolution:	Obtain QuickTime 7.4.5 and repeat the conversion process.

Fix 11006: Adobe Distiller Exclude AdobePDFSettings

The following table documents this message:

Table 10-226 • Fix 11006: Adobe Distiller Exclude AdobePDFSettings

Category	Description
Type:	Fix
Message:	`Excluding the registry key Software\Adobe\Acrobat Distiller\AdobePDFSettings. Adobe Distiller will recreate these settings on first use.`
Action:	Automated Application Converter will exclude the AdobePDFSettings registry settings.

Fix 11007: Exclude URL Shortcut

The following table documents this message:

Table 10-227 • Fix 11007: Adobe Distiller Exclude AdobePDFSettings

Category	Description
Type:	Fix
Message:	`Excluding shortcut to .URL file. App-V does not launch these files properly.`
Action:	Automated Application Converter will exclude the shortcut to the .URL file.

Steps to Take Before Calling Technical Support

Before contacting AdminStudio Technical Support, perform the following steps to attempt to clearly identify the problem you are having:

- **Check package**—To determine if this error is caused by a problem with the specific package you are converting, try to build a virtual package of a simple package that contains only one file.
- **Check machine and OS**—To determine if this error is caused by a configuration on a particular machine or operating system, attempt to build this virtual package on another machine or operating system.
- **Check prerequisites**—Check to make sure that the machine where you are performing the conversion has all of the required prerequisite software installed:
  - **App-V**—See Comparison of the App-V 5.0 Conversion Methods.
  - **VMware ThinApp**—See Prerequisites for Building a ThinApp Application.
- **Check individual files**—To determine if this is error limited to a specific item, find out if removing or excluding a particular item will build error free.

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a Citrix profile. Therefore, some additional pre- or post-conversion actions must be taken in order for the application profile to run on Citrix XenApp.

One action you could take to try to include ignored features in a Citrix profile is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a Citrix profile.

The following table lists the application features which require additional, manual conversion steps:

Table 10-228 • Application Features Ignored During Profile Conversion

Windows Installer Feature	Manual Conversion Steps
User-Defined Custom Actions	When converting a Windows Installer package to a Citrix profile, all custom actions are ignored. For user-defined custom actions, a warning message is generated. Any modifications to a target machine that a custom action in this Windows Installer package may create will not be propagated into the Citrix profile. The resolution that you should perform depends upon the purpose of the custom action: ● If the custom action merely automatically enters a value or makes some other kind of minor modification, you can ignore this warning. ● If the custom action does something which could change the behavior of the installation (such as setting a Property), you may need to resolve this issue. To resolve this issue, first attempt to launch the converted package on Citrix XenApp. If you receive any application errors, you need to repackage this application, by performing the following steps. ***To successfully convert a package with user-defined custom actions:*** **1.** Use the Repackaging Wizard to repackage this application.The Repackaging Wizard monitors system changes as an application is installed, and then that data is converted into a Repackager project. **2.** Build the Repackager project to generate a revised Windows Installer package. This new Windows Installer package does not contain any custom actions, but (as a result of being repackaged) it will include the functionality performed by the original custom action.
Services	Citrix XenApp does not support any type of services. Therefore, to resolve this issue, you need to install any required services outside of the isolation environment on the user desktop machines. ***To successfully convert a package with services:*** **1.** If you have an application and a service bundled in the same Windows Installer package, you need to create a separate Windows Installer package containing just the service. **2.** Install the service on each of the user desktop machines. The Citrix profile of this application should now be able to run in an isolation environment on machines that already have the service installed.

Table 10-228 • Application Features Ignored During Profile Conversion

Windows Installer Feature	Manual Conversion Steps
COM+	While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, an application that contains COM+ services cannot be deployed on Citrix XenApp.

11

Using the MSIX Editor

MSIX Editor is a tool that enables you to edit MSIX packages, create Modification packages and perform various other tasks on MSIX Packages.

This user documentation contains information about the functionality and features of MSIX Editor, and is presented in the following sections:

Table 11-1 • AdminStudio MSIX Editor Help Library

Section	Description
About MSIX Editor	Introduces some basic concepts of MSIX package format and various functionalities of the tool.
Getting Started with the MSIX Editor	Contains information to help you become familiar with the MSIX Editor, begin editing a virtual package, and customize the Virtual Package Editor user interface.
Editing MSIX Packages	Explains how to edit MSIX packages and guides you through every step of the process.
Fix Runtime Issues of an MSIX Package	Explains how to identify and fix run time issues of your MSIX package.
Create Modification Packages	Explains how to create modification packages.
MSIX Editor Reference	Contains information to help you become familiar with the MSIX Editor properties.

About MSIX Editor

MSIX is a Windows application package format that provides a modern packaging experience for all Windows apps. AdminStudio includes MSIX Editor, which you can use to edit MSIX packages.

MSIX Editor is a tool which enables you to perform the following tasks on the MSIX packages:

- Open MSIX and Modification packages and get insights into the package.
- Make your desired changes to MSIX packages.
- Save the changes made on a MSIX package as a modification package.
- Digitally sign MSIX and modification packages.
- Trace MSIX packages to identify run time issues.
- Apply Package Support Framework fixups to fix run time issues.

MSIX Package Format

MSIX is a Windows app package format that combines the best features of MSI, .appx, App-V, and ClickOnce to provide a modern and reliable packaging experience. The MSIX package format preserves the functionality of existing app packages and/or install files in addition to enabling new, modern packaging and deployment features to Win32, WPF, and Windows Forms apps. Apps that are packaged using MSIX run in a lightweight app container. The MSIX app process and its child processes run inside the container and are isolated using file system and registry virtualization. All MSIX apps can read the global registry. An MSIX app writes to its own virtual registry and application data folder, and this data will be deleted completely thus making the uninstallation of the app clean with no footprints of the app left on the machine.

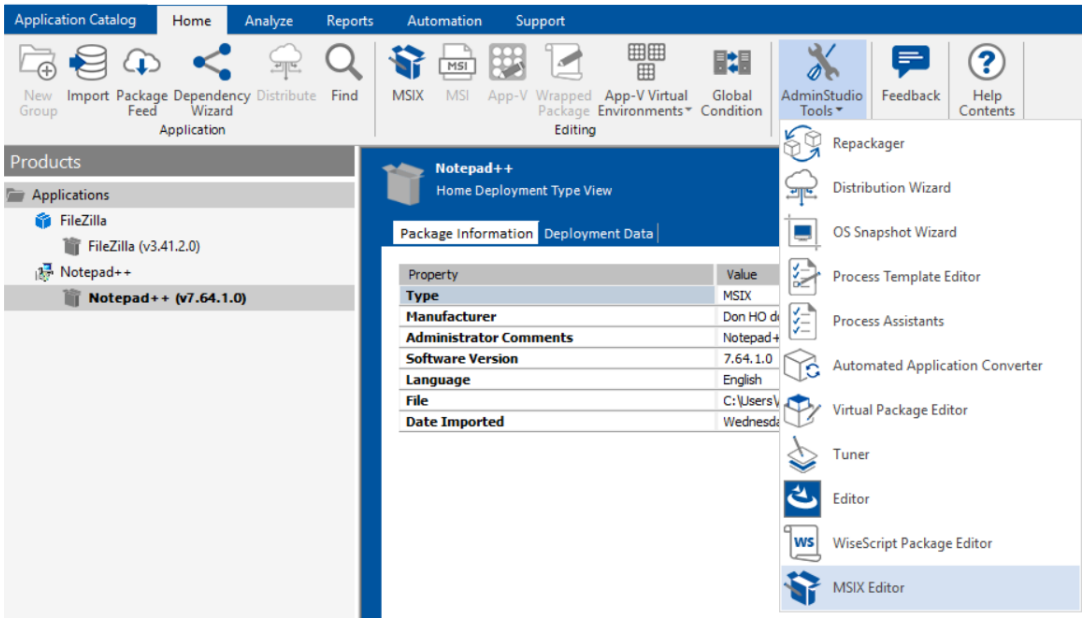
Getting Started with the MSIX Editor

The MSIX Editor provides powerful features that make editing MSIX packages and creating Modification packages easy. This section of the documentation contains information to help you become familiar with the MSIX Editor, its basic functionalities and user interface.

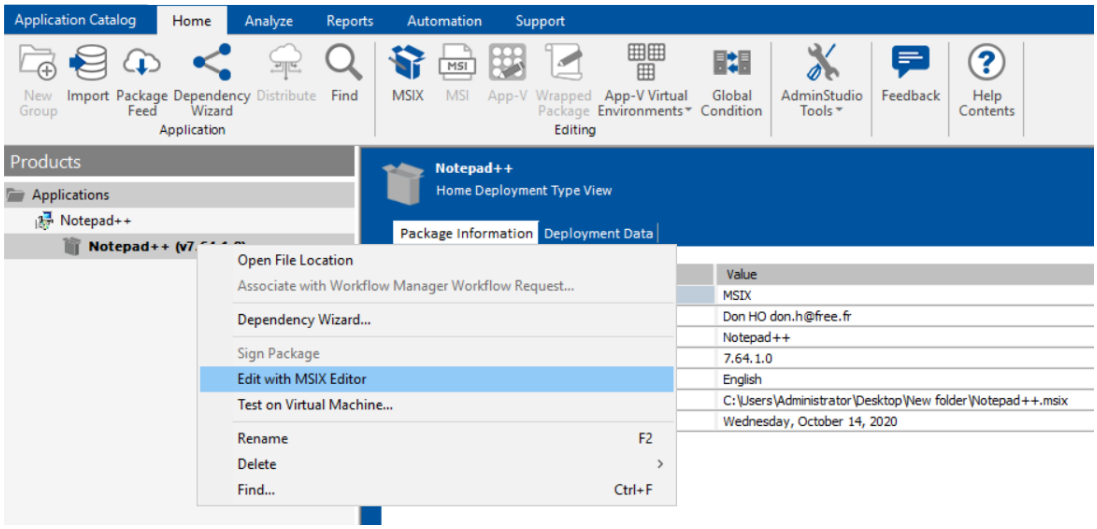
Launching MSIX Editor

To launch MSIX Editor, do one of the following:

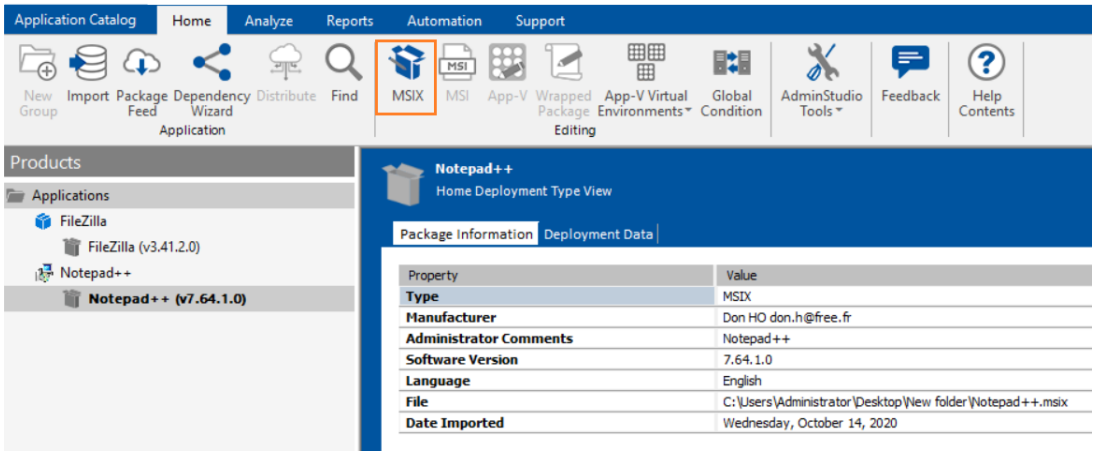
- In the **Home** tab of Application Manager, select **MSIX Editor** from the **AdminStudio Tools** drop down to launch MSIX Editor.



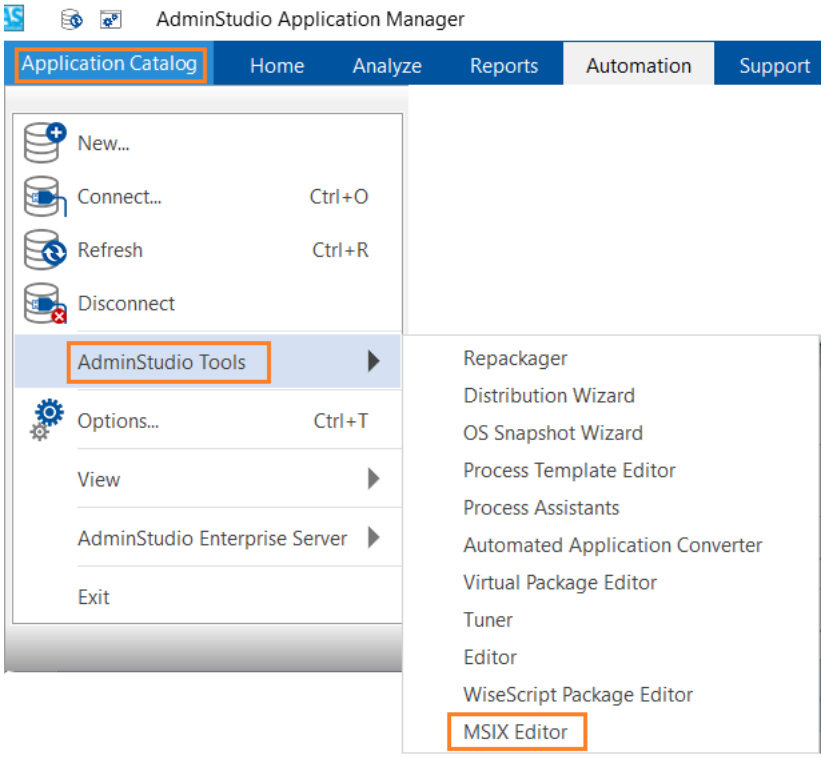
- In the **Home** tab of Application Manager, select an MSIX package in the Applications tree in the left pane, and then click **Edit with MSIX Editor** from the right click menu options to open the selected package in MSIX Editor.



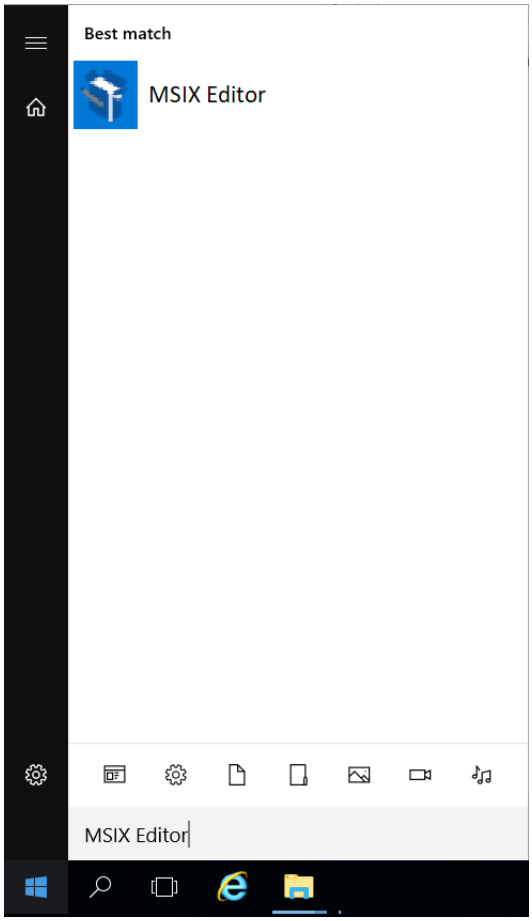
- In the **Home** tab of Application Manager, select an MSIX package in the Applications tree in the left pane and click on the **MSIX** button in the ribbon to open the selected package in MSIX Editor.



- From the **Applications Catalog** menu, click on **AdminStudio Tools**, and then select **MSIX Editor** from the list of tools to launch MSIX Editor.



- Type MSIX Editor in the Windows **Start** Menu and click on **MSIX Editor** to launch MSIX Editor.



Opening an MSIX Package

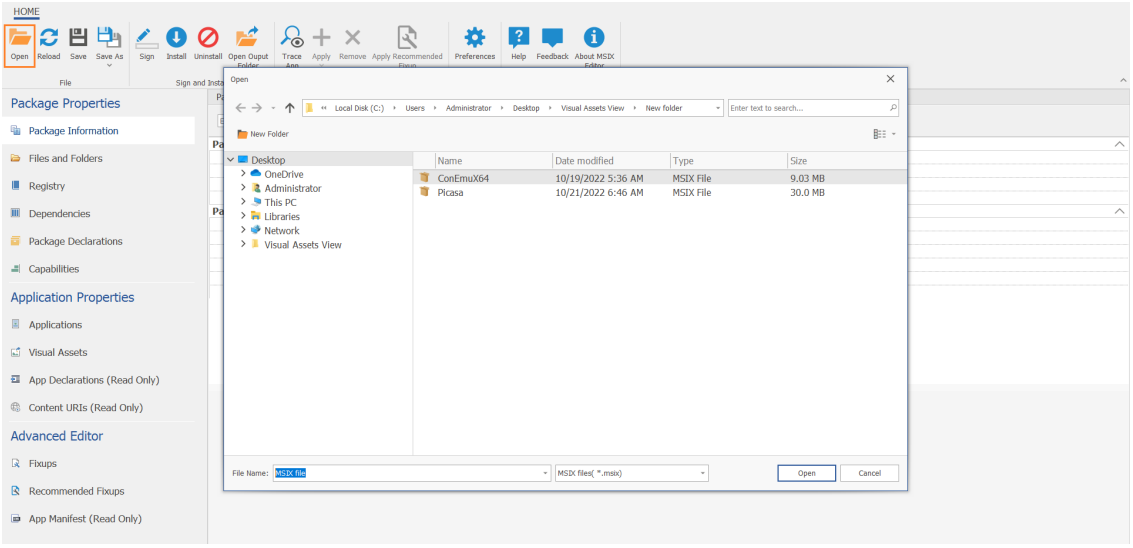
To open an MSIX package in the MSIX Editor, perform the following steps:

Task ***To open an MSIX package in the MSIX Editor:***

1. In the MSIX Editor, click the **Open** button in the ribbon.

   A dialog box appears, browse the desired MSIX package you want to edit, and then click the **Open** button.

   The Selected package details appears in the MSIX Editor.



Note • *In the windows title bar of the MSIX Editor, you can see the File Path when ever you* ***Open*** *or* ***Save****, or* ***Save As*** *the package.*

Working with MSIX Editor Interface

The MSIX Editor's graphical user interface has various UI elements. This section includes topics that explain how to perform basic tasks and the use of these elements.

MSIX Editor Ribbon

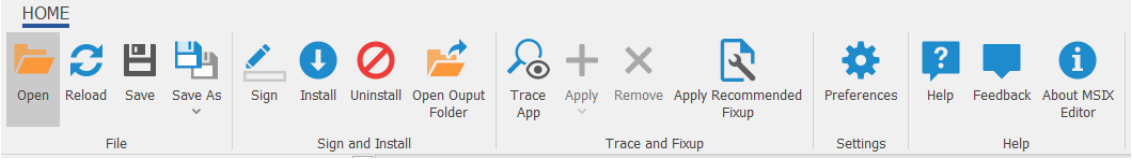
The main tasks that you perform using MSIX Editor involves clicking one of the following buttons on the ribbon:

Table 11-2 • MSIX Editor Ribbon

Button	Description
Open	Click to open an MSIX package and Modification package.
Reload	Click to discard all the changes made to the package and revert back to the original content of the package.
Save	Click to save the open package with the changes. ***Note*** • *The changes will take effect only after you click* ***Save*** *or* ***Save As****.*

Table 11-2 • MSIX Editor Ribbon

Button	Description
Save As	Click to save the changes to the open package as: • A new package • A modification package. For more information, see Saving as Modification Package.
Sign	Click to sign the package with the digital certificate specified on the Preferences dialog box.
Install	Click to install the package that is currently open in the MSIX Editor. ***Note*** *• MSIX packages must be digitally signed before they can be installed.*
Uninstall	Click to uninstall the open package if it is already installed.
Trace App	Click to identify the run times issues of the MSIX Package. Tracing will perform the following actions: • Signs the package with the signing option specified in **Preferences**. • Installs the package. • Starts tracing upon successful installation. ***Note*** *• You must perform functional behavior testing of the package to identify any run time issues while tracing is going on. Trace logs can be seen in the Output window.*
Apply	Click to add various fixups to an MSIX package. For more information, see Applying Fixups.
Remove	Click to remove the selected fixup.
Apply Recommended Fixup	Click to fix the run time issues automatically which were identified during the trace. For more information, see Applying Recommended Fixup.
Open Output Folder	Click to open the Output Folder to view the saved packages.
Preferences	Click to see various properties to be configured for MSIX Editor. For more information, see **Preferences**.
Feedback	Click to get redirected to the AdminStudio Forum in Flexera Community to provide your feedback.
About MSIX Editor	Click to view version and other information about MSIX Editor.



Preferences

Preferences, which appears when you click on the **Preferences** button on the MSIX Editor ribbon, you can configure various properties required by the MSIX Editor. The properties are grouped as mentioned below:

Location

Location lets you can configure the following properties:

Table 11-3 • Preferences

Property	Description
Editor Working Location	This is the location on your computer which MSIX Editor uses to temporarily place the MSIX packages opened in the MSIX Editor for performing your desired tasks on the package. After all the tasks are completed, all the filed will be deleted. The default path is temp location of your machine. In this field, specify a location that MSIX Editor has access to.

Signing

MSIX packages must be digitally signed for successful installation. The following signing options are supported by MSIX Editor:

- Certificate File (.pfx)
- Certificate Store

Certificate File (.pfx)

Signing lets you configure the following properties:

Table 11-4 • Certificate File details

Property	Description
PFX File Location	Specify a path to a valid PFX certificate file.
Password	Specify the password for the PFX certificate file.
Certificate Subject	Certificate Subject populates with respect to the selected certificate.
Time Stamp Server URL	Specify the valid Time Stamp Server URL. You can now specify the Time Stamp Server URL while signing the MSIX packages using a certificate file (.pfx) in the MSIX Editor.

Table 11-4 • Certificate File details

Property	Description
Override Publisher with the Certificate Subject Name	Select the **Override Publisher with the Certificate Subject Name** check box, if you can now override the Publisher Name of an MSIX package by the Subject Name of the Digital Certificate while signing an MSIX package by checking this newly introduced check box so the MSIX package will be signed successfully.

Certificate Store

Note *• Make sure that the Certificate has been imported. For more details on importing certificate, see Import Certificate.*

Signing lets you configure the following properties:

Table 11-5 • Certificate Store details

Property	Description
Certificate Store Location	In the **Certificate Store Location** drop down, select either **User** or **Machine**.
Certificate Store Name	In the **Certificate Store Name** drop down, select any one of the store name from the list.
Certificate Subject	In the **Certificate Subject** drop down, select respective certificate subject from the list. After selecting, click **View Details** button to view the certificate details. ***Note*** *• The Publisher Name of the MSIX package should match with Certificate Subject of the Certificate Store file for successful signing.*
Override Publisher with the Certificate Subject Name	Select the **Override Publisher with the Certificate Subject Name** check box, if you can now override the Publisher Name of an MSIX package by the Subject Name of the Digital Certificate while signing an MSIX package by checking this newly introduced check box so the MSIX package will be signed successfully.

Output Window

When you perform any tasks like open, sign, make changes, and save on MSIX package, the **Output Window** appears at the bottom of the interface. It lists the Errors, Warnings, and other log messages that are generated while performing a task or functional testing of the MSIX package during tracing.

Note *• In the Output window of the MSIX Editor, you can see the hyper link saying* ***Click here*** *to open a log file. By clicking on the link it will launch the log file in a new window to evaluate.*

Editing MSIX Packages

The Appxmanifest.xml is the most important file of the MSIX package. This is an XML file which defines all the important aspects of the package and its features. Appxmanifest.xml dictates the installation and run time behavior of the package. The MSIX Editor is organized into various views in the left pane which is based on the Appxmanifest.xml structure. This arrangement of the views makes the tool intuitive to easily identify and understand all the properties of an MSIX package to make edits. You can edit the following views:

- Editing MSIX Package Information
- Editing Files and Folders
- Editing Registry
- Editing Dependencies
- Editing Package Declarations
- Editing Capabilities
- Editing Applications
- Editing Visual Assets
- Signing Package

Editing MSIX Package Information

When you open an existing MSIX package in the MSIX Editor, you may need to view or specify important package information. This includes information such as the Package Name, Publisher Name, Version and others. In the Package Information view, you can view, and if appropriate, edit information about your MSIX package.

To configure package information for your MSIX package:

- In the left pane view list under **Package Properties**, click the **Package Information**.
- In the properties grid that appears in the right pane, configure the package properties as needed. For details about each property, see Package Information View.

Editing Files and Folders

The Files and Folders view lets you make edits to the files and folders of your MSIX Package. You will be able to add new files and folders to your MSIX package and delete the files and folders from your MSIX package in this view.

- Adding a File or Folder to an MSIX Package
- Deleting a File or Folder From an MSIX Package

Adding a File or Folder to an MSIX Package

The Files and Folders view of the MSIX Editor lets you add a file or folder to your MSIX package.

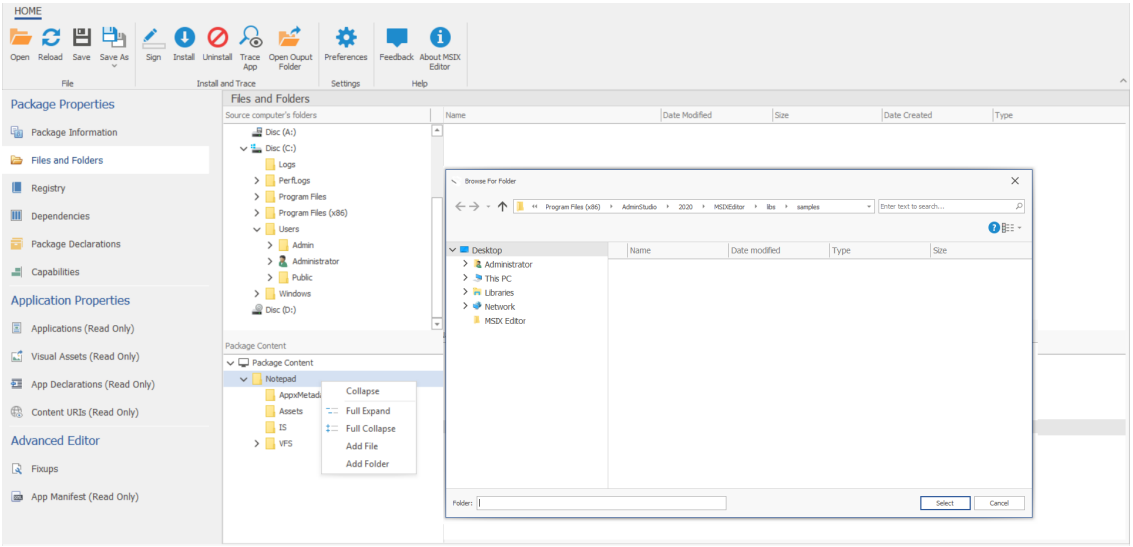
Task ***To add a file or folder to an MSIX package:***

1. Click on **Files and Folder** view in the left pane view list under Package Properties.

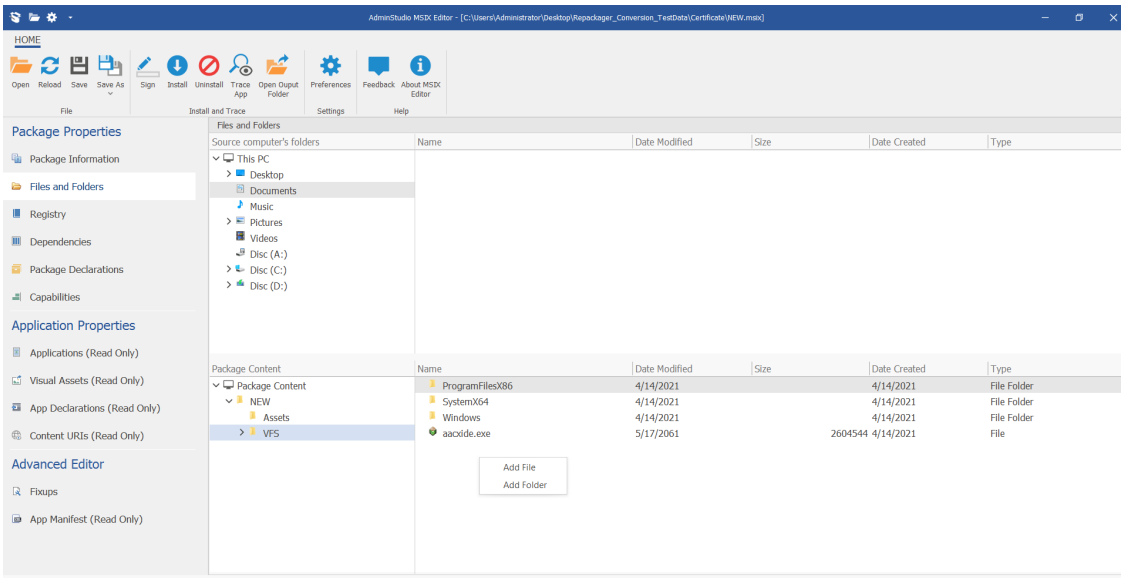
   The Files and Folder page appears.

2. In the **Package Content** pane on the bottom left, right-click on any folder where you want to add files or folder, and then click on **Add File** or **Add Folder** as appropriate.

3. A dialog box appears, browse the desired file or folder that you want to add, and then click the **Open** button.



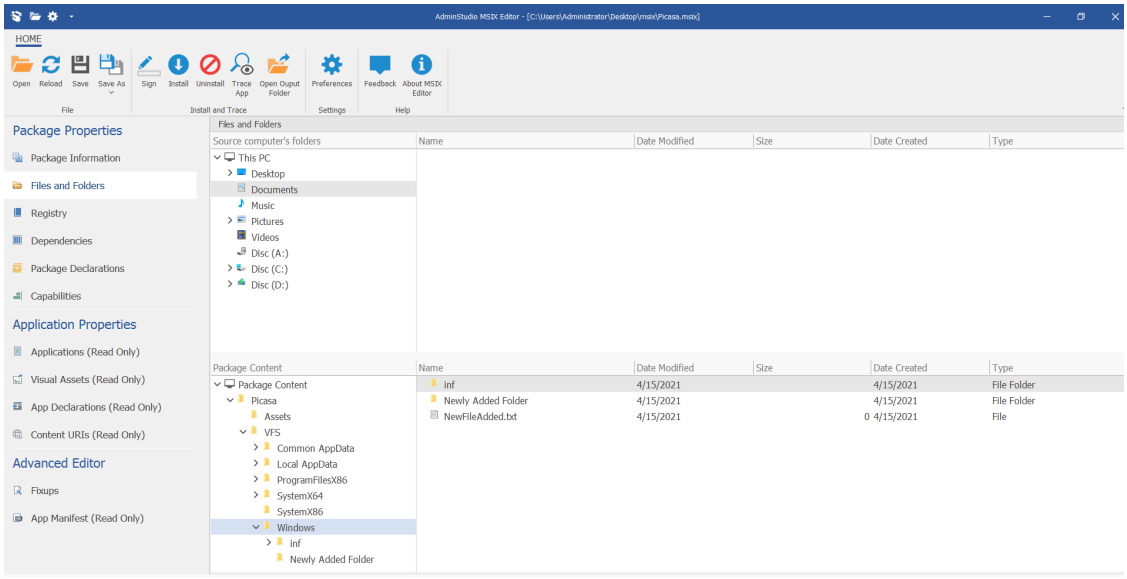
4. In the **Package Content** destination folder, right-click and select either **Add File** or **Add Folder**, and then browse to select the desired file or folder from the preferred location.



5. Click **Save**.

Note • *Added files or folders in the Destination details view will be added only when you save the package.*

6. Upon successful saving, files or folder will be added under the destination details view as shown in the following image.



Deleting a File or Folder From an MSIX Package

The **Files and Folders** view of the MSIX Editor lets you delete a file or folder to your MSIX package.

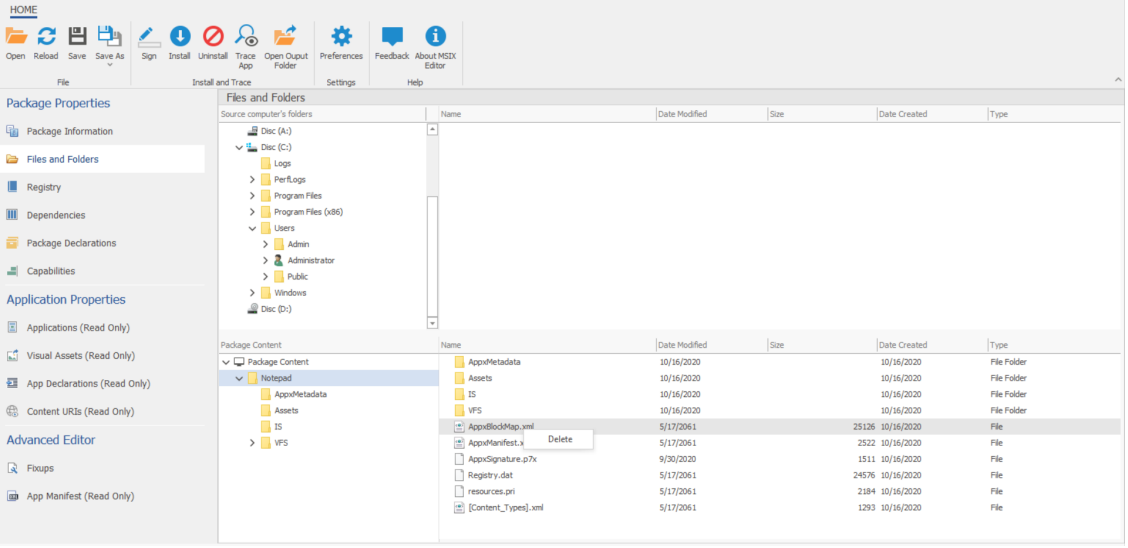
To remove a file or folder from an MSIX package, perform the following steps:

Task ***To delete a file or folder from an MSIX package:***

1. Click on **Files and Folder** view in the left pane view list under Package Properties.

   The Files and Folder page appears.

2. In the **Package Content** pane on the bottom right, right-click the file or folder that you want to delete, and then click **Delete**.



3. The deleted files or folders will be removed from your MSIX package upon saving the package.

Editing Registry

In Registry view you can make edits to the registries of the selected MSIX Package. You can perform the following actions under registry view:

- Importing a Registry File
- Adding a Registry Key
- Adding a Registry Value and Data to a Registry Key
- Modifying a Registry Value and Data of a Registry Key
- Deleting a Registry Key or Registry Value

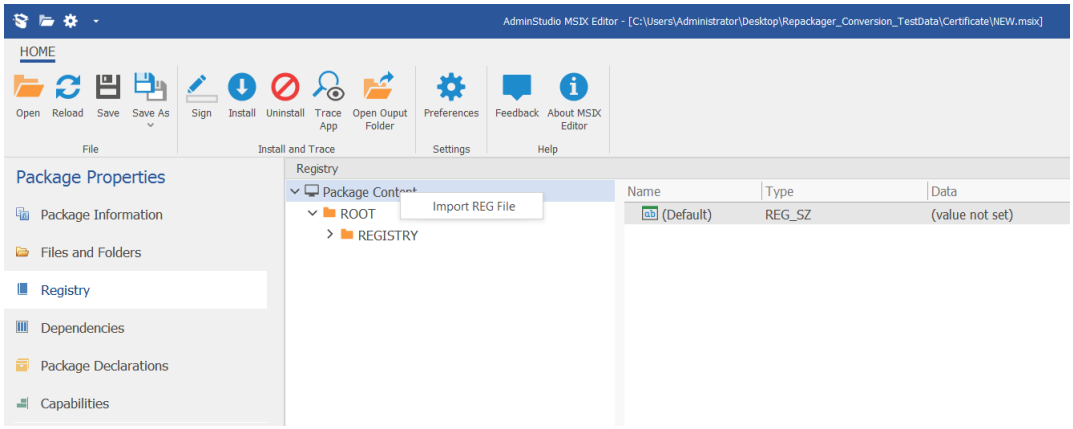
Importing a Registry File

To import registry file, perform the following steps:

Task ***To import a registry file:***

1. Click on **Registry** view in the left pane under Package Properties.
2. In the **Registry** pane, right-click on Package Content node, and then select **Import REG file**.

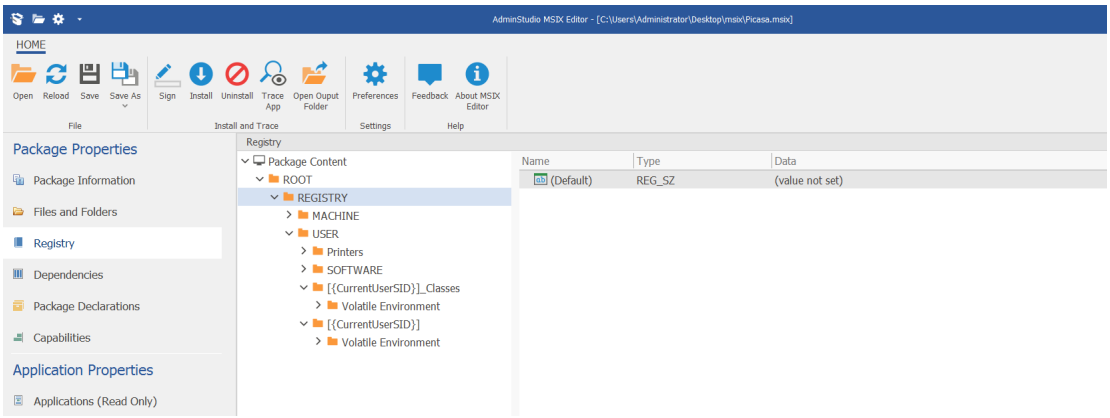
   Browse and select the desired registry file from the preferred location.



3. Click **Save** to import registry files to the selected package.

Note • *Imported registry files will be added only when you save the package.*

4. Upon successful saving, registry files will be imported under the respective nodes as shown in the following image.



Adding a Registry Key

To add a registry key, perform the following steps:

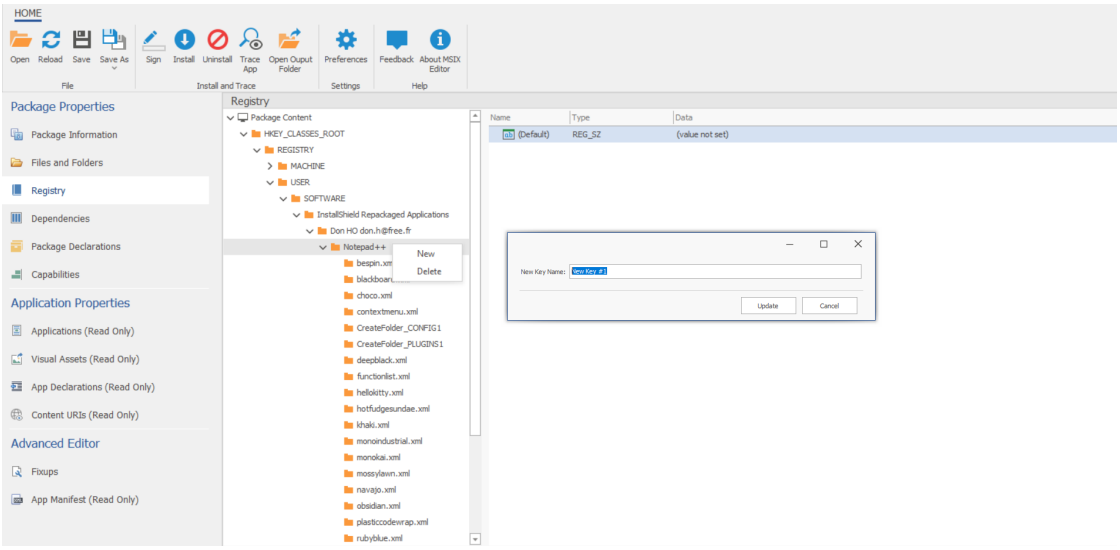
Task ***To add a registry key:***

1. Click on **Registry** view in the left pane view list under Package Properties.

   The Registry page appears.

2. In the **Registry** pane, right-click on any folder where you want to add registry key, and then click on **New**.

3. A dialog box appears, enter the desired key name, and then click the **Update** button.



4. Upon saving, the new registry key will be added to the selected MSIX package.

Adding a Registry Value and Data to a Registry Key

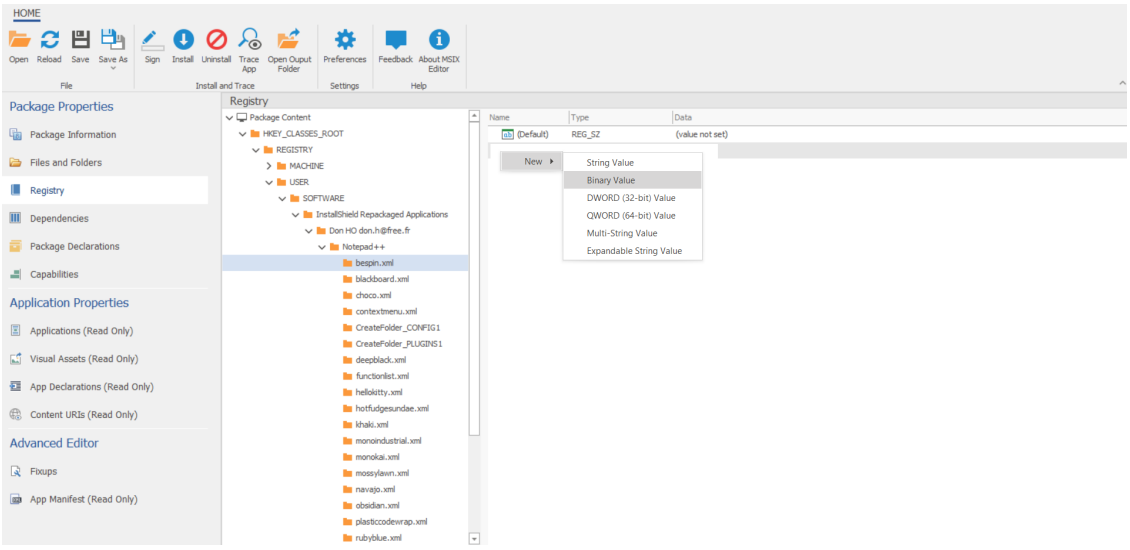
To add a value and data to a registry key, perform the following steps:

Task ***To add a value and data to a registry key:***

1. Click on **Registry** view in the left pane view list under Package Properties.

   The Registry page appears.

2. In the **Registry** pane, select desired registry key. To add registry value and data to the selected registry key, right click on the right pane, click on **New** drop down, and then select one of the value type from the list.



3. Upon saving, the new registry value and data will be added to the selected registry key.

Modifying a Registry Value and Data of a Registry Key

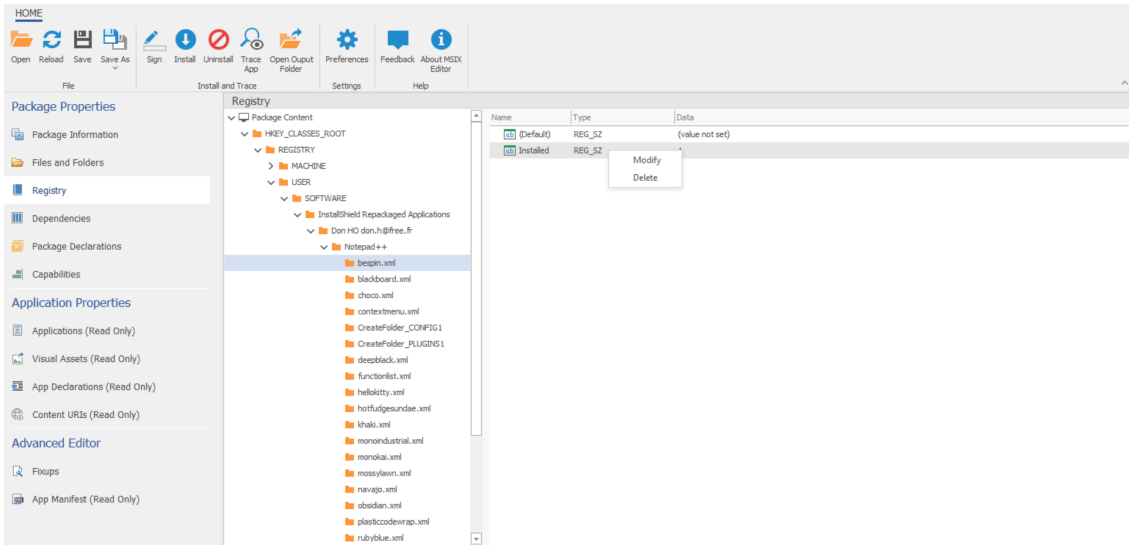
To modify a value and data of a registry key, perform the following steps:

Task ***To modify a value and data of a registry key:***

1. Click on **Registry** view in the left pane view list under Package Properties.

   The Registry page appears.

2. In the **Registry** pane, select desired registry key. To modify registry value and data to the selected registry key, right click on the desired value, and then select on **Modify**.



3. A dialog box appears, modify the desired values, and then click the **Update** button.

4. Upon saving, the new registry value and data will be updated.

Deleting a Registry Key or Registry Value

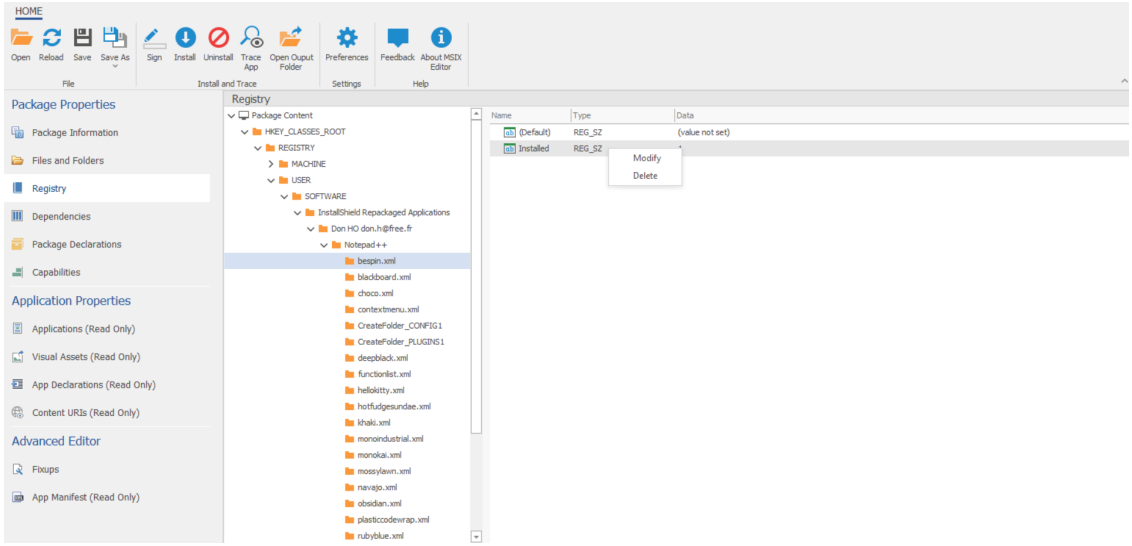
To delete a registry key or registry value, perform the following steps:

Task ***To delete a registry key or registry value:***

1. Click on **Registry** view in the left pane view list under Package Properties.

   The Registry page appears.

2. In the **Registry** pane, right-click the registry key or registry value that you want to delete, and then click **Delete**.



3. Click **Yes** and confirm to proceed the changes.
4. Upon confirming, the selected registry key or registry value will be removed from the registry.

Editing Dependencies

In the Dependencies view you make edits to the dependencies of the selected MSIX Package.

You can perform the following actions under dependencies view:

- Adding Dependency
- Modify Dependency
- Delete Dependency

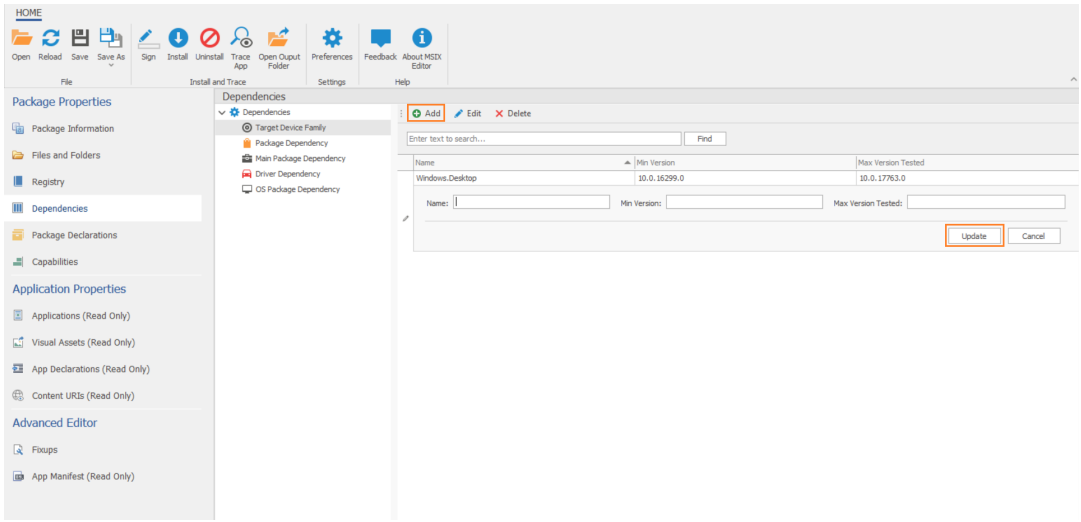
Adding Dependency

To add dependencies to the selected MSIX Package, perform the following steps

Task ***To add dependency:***

1. Click on **Dependencies** view in the left pane view list under Package Properties.

   The Dependencies page appears.
2. In the **Dependencies** pane, select the desired dependency type, and then click on **Add** button.
3. Selected dependency details field opens, enter the details and then click **Update** button.



4. Upon saving, a new dependency is added.

Modify Dependency

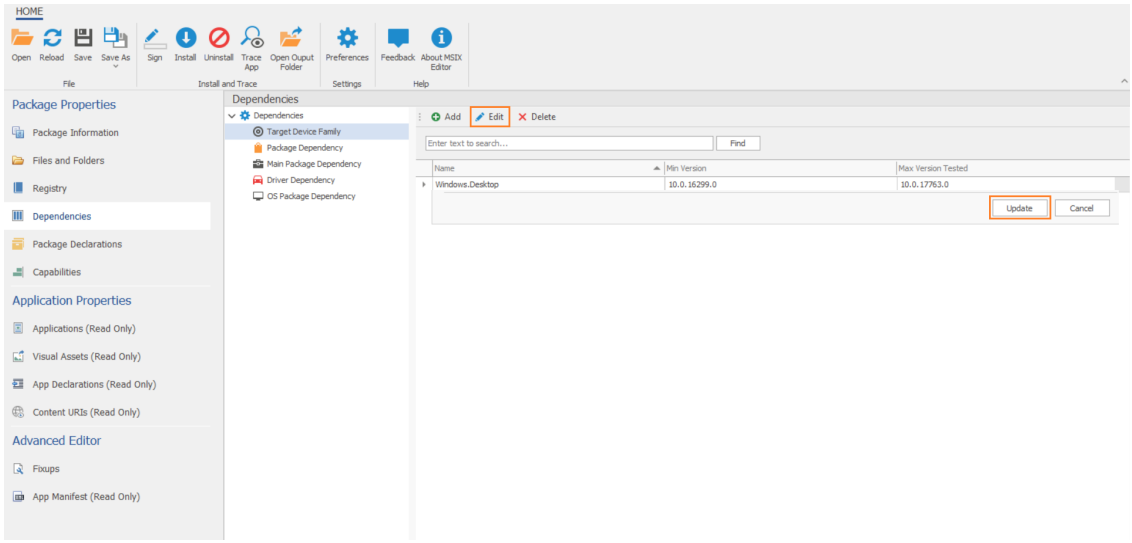
To modify the selected dependency, perform the following steps:

Task ***To modify:***

1. Click on **Dependencies** view in the left pane view list under Package Properties.

   The Dependencies page appears.

2. To edit, select the desired dependency, and then click the **Edit** button. The selected dependency is now enabled. Modify the desired fields and then click the **Update** button to save the changes.



Delete Dependency

To delete the selected dependency, perform the following steps:

Task ***To delete:***

1. Click on **Dependencies** view in the left pane view list under Package Properties.

   The Dependencies page appears.

2. To delete, select the desired dependency, and then click the **Delete** button.

Editing Package Declarations

In the Package declarations view you can make edits to the declarations of your MSIX Package. You can perform the following actions under the Package Declarations view:

- Add Package Declarations
- Modify Package Declarations
- Delete Package Declarations

Add Package Declarations

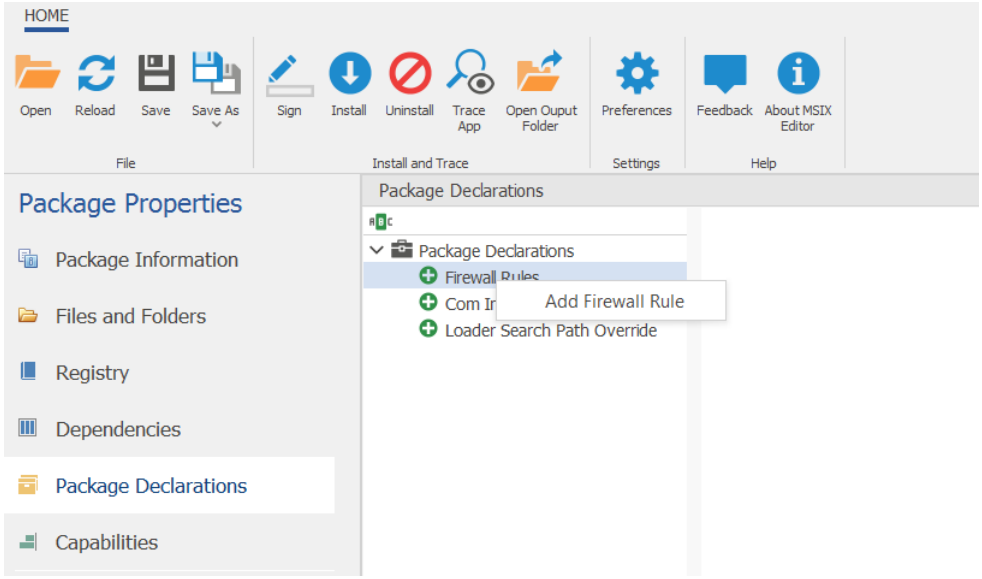
To add package declarations, perform the following steps:

Task ***To add package declarations:***

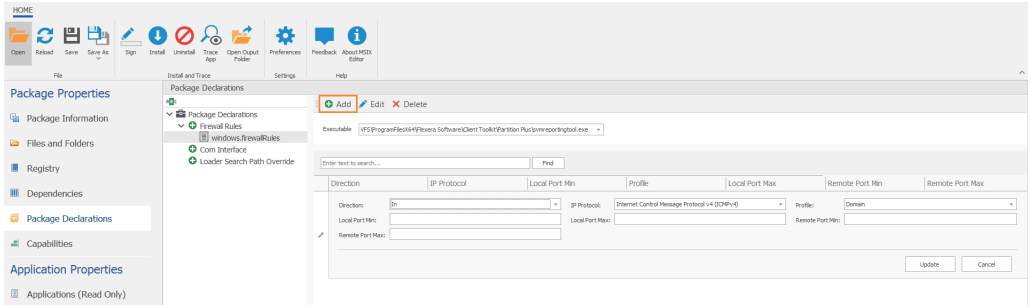
1. Click on **Package Declarations** view in the left pane view list under Package Properties.

   The package declarations page appears.

2. In the **Package Declarations** pane, right-click on any declaration where you want to add properties, and then click on **Add Rule** as appropriate.



3. Enter the details in the respective fields to configure the newly added Package Declaration properties and then click **Update** button.



4. Upon saving a new package declaration properties are added to the respective package declarations.

Modify Package Declarations

To modify package declarations, perform the following steps:

Task ***To modify:***

1. Click on **Package Declarations** view in the left pane view list under Package Properties.

   The package declarations page appears.

2. To edit, select the desired package declarations, and then click the **Edit** button. The selected package declaration is now enabled. Modify the desired fields and then click the **Update** button to save the changes.

Delete Package Declarations

To delete package declarations, perform the following steps:

Task ***To delete:***

1. Click on **Package Declarations** view in the left pane view list under Package Properties.

   The package declarations page appears.

2. To delete, select the desired package declarations, and then click the **Delete** button. The selected package declaration are removed.

Editing Capabilities

In the Capabilities view you can check or uncheck the desired capabilities by clicking on the respective check box button.

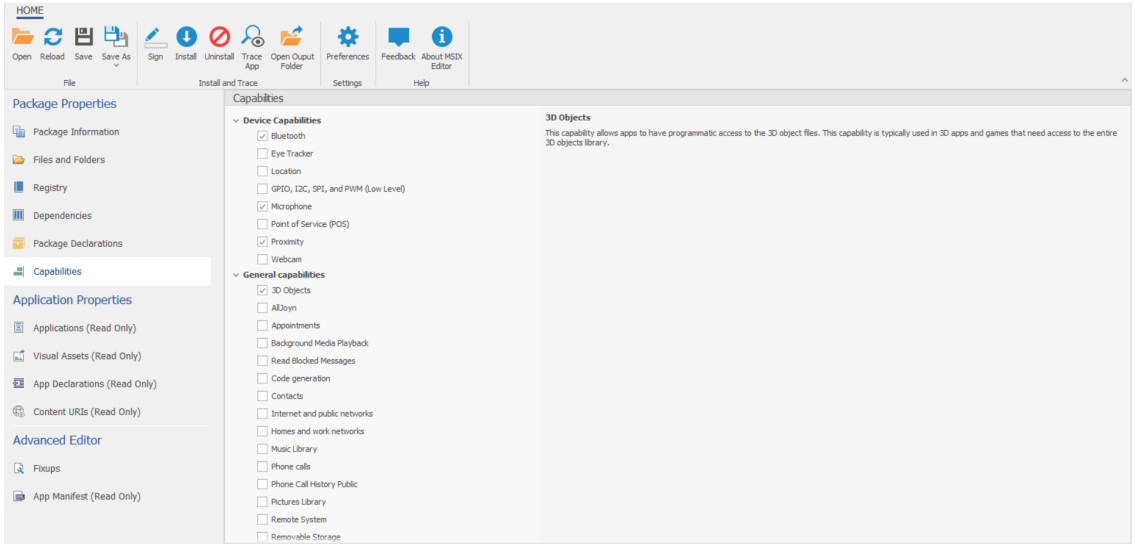
Task ***To modify the desired capabilities:***

1. Click on **Capabilities** view in the left pane view list under Package Properties.

   The capabilities page appears.

2. Click on the check box button and select the capability that you want to add to the package manifest file.

3. Click on the check box button and deselect the capability that you want to remove it from the package manifest file.



Editing Applications

In the Applications view you can view, create, edit, and delete applications of an MSIX Package. You can perform the following actions in the Applications view.

- Add a New Application
- Add a New Non-EXE Application
- Modify an Existing Application

- Delete an Existing Application

Add a New Application

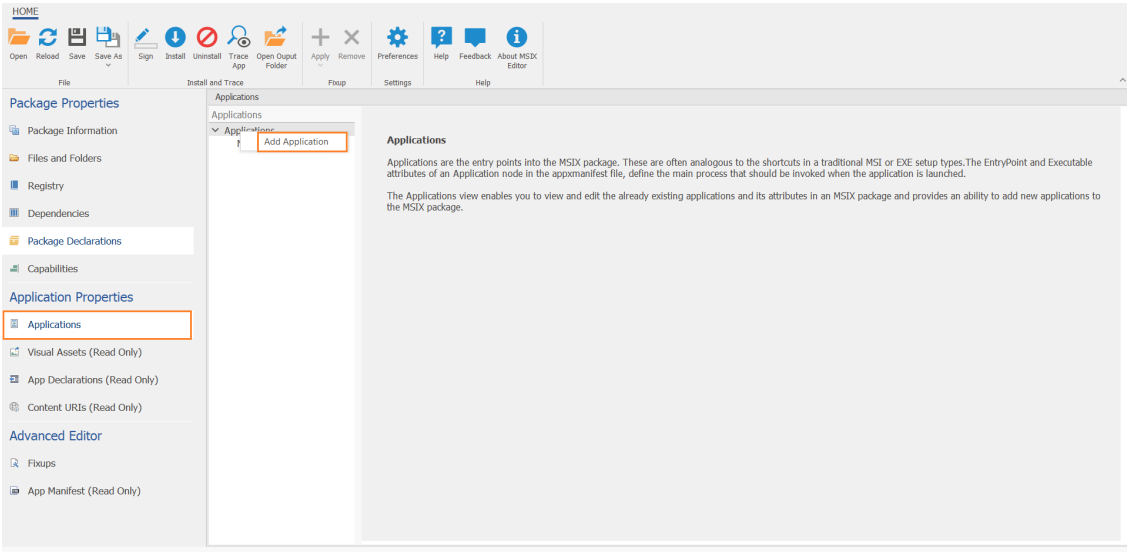
To add an application, perform the following steps:

Task ***To add an application:***

1. In the left pane, click on **Applications** under **Application Properties**.

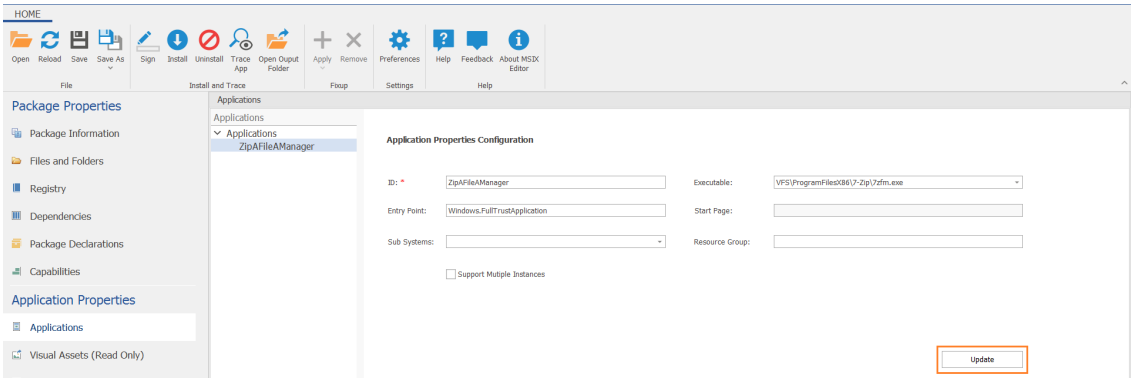
   The **Applications** page opens.

2. Right-click on the **Applications** node, and then select **Add Application** from the context menu.



   The **Application Properties Configuration** view opens.

3. Enter the details in the respective fields and then click **Update** button. For more information on Applications properties, see Applications View.



4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Add a New Non-EXE Application

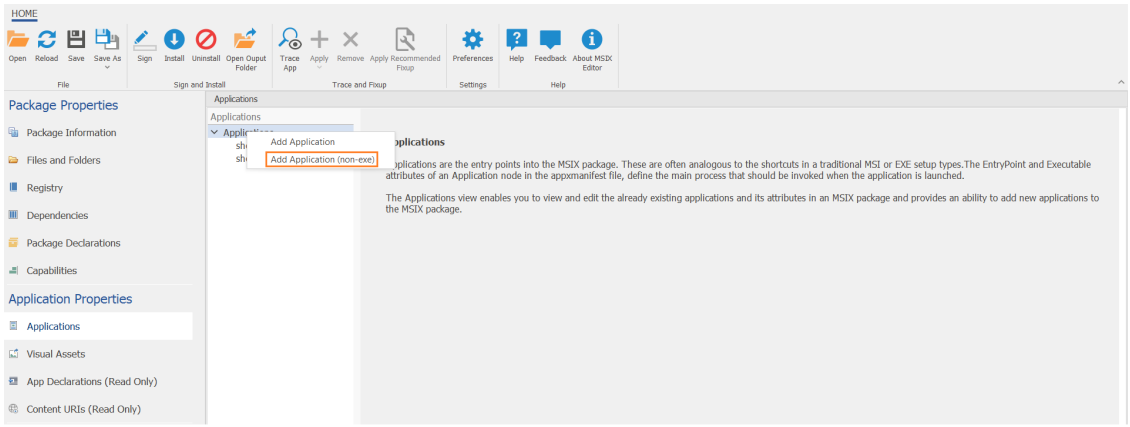
You can add a non-exe application to an MSIX package point to one the following file formats: .bat, .cmd, .txt, .html, and .docx. To create a non-exe application, perform the following steps:

Task ***To create a non-exe application:***

1. Navigate to the **Applications** view under the **Application Properties** menu.

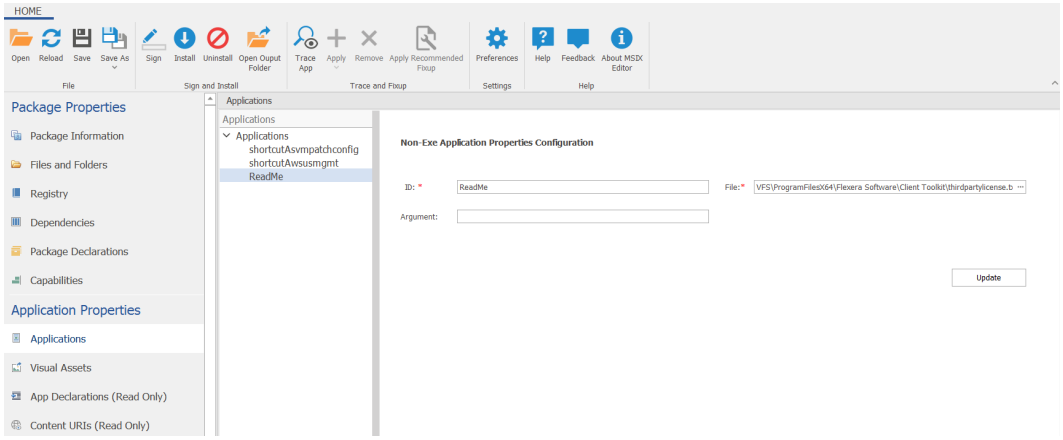
   The **Applications** page opens.

2. Right-click on the **Applications** node, and then select **Add Application (non-exe)** from the context menu.



   The **Non-Exe Application Properties Configuration** view opens.

3. Enter the details in the respective fields and then click **Update** button. For more information on Applications properties, see Applications View.



4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Modify an Existing Application

To modify the selected application, perform the following steps:

Task ***To modify:***

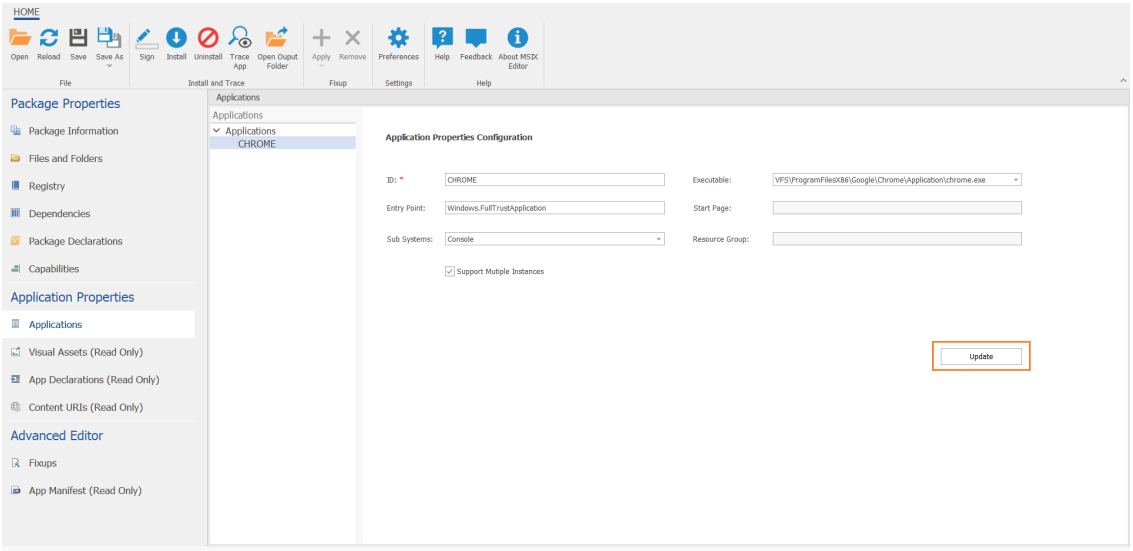
1. In the left pane, click on **Applications** under **Application Properties**.

   The **Applications** page opens.

2. Select the desired application under the **Applications** node.

   The **Application Properties Configuration** view opens.

3. Modify as necessary and click **Update**.



4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Delete an Existing Application

To delete an application, perform the following steps:

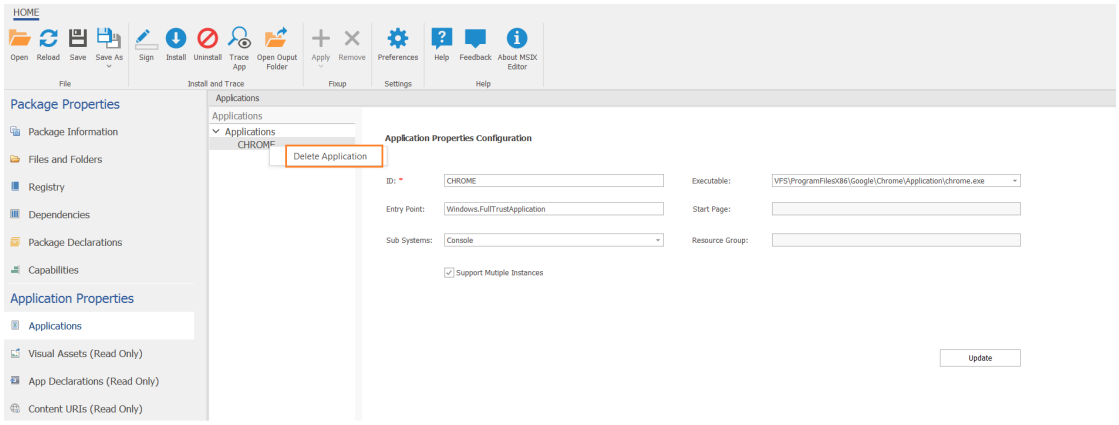
Task ***To delete:***

1. In the left pane, click on **Applications** under **Application Properties**.

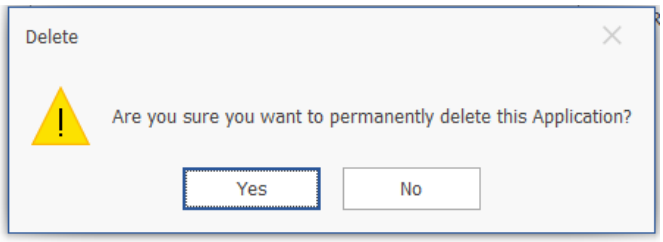
   The **Applications** page opens.

2. Select the desired application under the **Applications** node.

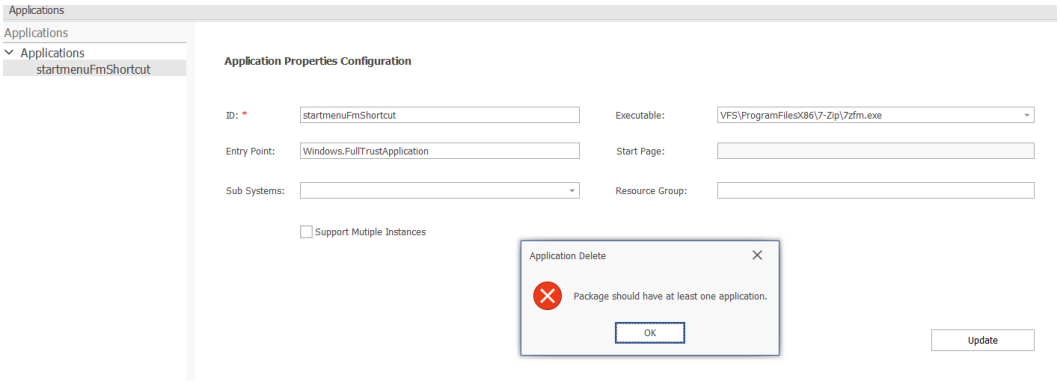
3. Right-click on application, and then select **Delete Application** from the context menu.



4. Click **Yes** and confirm to proceed the changes.



Note • *Make sure that at least one application is required in the MSIX package, if not below error message will appears.*



5. Click **Save** or **Save As** in the ribbon for the change to take effect.

Editing Visual Assets

The Visual Assets view enables you to view and edit visual assets of an MSIX package. You can also generate new visual assets for an MSIX package. This view allows you to edit each visual asset of an MSIX package.

You can perform the following actions under the Visual Assets view:

- Modify Visual Assets
- Generate Visual Assets

- Delete Visual Assets

Modify Visual Assets

To modify Visual Assets, perform the following steps.

Task ***To modify:***

1. In the left pane, click on **Visual Assets** under Application Properties.

   The Visual Assets page opens.

2. All fields in the **Display Settings**, **Small Tile**, **Medium Tile**, **Wide Tile**, **Large Tile**, **App Icon**, **Splash Screen**, **Badge Logo**, and **Package Logo** tabs are editable.

3. Modify as necessary and click **Save** or **Save As** in the ribbon for the changes to take effect.

Generate Visual Assets

To generate visual Assets, perform the following steps.

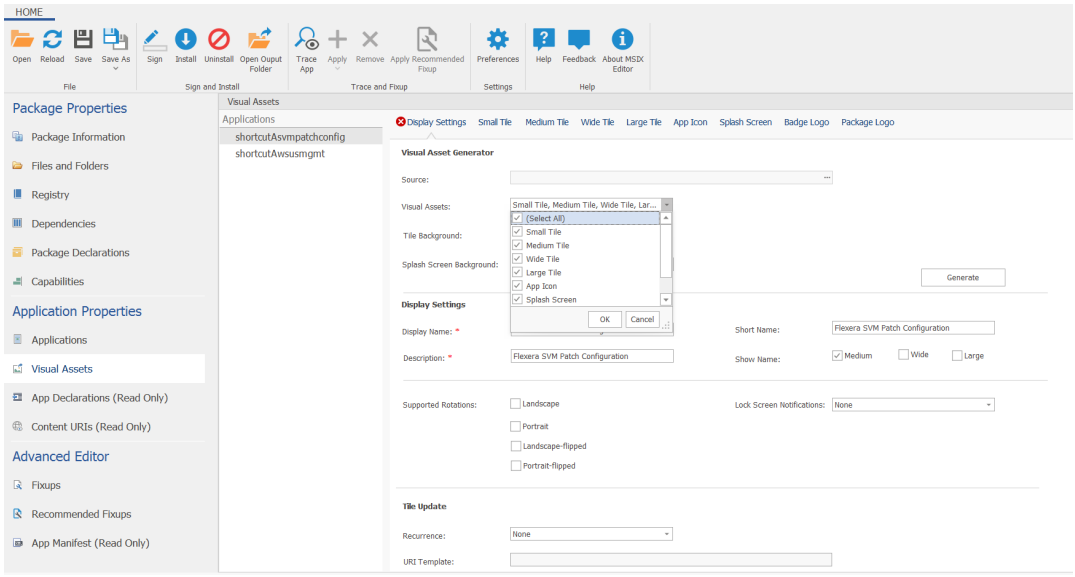
Task ***To generate Visual Assets:***

1. Click on **Visual Assets** view in the left pane under Application Properties.

   The Visual Assets page appears.

2. In the **Source** field, specify the source file location, or click **Browse** and select the desired file (in .png, .jpg, .jpeg, .gif, .pdf, .tiff, .bmp, .svg, .ico, format).

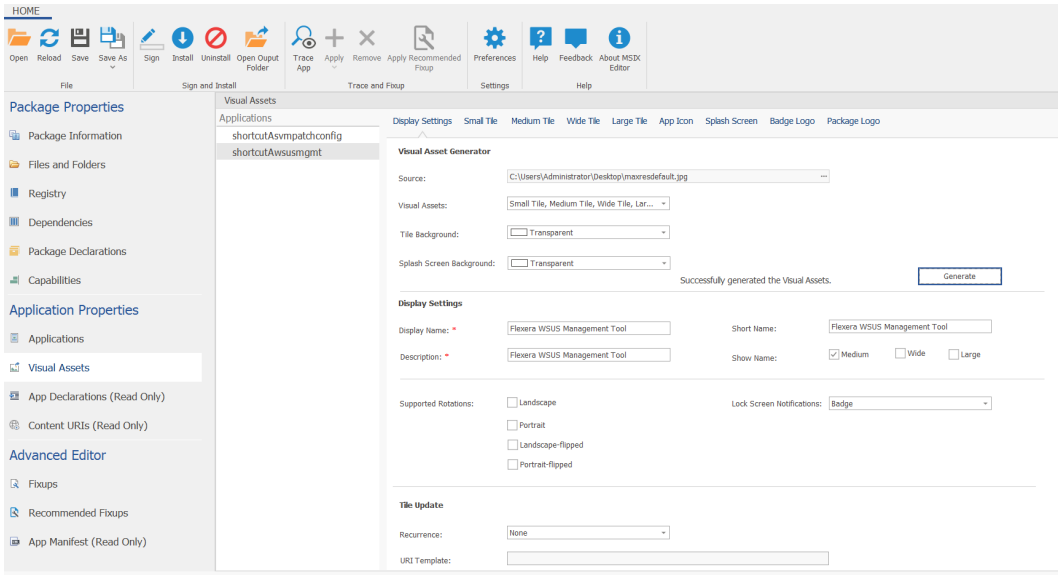
3. In the **Visual Assets** drop down, select desired visual assets, and then click **Ok** button.



Note • *Multiple selections are allowed.*

4. In the **Tile Background** drop down, select desired tile background color from the list.
5. In the **Splash Screen Background** drop down, select desired splash screen color from the list.
6. After entering the above fields, click **Generate** button.
7. Upon successful action, selected visual assets are generated. The following message will be displayed on the page.

   Successfully generated the Visual Assets.



8. Click **Save** or **Save As** in the ribbon for the changes to take effect.

Delete Visual Assets

To delete Visual Assets, perform the following steps.

Task ***To delete:***

1. In the left pane, click on **Visual Assets** under Application Properties.

   The Visual Assets page opens.

2. In the **Small Tile**, **Wide Tile**, **Large Tile**, **Splash Screen**, and **Badge Logo** tabs all images can be deleted by clicking on **Clear** button.

3. In the **Package Logo** tab, all images are mandatory. You can not delete images.

4. In the **Medium Tile** tab 150x150 resolution image can not be deleted because this image is mandatory. Remaining images can be deleted by clicking on **Clear** button.

5. In the **App Icon** tab 44x44 resolution image can not be deleted because this image is mandatory. Remaining images can be deleted by clicking on **Clear** button.

Note • *You can also delete images at each Image-level.*

Signing Package

MSIX packages must be digitally signed for successful installation. The MSIX Editor supports **Certificate File (.pfx)** and **Certificate Store** signing option.

Fix Runtime Issues of an MSIX Package

Using MSIX Editor you can identify and fix run time issues of your MSIX package.

- Tracing App
- Applying Fixups
- Applying Recommended Fixup

Tracing App

Tracing of your MSIX package enables you to identify any run time issues of your package.

Task ***To trace an MSIX package:***

1. Open your desired MSIX package in the MSIX Editor using the **Open** button in the MSIX Editor ribbon.

2. If the MSIX Package is not already signed, specify a certificate file (.pfx) or Certificate Store in the **Signing** section of **Preferences**.

3. Click on the **Trace App** button in MSIX Editor ribbon.
4. Upon launching, run the application and notice the logs in the **Output** window to identify the run time issues while tracing is running in the background.

Note • *An MSIX package must be signed and installed for successful tracing. The package will first be signed, if not already signed and then installed to begin tracing.*

Applying Fixups

Repackaging a legacy Win32 application to an MSIX format to run inside a container may lead to runtime failures at times. The Package Support Framework (PSF) is an open-source kit from Microsoft that helps you resolve some of the runtime failures when you don't have access to the source code. PSF contains a collection of software components that work as runtime fixes, also referred to as fixups, to overcome some commonly seen failures.

The Fixups view in the MSIX Editor enables you to easily apply the fixups to the applications in an MSIX package.

The MSIX Editor supports the following fixups that can be applied to the applications which encounter runtime failures:

- File Redirection Fixup
- PowerShell Script Fixup
- File Not Found Fixup
- Executable Arguments Fixup
- Registry Fixup
- Environment Variable Fixup
- Custom Fixup

File Redirection Fixup

Unlike the traditional installers like MSI and EXE, the MSIX package format does not allow writing into the installation directory. If an application, packaged to an MSIX format, attempts to write a file into the installation directory then the application may encounter an error or may even crash. Similarly, if an application is trying to look for a file in its current working directory then the application may encounter an error that the file is not found though the file is present, the reason being in the MSIX format the current working directory of an application is returned as C:\Windows\System32 instead of the actual current working directory of the application.

The File Redirection Fixup (FRF) helps to address these runtime errors by redirecting the files to be written in the installation directory to the LocalAppData folder and by returning the correct current working directory to the application looking for the file.

You may choose one of the following three FRF fixups depending on where do you expect the file would be written to by the application:

Table 11-6 • Applying a File Redirection Fixup

Type	Description
Package Relative	Choose this fixup if the application is expected to write a file into its installation directory.
Package Drive Relative	Choose this fixup if the application is expected to write a file in any directory under the drive were the package is installed.
Known Folders	Choose this fixup if the application is expected to write a file to one of the standard Windows pre-defined folders, for example SystemX86.

Package Relative

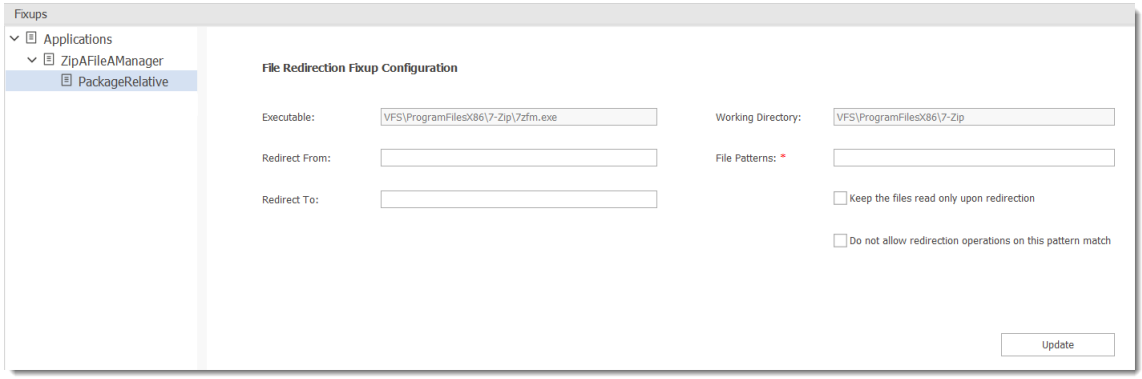
To add a **Package Relative** fixup, perform the following steps:

Task ***To add Package Relative fixups:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **File Redirection Fixup > Package Relative** from the menu.

   Note • *You could also select an application, right click, and select **Apply File Redirection Fixup > Package Relative** from the context menu.*

   The **File Redirection Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.

Property	Description
Redirect From	Microsoft refers to this property as **base**. Specify the path where the file is expected to be created by the application in the drive where the package is installed. This property should be a folder path that should be relative to the drive where the package is installed to (generally C:). ***Note*** • *If not specified, then the working directory of the application will be used.*
Redirect To	Microsoft refers to this property as **redirectTargetBase** Specify the folder where you wish to redirect the file to. ***Note*** • *If not specified, the file will be redirected to the LocalAppData folder on the device.*
Working Directory	This property displays the working directory of the application. This is a non-editable property.
File Patterns	Microsoft refers to this property as **patterns**. Specify the file extensions to be considered for the file redirections. If you wish to redirect any file extension or all the file extensions then specify .* This is a mandatory property.
Keep the files read only upon redirection	If you select this check box, then a copy on access and redirection operation is performed. The redirected files will be read-only and non-edited. By default this property is unchecked.
Do not allow redirection operations on this pattern match	If you select this check box, then redirection operations are not allowed on this pattern match. By default this property is unchecked.

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Package Drive Relative

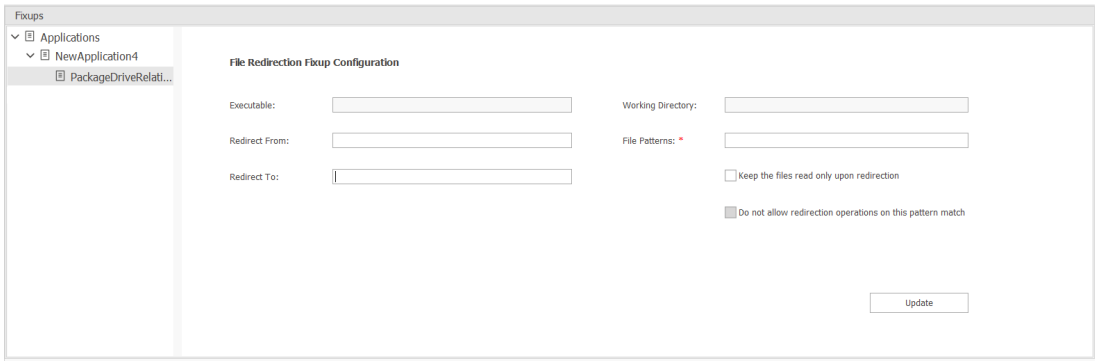
To add a **Package Drive Relative** fixup, perform the following steps:

Task ***To add Package Drive Relative fixups:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **File Redirection Fixup > Package Drive Relative** from the menu.

   Note • *You could also select an application, right click, and select* ***Apply File Redirection Fixup > Package Drive Relative*** *from the context menu.*

   The **File Redirection Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Redirect From	Microsoft refers to this property as **base**. Specify the path where the file is expected to be created by the application in the drive where the package is installed. This property should be a folder path that should be relative to the drive where the package is installed to (generally C:). ***Note*** • *If not specified, then the working directory of the application will be used.*
Redirect To	Microsoft refers to this property as **redirectTargetBase** Specify the folder where you wish to redirect the file to. ***Note*** • *If not specified, the file will be redirected to the LocalAppData folder on the device.*
Working Directory	This property displays the working directory of the application. This is a non-editable property.

Property	Description
File Patterns	Microsoft refers to this property as **patterns**. Specify the file extensions to be considered for the file redirections. If you wish to redirect any file extension or all the file extensions then specify .* This is a mandatory property.
Keep the files read only upon redirection	If you select this check box, then a copy on access and redirection operation is performed. The redirected files will be read-only and non-edited. By default this property is unchecked.
Do not allow redirection operations on this pattern match	If you select this check box, then redirection operations are not allowed on this pattern match. By default this property is unchecked.

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Known Folders

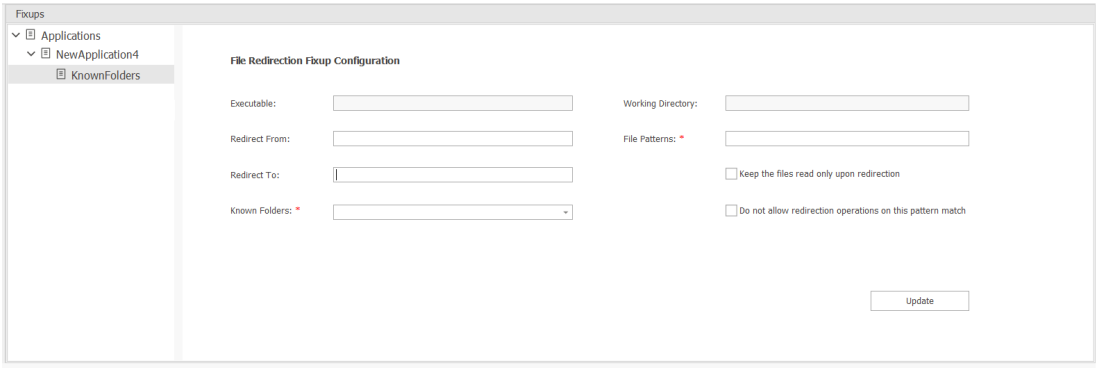
To add a **Known Folders** fixup, perform the following steps:

Task ***To add Known Folders fixups:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **File Redirection Fixup > Known Folders** from the menu.

Note • *You could also select an application, right click, and select* ***Apply File Redirection Fixup > Known Folders*** *from the context menu.*

The **File Redirection Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Redirect From	Microsoft refers to this property as **base**. Specify the path where the file is expected to be created by the application in the drive where the package is installed. This property should be a path relative to the Windows predefined folder selected for the Known Folder property. ***Note*** • *If not specified, then the working directory of the application will be used.*
Redirect To	Microsoft refers to this property as **redirectTargetBase** Specify the folder where you wish to redirect the file to. ***Note*** • *If not specified, the file will be redirected to the LocalAppData folder on the device.*
Known Folders	Select one of the following: • LocalAppData • ProgramData • ProgramFilesX86 • ProgramFilesX64 • ProgramFilesCommonX86 • ProgramFilesCommonX64 • RoamingAppData • System • SystemX86 • Windows This is a mandatory property.
Working Directory	This property displays the working directory of the application. This is a non-editable property.
File Patterns	Microsoft refers to this property as **patterns**. Specify the file extensions to be considered for the file redirections. If you wish to redirect any file extension or all the file extensions then specify .* This is a mandatory property.

Property	Description
Keep the files read only upon redirection	If you select this check box, then a copy on access and redirection operation is performed. The redirected files will be read-only and non-edited. By default this property is unchecked.
Do not allow redirection operations on this pattern match	If you select this check box, then redirection operations are not allowed on this pattern match. By default this property is unchecked.

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Note • *When multiple FRF fixups are applied to an application in an MSIX package, a single config.json file will be generated merging all the fixups configuration and injected into the MSIX package along with the fixup binary files.*

PowerShell Script Fixup

The PowerShell Script Fixup enables you to specify PowerShell scripts to be executed before and after the launch of an application. This will be an ideal fixup to apply to an application if you have any first time run tasks to be executed upon launching of an application to the first time and perform any cleanup tasks on closing the application.

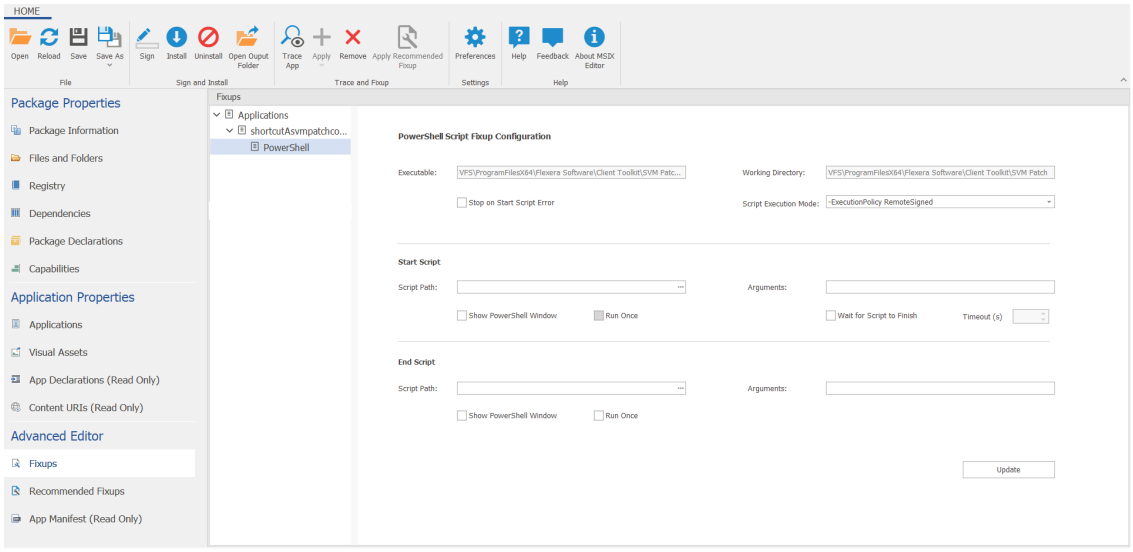
To add a **PowerShell Script Fixup**, perform the following steps:

Task ***To add PowerShell Script Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **PowerShell Script Fixup** from the menu.

   Note • *You could also select an application, right click, and select* ***Apply PowerShell Script Fixup*** *from the context menu.*

   The **PowerShell Script Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Working Directory	This property displays the working directory of the application. This is a non-editable property.
Script Execution Mode	Provide the execution mode of the PowerShell script as a part of Start Script or End Script. Select one of the following: ● ExecutionPolicy RemoteSigned ● ExecutionPolicy AllSigned ● ExecutionPolicy Bypass ● ExecutionPolicy Default ● ExecutionPolicy Restricted ● ExecutionPolicy RemoteSigned ● ExecutionPolicy Undefined ● ExecutionPolicy Unrestricted
Stop on Start Script Error	If you select this check box, the Application will not be launched if the Start Script encounters an error.

Property	Description
Start Script	Fill in the following fields to define the start script, a PowerShell script that will be run prior to launching of an application: • **Script Path**: Click the ellipses (...) to browse for a PowerShell script file (ps1) that should be executed upon launch of the application. • **Arguments**: Specify the arguments to be passed to the PowerShell start script specified in the **Script Path** property. Use of pseudo-variables or environment variables are supported. • **Show PowerShell Window**: If you select the check box, the PowerShell Window will be displayed during the execution of the script. By default it is unchecked. • **Run Once**: If you select the check box, the script will run only when you run launch application for the first time. By default it is unchecked. • **Wait for Script to Finish**: If you select the check box, launching of the application will be delayed until the start script execution is complete or a timeout occurs. • **Timeout (s)**: Set the maximum number of seconds to wait for the script to finish. If a timeout occurs it is treated as an error for the purpose of Stop on Script Error. The value 0 means an immediate timeout, if you do not want a timeout do not specify a value. ***Note*** *• Applicable only if you set* ***Wait for Script to Finish*** *option to True.*
End Script	Fill in the following fields to define the end script, a PowerShell script that will be run after closing an application: • **Script Path**: Click the ellipses (...) to browse for a PowerShell script file (ps1) that should be executed upon launch of the application. • **Arguments**: Specify the arguments to be passed to the PowerShell start script specified in the **Script Path** property. Use of pseudo-variables or environment variables are supported. • **Show PowerShell Window**: If you select the check box, the PowerShell Window will be displayed during the execution of the script. By default it is unchecked. • **Run Once**: If you select the check box, the script will run only when you run launch application for the first time. By default it is unchecked.

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Note *• One of the following three properties must to specified to apply a PowerShell Script Fixup: Arguments, Start Script Path, and End Script Path.*

Note • *Only one PowerShell Script Fixup can be applied to an application.*

File Not Found Fixup

File Not Found fixup is applied to an application when a file is not found by the application at runtime. This fixup requires the working directory of the file, that will be automatically populated in the fixup's configuration.

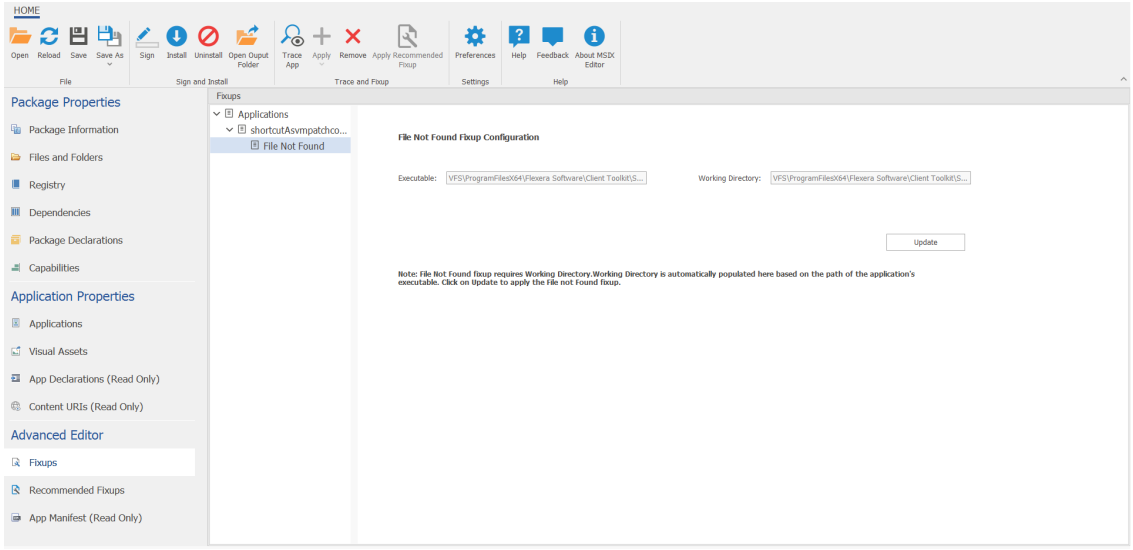
To create a **File Not Found Fixup**, perform the following steps:

Task ***To create File Not Found Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **File Not Found Fixup** from the menu.

   Note • *You could also select an application, right click, and select* ***File Not Found Fixup*** *from the context menu.*

   The **File Not Found Fixup Configuration** view opens.



2. Below properties are auto populated:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Working Directory	This property displays the working directory of the application. This is a non-editable property.

3. After verifying the above fields, click **Update** button.

Note • *If File Not Found fixup is combined with the other fixups then it will not be shown in the User Interface when the package is reloaded.*

4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Note • *Only one File Not Found Fixup can be applied to an application.*

Executable Arguments Fixup

MSIX packages inherently do not support passing of arguments to an executable (.exe) while launching an application. To pass arguments to an executable you need to apply an Executable Arguments fixup to the application.

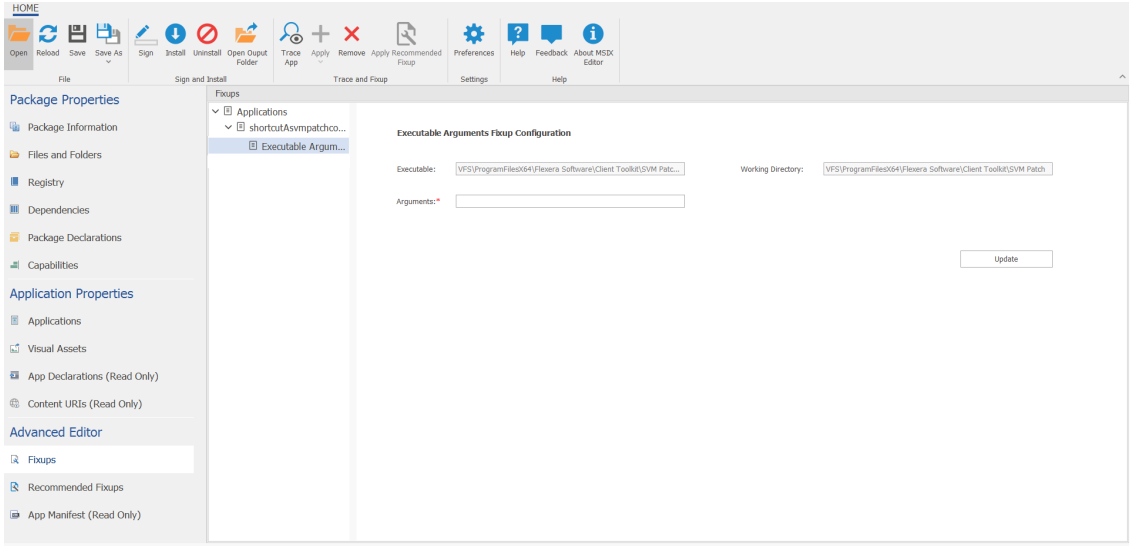
To create an Executable Arguments Fixup, perform the following steps:

Task ***To create Executable Arguments Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **Executable Arguments Fixup** from the menu.

Note • *You could also select an application, right click, and select* ***Executable Arguments Fixup*** *from the context menu.*

The **Executable Arguments Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.

Property	Description
Working Directory	This property displays the working directory of the application. This is a non-editable property.
Arguments	Specify the command-line arguments that you wish to pass to executable of an application upon its launch. This is a mandatory property.

3. After providing argument property value, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Note • *Only one Executable Arguments Fixup can be applied to an application.*

Registry Fixup

There are two types of Registry Fixups: Modify Access and Fake Delete. Some applications may request for more permissions than they need while reading or creating registries. Such requests may sometimes be denied for an application running in an MSIX container thus failing the subsequent registry actions. The Modify Access Registry Fixup allows you to modify the requested access to return success for the registry calls that otherwise would have failed. The Fake Delete Registry Fixup returns a fake success to a registry delete request made by an application to make it believe that the registry was deleted.

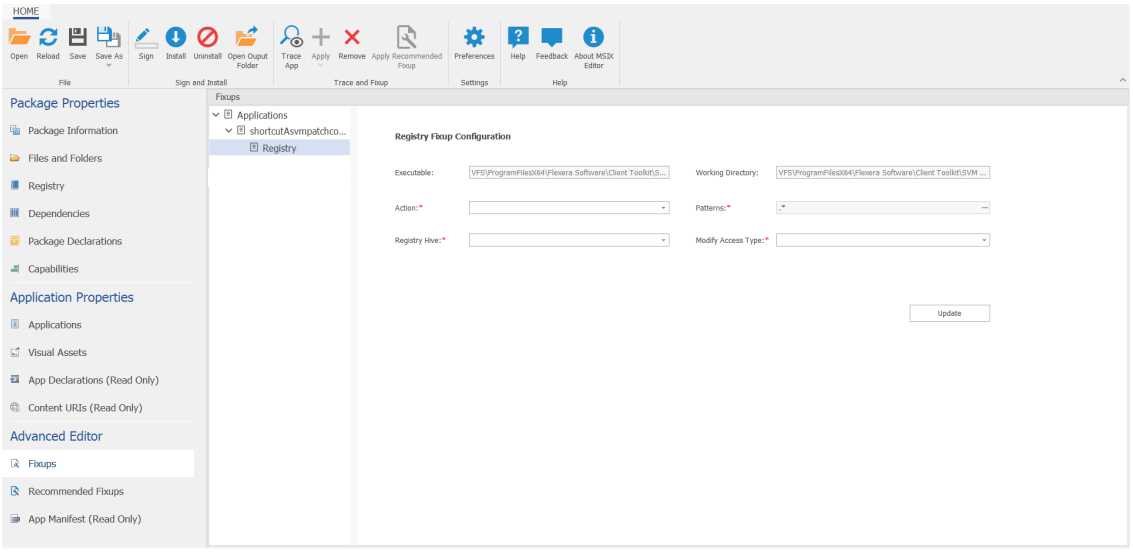
To create a **Registry Fixup**, perform the following steps:

Task ***To create a Registry Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **Registry Fixup** from the menu.

   Note • *You could also select an application, right click, and select **Registry Fixup** from the context menu.*

   The **Registry Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Working Directory	This property displays the working directory of the application. This is a non-editable property.
Action	Select one of the following: ● Modify Registry Access ● Delete Registry This is a mandatory property.
Patterns	This property is defaulted to .* This is a mandatory property.
Registry Hive	Select one of the following: ● Local Machine (HKLM) ● Current Machine (HKCU) This is a mandatory property.

Property	Description
Modify Access Type	Allows to select required access type: • Full2R - Full to Read • Full2RW - Full to Read Write • Full2MaxAllowed - Full to Max Allowed • RW2R - Read/Write to Read • RW2RMaxAllowed - Read/Write to Max Allowed This is a mandatory property. ***Note*** *• If* ***Delete Registry*** *under* ***Action*** *property is selected, this field is disabled.*

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Environment Variable Fixup

MSIX does not support new and changes environment variables inside a package. The Environment Variables Fixup allows to specify an application-specific scope environment variable to be read or modified by application in an MSIX package.

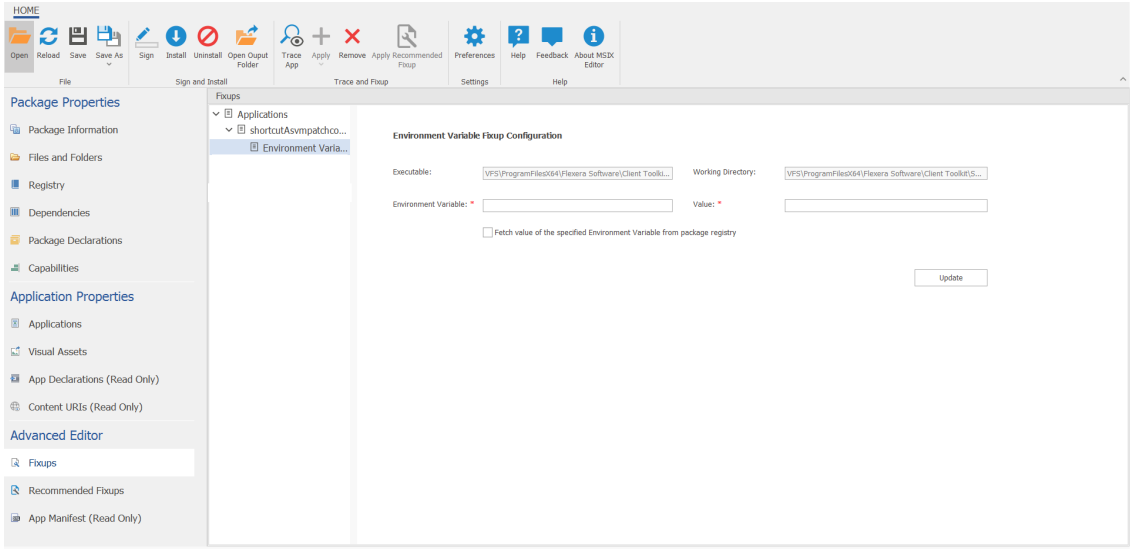
To add a **Environment Variable Fixup**, perform the following steps:

Task ***To add Environment Variable Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **Environment Variable Fixup** from the menu.

   Note *• You could also select an application, right click, and select* ***Environment Variable Fixup*** *from the context menu.*

   The **Environment Variable Fixup Configuration** view opens.



2. Enter the following information:

Property	Description
Executable	This property displays the executable that will be invoked when the application is launched. This is a non-editable property.
Working Directory	This property displays the working directory of the application. This is a non-editable property.
Environment Variable	Specify environment variable name. This is a mandatory property.
Value	Specify the value for the configured environment variable. This is a mandatory property.
Fetch value of the specified Environment Variable from package registry	If you select this check box, value of the provided environment variable will be fetched from the system registry.

3. After adding the above fields, click **Update** button.
4. Click **Save** or **Save As** in the ribbon for the change to take effect.

Custom Fixup

The Custom Fixup can be applied for the fixups that are not supported by MSIX Editor currently. The Custom Fixup requires the config.json file to be generated manually. The config.json along with all the runtime binary files required by the Custom Fixup must be placed in the same folder. The path to the config.json file must be specified in the **Json Path** property

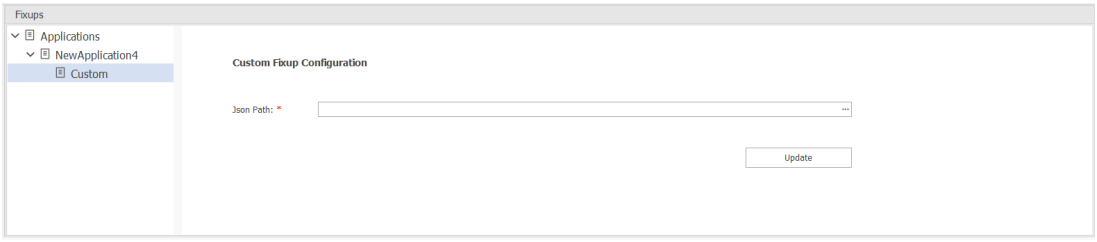
To add a **Custom Fixup**, perform the following steps:

Task ***To add Custom Fixup:***

1. Select an application in the **Applications** node, click the **Apply** button in the MSIX Editor ribbon, and then select **Custom Fixup** from the menu.

   Note • *You could also select an application, right click, and select* ***Apply Custom Fixup*** *from the context menu.*

   The **Custom Fixup Configuration** view opens.

2. Click ellipses (...) to browse and select the Json file that you want to add and click **Update** button. This is a mandatory property.

Note • *In Custom fixup, you can provide other fixups which are not supported by MSIX Editor.*

Note • *Custom fixup will not be shown in the User Interface when the package is reloaded.*

3. Click **Save** or **Save As** in the ribbon for the change to take effect.

Applying Recommended Fixup

MSIX Editor can automatically recommend fixups for the run time issues encountered while testing an application during trace. The automatic recommendation of fixups currently supports only the File Redirection Fixup and Registry Fix.

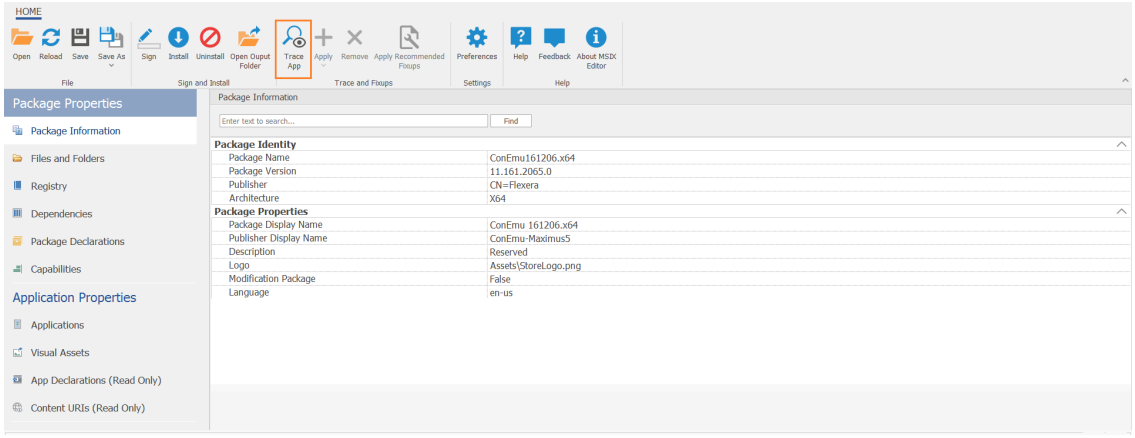
Perform the following steps to apply recommended fixup.

Task ***To apply recommended fixup:***

1. Open an MSIX package in the MSIX Editor using the **Open** button in the ribbon.

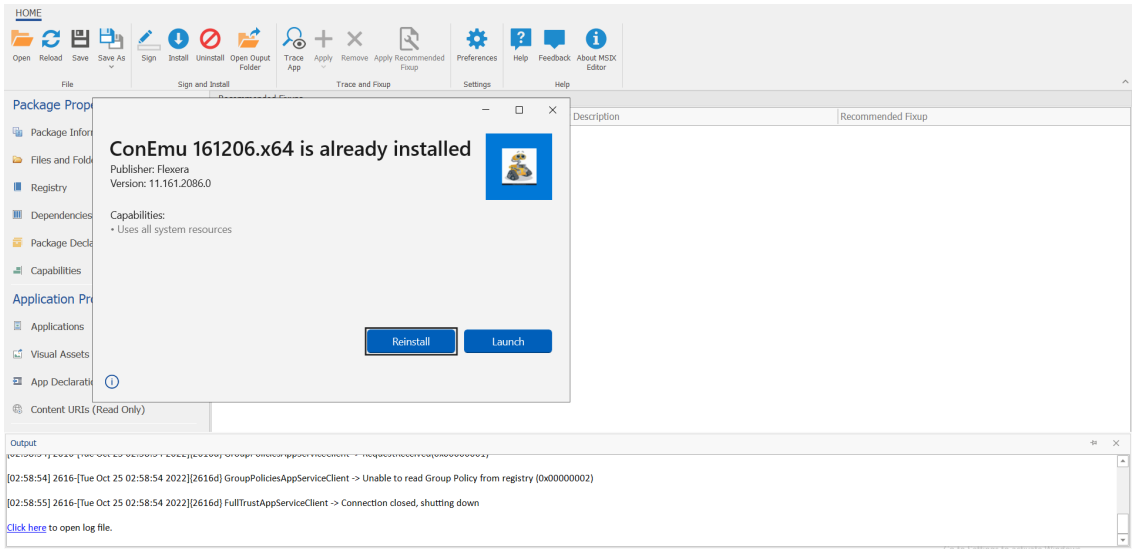
   Note • *Make sure that the MSIX Package is digitally signed. For more information, see Signing section of Preferences.*

2. Click on the **Trace App** button in MSIX Editor ribbon.



It navigates to the **Recommended Fixups** view.

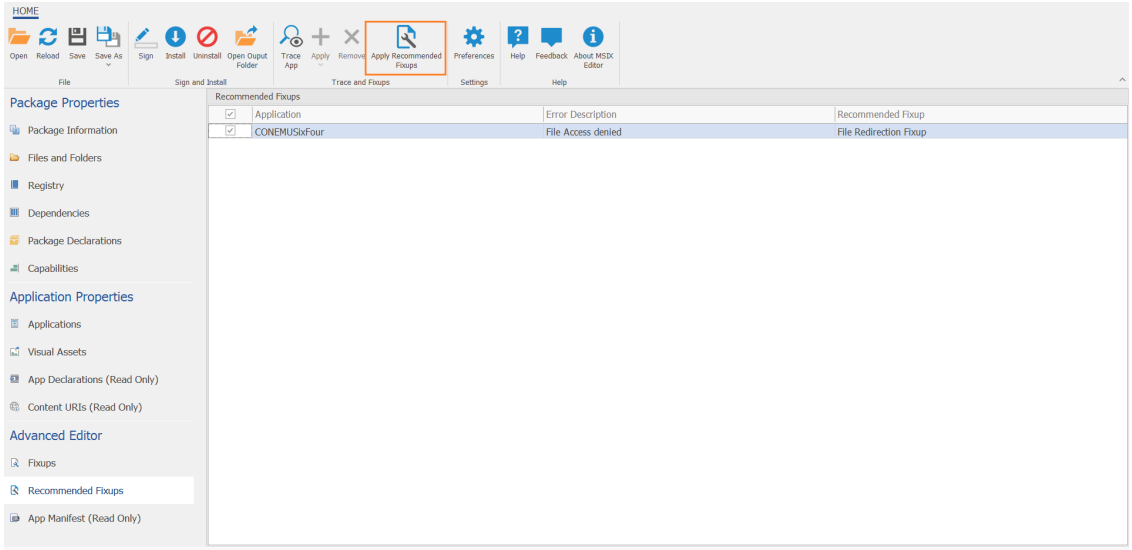
3. Application will install and a dialog box appears on the screen. Click **Launch** button to launch the application.



4. Perform the user acceptance testing. Application tracing logs will be displayed in the **Output** window while performing testing.

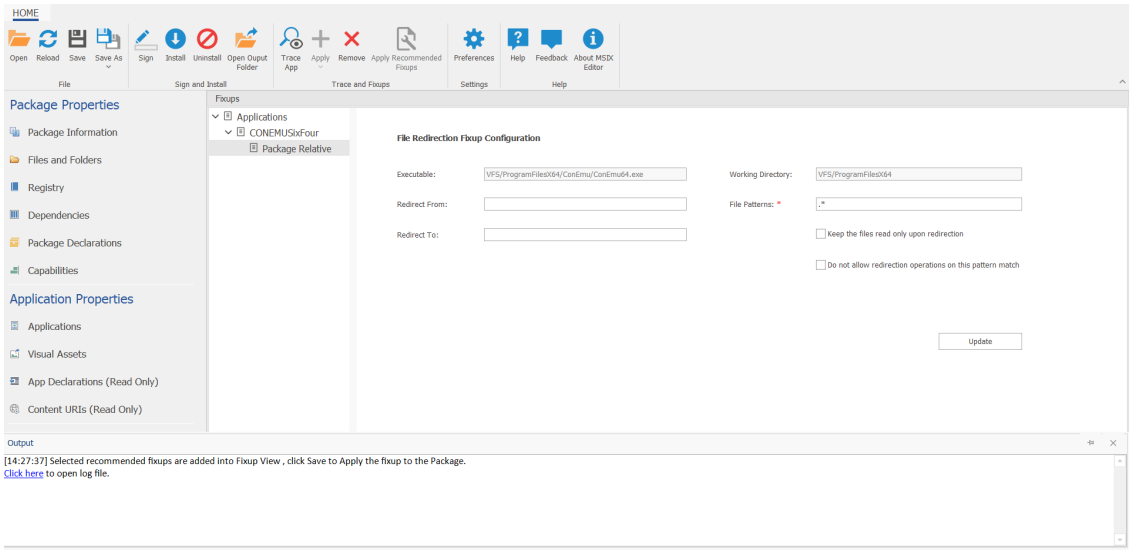
5. The **Recommended Fixups** view is auto populated with the fixup suggestions for the runtime issues encountered while testing.

6. Select the recommended fixups and click on **Apply Recommended Fixups** button in the ribbon.

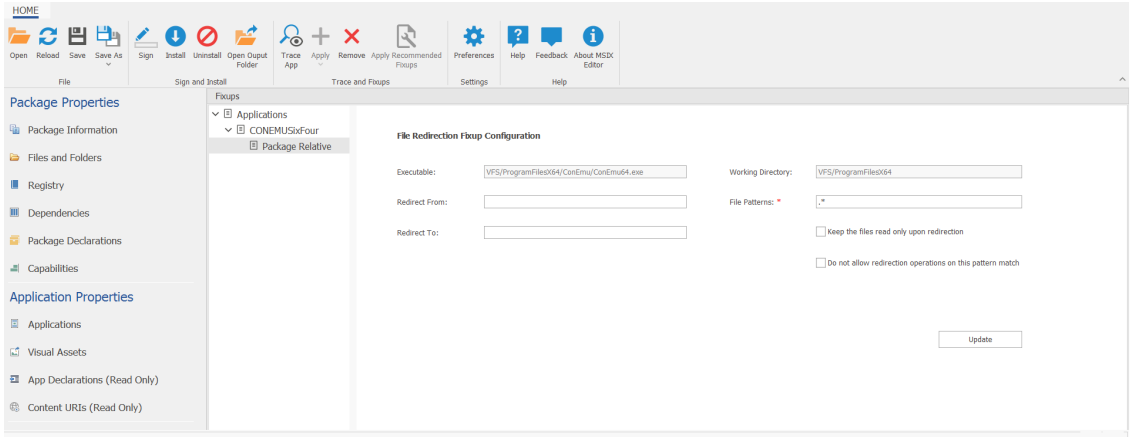


Note • *The Recommended Fixup button in MSIX Editor ribbon is enabled only when you select fixup in the Recommended Fixups view.*

7. Upon successful action, a message will be shown in the **Output** window.



8. Applied recommended fixups will be shown in the **Fixups** view for your review.



9. Click on **Save** or **Save As** to apply the recommended fixups to the MSIX package.

Create Modification Packages

A Modification package lets you customize your main MSIX package. Modification package is equivalent to a Transform (.mst) for an MSI package. The key feature of a modification package is that it is not tied to a specific version of its main MSIX package, which give you an advantage to create a modification package once and use it for all the current and future version of its main MSIX packages, unless you want to add any additional customizations.

Modification packages have the same extension as MSIX packages but contain only files and registry keys. Modification packages cannot be installed on a machine independently. The main MSIX package must be already installed on a machine to install its modification package on the same machine to apply customization successfully. When a modification package is installed, Windows will see its content as part of the same container as that of the main MSIX package thus treating every file and registry key in a modification package as part of the its main MSIX package.

- Specifying MSIX Package Information
- Including Files and Folders
- Including Registry
- Saving as Modification Package
- Signing Modification Package

Specifying MSIX Package Information

When you open an existing MSIX package in the MSIX Editor, you may need to view or specify important package information. This includes information such as the Package Name, Publisher Name, Version and others. In the Package Information view, you can view, and if appropriate, edit information about your MSIX package.

To configure package information for your MSIX package:

- In the left pane view list under **Package Properties**, click the **Package Information**.
- In the properties grid that appears in the right pane, configure the package properties as needed. For details about each property, see Package Information View.

Including Files and Folders

In the Files and Folders view you can make edits to the files and folders of your MSIX Package. You will be able to add new files and folders to your MSIX package and delete the files and folders from your MSIX package in this view.

- Adding a File or Folder to an MSIX Package
- Deleting a File or Folder From an MSIX Package

Including Registry

In Registry view you can make edits to the registries of the selected MSIX Package. You can perform the following actions under registry view:

- Importing a Registry File
- Adding a Registry Key
- Adding a Registry Value and Data to a Registry Key
- Modifying a Registry Value and Data of a Registry Key
- Deleting a Registry Key or Registry Value

Saving as Modification Package

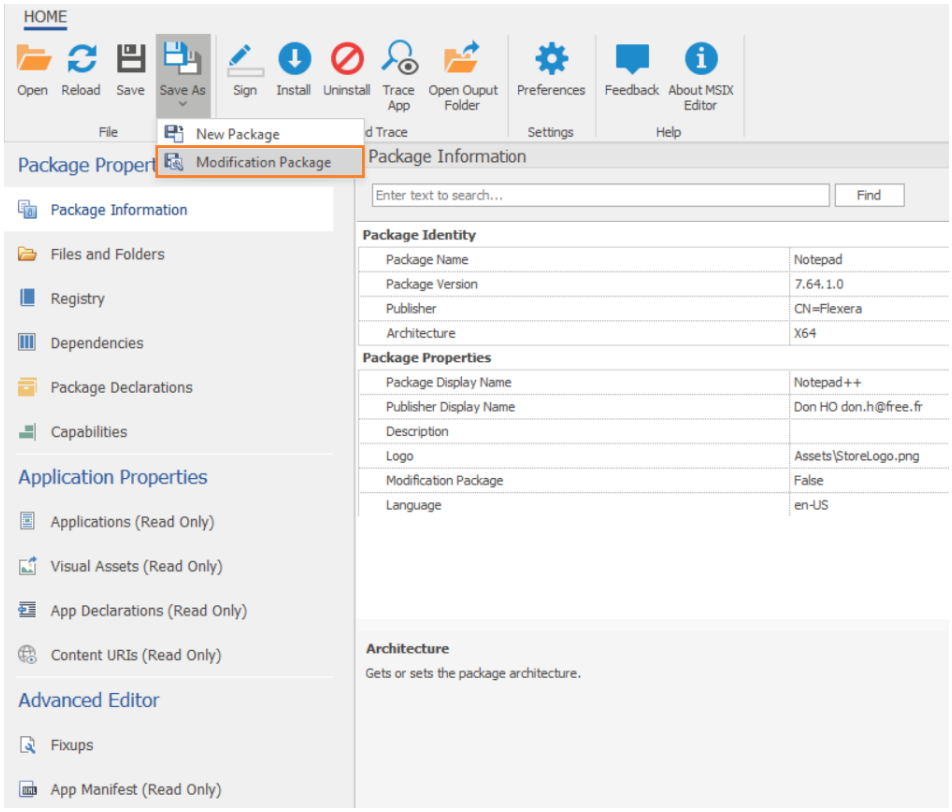
After editing the MSIX package, you can save the package as a modification package. To save a package as a modification package, perform the following steps:

Task ***To save as a Modification package:***

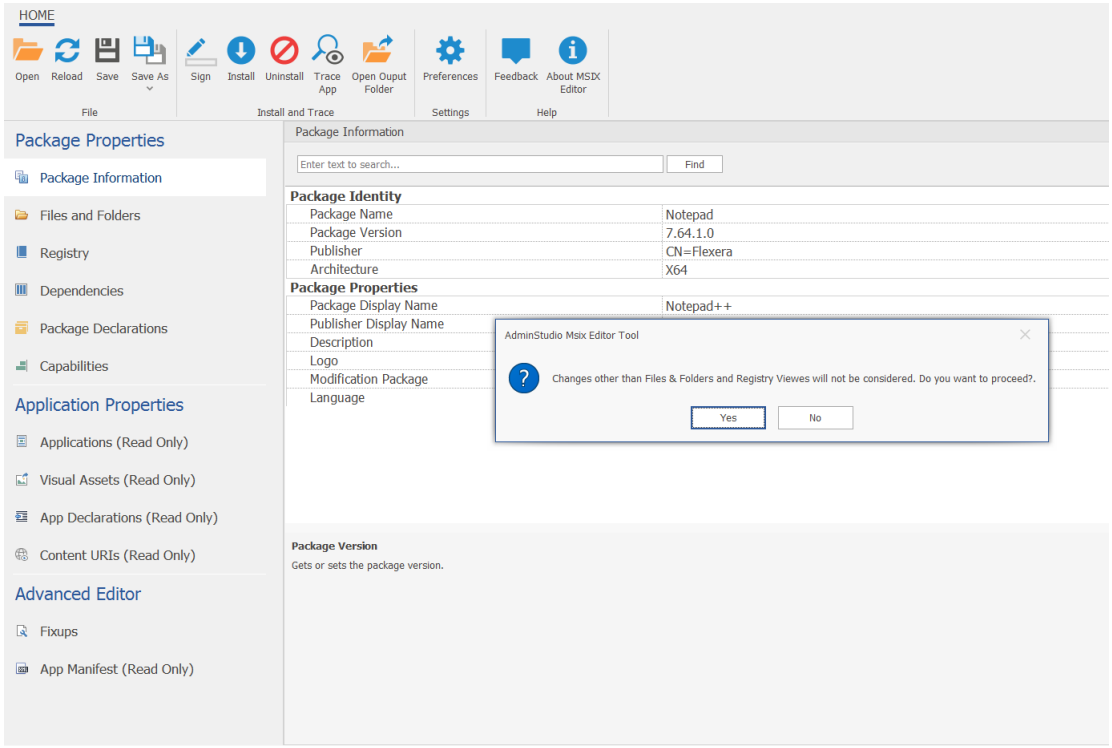
1. Open the desired MSIX Package in the MSIX Editor. For more information, see Opening an MSIX Package.

   Note • *Only* ***Files and Folders*** *and* ***Registry*** *view details are saved in the modification package.*

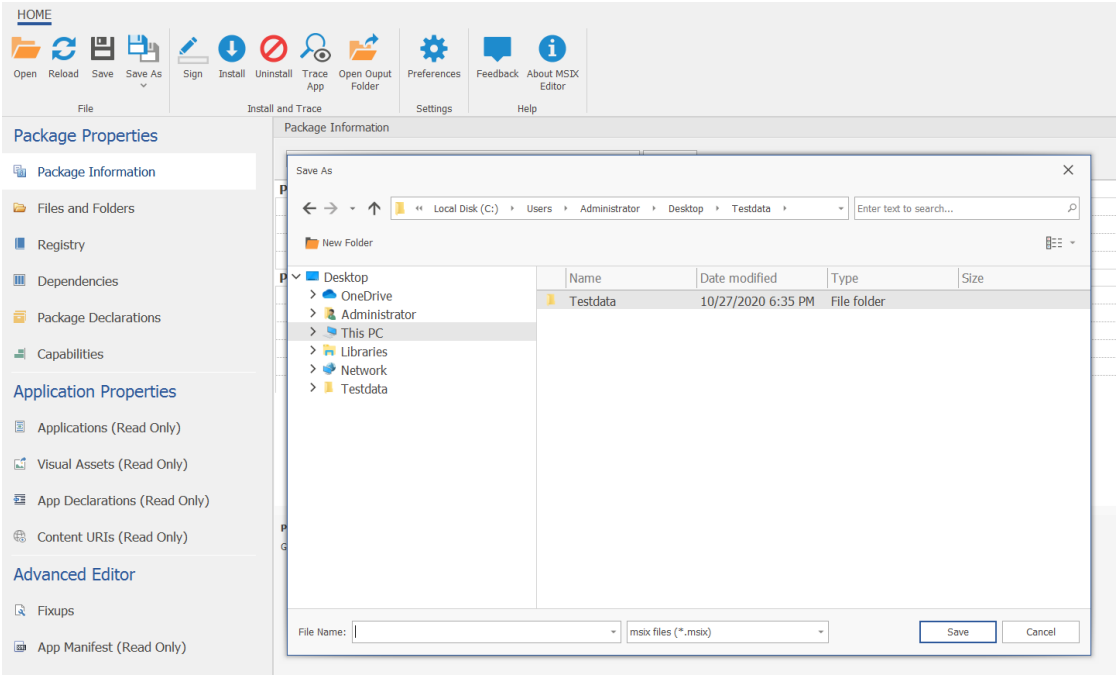
2. Modify the desired package properties for the selected MSIX Package. For more details, see Editing MSIX Packages.
3. After modifying, click the **Save As** button, and then select **Modification Package**.



4. Click **Yes** and confirm to proceed the changes.

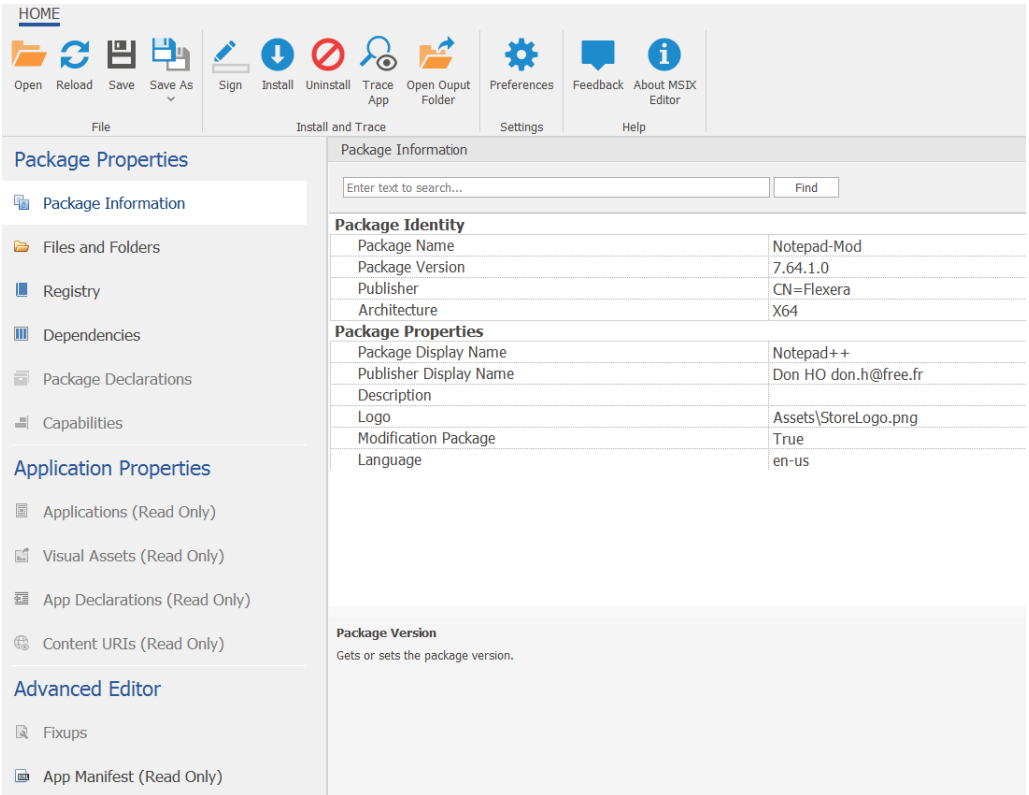


5. Enter the valid package name and then click the **Save** button to save as a modification package.



6. After saving, you can see the modification package loaded in the MSIX Editor.

Note • *In the modification package, only* ***Package Information***, ***Files and Folders***, ***Registry***, *and* ***Dependencies*** *views are enabled.*



Note • *You can not create a new modification package for the existing modification package.*

Signing Modification Package

MSIX packages must be digitally signed for successful installation. The MSIX Editor supports **Certificate File (.pfx)** and **Certificate Store** signing option.

MSIX Editor Reference

The Appxmanifest.xml is the most important file of the MSIX package. This is an XML file which defines all the important aspects of the package and its features. Appxmanifest.xml dictates the installation and run time behavior of the package. The MSIX Editor is organized into various views in the left pane which is based on the Appxmanifest.xml structure. This arrangement of the views makes the tool intuitive to easily identify and understand all the properties of an MSIX package to make edits.

The View List in the left pane is a navigational element that consists of views that you can click to open various areas within the MSIX Editor. The MSIX Editor views are grouped into the following three main views:

- Package Properties
- Application Properties
- Advanced Editor

Package Properties

The Package Properties view contains the views which have MSIX package properties defined and applicable at the package level. You can use the following views to configure MSIX package level properties:

- Package Information View
- Files and Folders View
- Registry View
- Dependencies View
- Package Declarations View
- Capabilities View

Package Information View

The properties displayed in the Package Information view are categorized in the following two categories:

- Package Identity
- Package Properties

Package Identity

Package Identity contains the properties which appear in the Package Identity tag of the appxmanifest.xml file. These properties uniquely identify the package. These properties are used to:

- Identify if the package is already installed on the machine
- Identify the main package to install the update package
- Identify the main package to install the modification package

Use the Package Identity category to specify the following MSIX package properties:

Table 11-7 • Package Identity Properties

Property	Description
Package Name	Package Name Uniquely identifies the package. It is case-sensitive and should be free text with alpha-numeric, period and dash characters with no spaces.
Package Version	Version number of the package. A version string should be of the format **Major.Minor.Build.Revision**. The last (fourth) section of the version number is reserved for Store use and must be left as 0 when saving the package.
Publisher	Name of the Publisher. The value must match with the Subject Name of the digital certificate file for successful signing.
Architecture	Describes the architecture of the code contained in the package.

Package Properties

Package Properties provides information about how the package appears to the user.

User Package Properties category to specify the following MSIX package properties:

Table 11-8 • Package Properties

Property	Description
Package Display Name	Friendly name for the package that appears in the Start menu and while installing the package.
Publisher Display Name	Friendly name for the publisher that appears in the Start menu and while installing the package.
Description	A friendly description that can be displayed to the users.
Logo	A path to a file which contains an image to be used as an icon for the package.
Modification Package	Shows value as True for Modification package else shows as False.

Table 11-8 • Package Properties

Property	Description
Language	Describes the language of the package.

Files and Folders View

The Files and Folders view is where you see files and folders within your MSIX package. This includes folders and files that are in the MSIX package's root folder, the virtual file system (VFS) folder, Assets, AppxMetadata and any other folders within MSIX package.

The Files and Folders view lets you make edits to the files and folders of your MSIX Package. You will be able to add new files and folders to your MSIX package and delete the files and folders from your MSIX package in this view.

The following four panes are available in the Files and Folders view:

Table 11-9 • Four Panes of the Files and Folders view

Pane	Description
Source computer's folders (Top Left)	The Source computer's folders pane on the top left is similar to the left pane in Windows Explorer. This pane contains folders located either locally or on a network. From here, you can navigate to the folder that contains the files that you want to add to your MSIX package.
Source computer's folder (Top Right)	The Source computer's folders pane is the top right displays the content of the currently selected folder of the Source computer's folders pane on the top left. You can drag files from this pane and drop into the Package Content bottom right pane to add to your MSIX package.
Package content (Bottom Left)	Package content pane on the bottom left displays all the folders in your MSIX package.
Package content (Bottom Right)	Package content pane on the bottom right displays the content of the currently selected folder of the Package content pane on the bottom left.

You can add files and folders to your installation the following two ways:

Table 11-10 • Method of Adding Files and Folders

Method	Description
Dragging and dropping	You can add files to your MSIX package using the Files explorer in the Files and Folders view. The top two panes in this view are functionally equivalent to Windows Explorer. The bottom two panes represent the destination for your files and folders. You can drag source files from the top pane to the destination folder in the bottom pane.

Table 11-10 • Method of Adding Files and Folders

Method	Description
Right click and browse	MSIX Editor provide a context menu in the Package Content pane on the bottom right. Click on Add File in the context menu of any folder in the Package Content pane on the bottom right to browse and add a file.

The following context menu options are available in Files and Folders view:

Table 11-11 • Context Menu options

Context Menu Option	Description
Expand	Expand the selected folder.
Full Expand	Fully expand the folder.
Collapse	Collapse the selected folder.
Full Collapse	Collapse entire folder and files.
Add File	Add a file to the folder selected.
Add Folder	Browse to add a new folder to an existing folder
Delete	Delete existing folder.

Below is the list of some commonly found system folders within any installation package and how they map to the folder withing virtual file system (VFS) of a MSIX package:

Table 11-12 • System Folder

System Folder	Equivalent folder in VFS
C:\Program Files (x86)	ProgramFilesX86
C:\Program Files	ProgramFilesX64
C:\Windows\System32	SystemX86
C:\Windows\SysWOW64	SystemX64
C:\Program Files (x86)\Common Files	ProgramFilesCommonX86
C:\Program Files\Common Files	ProgramFilesCommonX64
C:\Windows	Windows

Table 11-12 • System Folder

System Folder	Equivalent folder in VFS
C:\ProgramData	Common AppData

Registry View

The Registry view enables you to view registry keys, values, and data that exist in for your MSIX package. This view also lets you add, delete, and modify the registry keys of the package.

An MSIX package typically contains the following .dat files which hold the registry information of the package:

- Registry.dat
- User.dat
- User.Classes.dat

The Regirsty.dat file maps to the HKLM which per-machine registry entries while the per-user registry entries are mapped to the other .dat files. These are virtualized registry keys and only the MSIX package will be able to read them. These keys will not be visible in the client system registry. Note that the registry entries that are configured in the Registry view affect only your MSIX package. They do not affect any applications that are installed to the client system.

The Registry view contains the following two panes:

Table 11-13 • Registry View Panes

Pane	Description
Package Content (Middle Pane)	The Package Content pane in the middle displays all the registry keys within your MSIX Package. It displays the following two hives: • Machine • User Machine hive contains per-machine registry keys. User hive contains per-user registry keys.
Package Content (Right Pane)	The Package Content pane in the right displays the registry values of the currently selected registry key in the Package Content middle pane.

Dependencies View

The Dependencies view displays all the dependencies your MSIX package to make it fully functional. You can specify the following package level dependencies in MSIX Editor:

- Target Device Family
- Package Dependency
- Main Package Dependency
- Driver Dependency

- OS Package Dependency

Target Device Family

Describes the device family that your MSIX package targets. Specify the following attributes for this dependency:

Table 11-14 • Attributes for the Target Device Family

Attributes	Description
Name	The name of the device family that your app is targeting. For a packaged MSIX package, this is set to Windows.Desktop. Examples: Windows.Desktop, Windows.Universal, Windows.Mobile
Min Version	The minimum version of the device family that your app is targeting. Used for applicability at deployment time. If the device family version of the system is lower than MinVersion, then the app is not considered applicable
Max Version Tested	The maximum version of the device family that your package is targeting that you have tested it against. This is used at runtime to determine the effective process space for quirks. If it's a vendor supplied MSIX, then this is version that the developer has tested the package against.

Package Dependency

Declares a dependency on another package that is marked as a framework package. Specify the following attributes for this dependency:

Table 11-15 • Attributes for the Package Dependency

Attributes	Description
Name	The name as it appears in the Name attribute of the Identity element of the dependency package.
Publisher	The publisher as it appears in the Publisher attribute of the Identity element of the dependency package.
Min Version	The minimum version of the dependency package.
Max Major Version Tested	The maximum version of the dependency package tested against. Used to determine whether frameworks will be staged side-by-side, and what framework gets loaded into the package graph for the package.
Optional	Indicates that a framework package dependency is optional for the app, meaning the app can be installed even if the optional framework dependencies are not installed.

Main Package Dependency

Describes the main app package to which this supplemental package applies. Applicable for Modification packages. Specify the following attributes for this dependency:

Table 11-16 • Attributes for the Main Package Dependency

Attributes	Description
Name	The name as it appears in the Name attribute of the Identity element of the dependency package.
Publisher	The publisher as it appears in the Publisher attribute of the Identity element of the dependency package.

Driver Dependency

Describes the driver dependencies of the package. If a package has a driver dependency, then the driver must be present for the app to load. A driver dependency should be specified along with the derive constraint. Specify the following attributes for driver constraint:

Table 11-17 • Attributes for the Driver Dependency

Attributes	Description
Name	The name of the driver in the form: <Provider>-<Filename>.INF.
Min Version	The version of the driver package.
Min Date	The date of the driver package.

OS Package Dependency

Describes OS Package Dependency of the package. Specify the following attributes for this dependency:

Table 11-18 • Attributes for the OS Package Dependency

Attributes	Description
Name	The name of the package dependency.
Version	The version of the package dependency.

Package Declarations View

A Declaration (also referred to as extension) is like an agreement between an app and the Operating System. Declaration is a way to add extensibility points to an app to interact with Operating System or other apps. In the Package Declarations view you can see the all the declarations defined in your MSIX package.

You can specify the following package level dependencies in MSIX Editor:

- Firewall Rules
- Com Interface
- Loader Search Path Override

Firewall Rules

Define a firewall exception rule using the following attributes:

Table 11-19 • Firewall Rules

Attributes	Description
Direction	Select the direction either In or Out.
IP Protocol	Click IP Protocol drop down and select the desired from the list.
Local Port Min	Enter local port minimum in the text field.
Profile	Click Profile drop down and select the desired from the list.
Local Port Max	Enter local port maximum in the text field.
Remote Port Min	Enter remote port minimum in the text field.
Remote Port Max	Enter remote port maximum in the text field.

Com Interface

Describes a package extension point of type windows.comInterface. The comInterface extension may include the following three types of registrations:

- Proxy Stub
- Interface
- Type Lib

Proxy Stub

Register a proxy stub using the following attributes:

Table 11-20 • Register Proxy Stub

Attributes	Description
ID	The proxy stub's CLSID.

Table 11-20 • Register Proxy Stub

Attributes	Description
Display Name	A localizable string corresponding to the default value of the proxy stub's CLSID key.
Path	The path relative to the package root. Path must reference a file in the package.
Proxy Stub Dll Path	A relative path to the .dll file in the app package.
Process Architecture	The processor architecture of the Proxy Stub registration.

Interface

Register an interface using the following attributes:

Table 11-21 • Register an Interface

Attributes	Description
ID	An interface Id (IID).
Use Universal Marsheller	Set this to true to use the OLE Universal Marshaler as the proxy stub.
Proxy Stub CLSID	Corresponds to the Proxy Stub CLSID registry value.
Synchronous interface	The Id of another interface registration containing AsynchronousInterface that references this registration. The other interface must be in the same comInterface registration.
Asynchronous interface	The Id of another interface registration containing SynchronousInterface that references this registration. The other interface must be in the same comInterface registration.
Type Lib ID	The type library ID.
Type Lib Version Number	The version of the type library.

Type Lib

Register a type lib using the following attributes:

Table 11-22 • Register a Type Lib

Attributes	Description
ID	The ID of the type library.

Loader Search Path Override

An extension that allows an app developer to declare a path in the app package, relative to the app package root path, to be included in the loader search path for the app's processes. Specify the following attributes for this package declaration:

Table 11-23 • Attributes for Load Search Path Override

Attributes	Description
FolderPath	A relative path used specify where to load content from an app package.

Capabilities View

Capabilities provide apps with access to certain Windows APIs and resources, such as pictures, music or devices such as the camera or the microphone. These capabilities are declared in the package manifest file. This view lists all the supported Capabilities for MSIX package. You can select or deselect any capability to add or remove it from the package manifest file.

Capabilities are used by UWP apps as well as other types of desktop apps that are packaged in an MSIX.

There are several types of capabilities:

- Device Capabilities
- General Capabilities
- Restricted Capabilities

You can check or uncheck the desired capabilities by clicking on the respective check box button.

Note *• Making changes to Capabilities will not be allowed for modification packages.*

Device Capabilities

Device capabilities allow your app to access peripheral and internal devices. Device capabilities are specified by using DeviceCapability elements in your app package manifest.

Table 11-24 • Device Capabilities

Option	Description
Bluetooth	The **bluetooth** device capability allows apps to communicate with already paired bluetooth devices over both Generic Attribute (GATT) or Classic Basic Rate (RFCOMM) protocol.
Eye Tracker	The gazeInput capability allows apps to detect where the user is looking within the application bounds when a compatible eye tracking device is connected.
Location	The location capability provides access to location functionality that is retrieved from dedicated hardware like a GPS sensor in the PC or is derived from available network info.

Table 11-24 • Device Capabilities

Option	Description
GPIO, I2C, SPI, and PWM (Low Level)	The lowLevel device capability provides access to GPIO, I2C, SPI, and PWM devices.
Microphone	The microphone capability provides access to the microphone's audio feed, which allows the app to record audio from connected microphones.
Point of Service (POS)	The pointOfService device capability enables access to APIs. This APIs provide a vendor-neutral interface for accessing POS devices from various manufacturers from a UWP app.
Proximity	The proximity capability enables multiple devices in close proximity to communicate with one another. This capability is typically used in casual multi-player games and in apps that exchange information. Devices attempt to use the communication technology that provides the best possible connection, including Bluetooth, Wi-Fi, and the Internet. This capability is used only to initiate communication between the devices.
Webcam	The webcam capability provides access to the video feed of a built-in camera or external webcam, which allows the app to capture photos and videos. The webcam capability only grants access to the video stream. In order to grant access to the audio stream as well, the microphone capability must be added.

General Capabilities

General-use capabilities are specified by using Capability elements in your app package manifest. These capabilities apply to the most common app scenarios.

Table 11-25 • General Capabilities

Option	Description
3D Objects	The objects3D capability allows apps to have programmatic access to the 3D object files. This capability is typically used in 3D apps and games that need access to the entire 3D objects library.
AllJoyn	The allJoyn capability allows AllJoyn-enabled apps and devices on a network to discover and interact with each other.
Appointments	The appointments capability provides access to the user's appointment store. This capability allows read access to appointments obtained from the synced network accounts and to other apps that write to the appointment store. With this capability, your app can create new calendars and write appointments to calendars that it creates.

Table 11-25 • General Capabilities

Option	Description
Background Media Playback	The **Background Media Playback** capability changes the behavior of the media-specific APIs to enable media playback while your app is in the background. All active audio streams will no longer mute, but will continue to be audible when an app transitions to the background. Additionally, app lifetime will be extended automatically while playback is occurring.
Read Blocked Messages	The blockedChatMessages capability allows apps to read SMS and MMS messages that have been blocked by the Spam Filter app.
Code generation	The Code Generation capability allows apps to access the functions which provide JIT capabilities to apps.
Contacts	The contacts capability provides access to the aggregated view of the contacts from various contacts stores. This capability gives the app limited access (network permitting rules apply) to contacts that were synced from various networks and the local contact store.
Internet and public networks	There are two capabilities that provide different levels of access to the Internet and public networks: • The internetClient capability indicates that apps can receive incoming data from the Internet. Cannot act as a server. No local network access. • The internetClientServer capability indicates that apps can receive incoming data from the Internet. Can act as a server. No local network access.
Homes and work networks	The privateNetworkClientServer capability provides inbound and outbound access to home and work networks through the firewall. This capability is typically used for games that communicate across the local area network (LAN), and for apps that share data across a variety of local devices. If your app specifies musicLibrary, picturesLibrary, or videosLibrary, you don't need to use this capability to access the corresponding library in a Home Group. On Windows, this capability does not provide access to the Internet.
Music Library	The musicLibrary capability provides programmatic access to the user's Music library, allowing the app to enumerate and access all files in the library without user interaction. This capability is typically used in jukebox apps that make use of the entire Music library.
Phone Calls	The Phone Call capability allows apps to access all of the phone lines on the device and perform the following functions
Phone Calls History Public	The Phone Calls history Public allows app to read cellular and some VoIP call history information on the device. This capabilities also allows the app to write VoIP call history entries.

Table 11-25 • General Capabilities

Option	Description
Pictures library	The **Pictures Library** capability provides programmatic access to the user's Pictures library, allowing the app to enumerate and access all files in the library without user interaction. This capability is typically used in photo apps that make use of the entire Pictures library.
Remote System	The **Remote System** capability allows apps to have access to a list of devices associated with the user's account. Access to the device list is necessary to perform any operations that persist across devices.
Removable Storage	The **Removable Storage** capability provides programmatic access to files on removable storage, like USB keys and external hard drives, filtered to the file-type associations declared in the package manifest. For example: If a document-reader app declares a .doc file-type association, it can open .doc files on the removable storage device, but not other types of files.
Spatial Perception	The Spatial Perception capability provides programmatic access to spatial mapping data, giving mixed reality apps information about surfaces in application-specified regions of space near the user. Declare the spatial perception capability only when your app will explicitly use these surface meshes, as the capability is not required for mixed reality apps to perform holographic rendering based on the user's head pose.
User Account Information	The **User Account Information** capability gives apps the ability to access the user's name and picture.
Videos library	The **Videos Library** capability provides programmatic access to the user's Videos, allowing the app to enumerate and access all files in the library without user interaction. This capability is typically used in movie-playback apps that make use of the entire Videos library.
VoIP Calling	The voipCall capability allows apps to access the VoIP calling.

Restricted Capabilities

If your app declares any restricted capabilities, you must provide info in order to be approved to publish the app.

The following table lists the restricted capabilities:

Table 11-26 • Restricted Capabilities

Option	Description
Elevation	The **Elevation** restricted capability allows apps that are created by Flexera partners and enterprises to preserve existing desktop functionality that requires auto-elevation on launch or during an app's lifetime.
Enterprise Authentication	The **Enterprise Authentication** capability is typically used in line-of-business apps that connect to servers within an enterprise. You donot need this capability for generic communication across the Internet.
Full Trust Permission Level	The **Full Trust Permission Level** restricted capability allows apps to run at the full trust permission level on the user's machine.
Shared User Certificates	The **Shared User Certificates** capability enables an app to add and access software and hardware-based certificates in the Shared User store, such as certificates stored on a smart card. This capability is typically used for financial or enterprise apps that require a smart card for authentication.
Package Query	The **Package Query** device capability allows apps to gather information about other apps.
Package Management	The **Package Management** restricted capability allows app to manage other apps directly.
Local System Services	The **Local System Services** restricted capability allows applications that are created by Flexera partners and enterprises to install one or more Local System services along with the app (that is, your application can declare the StartAccount for the services to be LocalSystem). This scenario also requires the packagesServices capability.
Package Services	The **Package Services** restricted capability allows applications that are created by Flexera partners and enterprises to declare the extension in its package manifest so that it can install one or more services along with the app. These services can be configured to run under the Local Service, Network Service or Local System accounts. Local Service and Network Service services only require the packageServices capability. Local System services require both the packageServices and localSystemServices capabilities.

Application Properties

The Application Properties view contains the views which have MSIX package properties defined and applicable at the application level of the package. You can use the following views to configure MSIX application level properties:

- Applications View
- Visual Assets View

- App Declarations View
- Content URIs View

Note • *Only **Applications View** and **Visual Assets** view are editable. Remaining views under **Application Properties** are currently read-only.*

Applications View

Applications are the entry points into the MSIX package. These work like shortcuts to launch the application.

Specify the following attributes for an Application:

Table 11-27 • Application View Attributes

Attributes	Description
Id	The unique identifier of the application within the package. This value is sometimes referred to as the package-relative app identifier (PRAID). The ID is unique within the package but not globally. There may be another package on the system that uses the same ID. The same ID cannot be used more than once in the same package. This string contains alpha-numeric fields separated by periods. Each field must begin with an ASCII alphabetic character. You cannot use these as field values: "CON", "PRN", "AUX", "NUL", "COM1", "COM2", "COM3", "COM4", "COM5", "COM6", "COM7", "COM8", "COM9", "LPT1", "LPT2", "LPT3", "LPT4", "LPT5", "LPT6", "LPT7", "LPT8", and "LPT9".
Executable	The default launch executable for the app. This file must be present in the package. If you specify this attribute you must specify the **EntryPoint** attribute. If you specify this attribute you must not specify the **StartPage** attribute.
Entry Point	The activatable class ID, such as ""Office.Winword.Class". If you specify this attribute, you must also specify the **Executable** attribute. If you specify this attribute you must not specify the **StartPage** attribute.
Start Page	The default launch HTML page for the app. This can be a relative Windows file path referencing a document in the app's package, or it can be an absolute URL. The URL can only be starting with http://, https:// or ms-appx-web://. This is the entry point document that will be loaded by WWAHost when starting a WWA for that app. Technically, the value may be a URL or an IRI-the non-ASCII version of a URI. An IRI must support up to 2084 characters and must be allowed to contain the %, and reserved and unreserved characters. If you specify this attribute, you cannot specify either the **EntryPoint** attribute or the **Executable** attribute.

Table 11-27 • Application View Attributes

Attributes	Description
Resource Group	ResourceGroup is a tag that you can use to group extension activations together for resource management purposes (for example, CPU and memory). If you set Support Multiple Instances to True, this property will be disabled.
Subsystem	Indicates whether the app is a standard UWP app or a UWP Console App. If you select **Console App**, then **Support Multiple Instances** has to be True.
Support Multiple Instances	Indicates support of multiple, separate instances of UWP apps.

Specify the following attributes for an Non-EXE Application:

Table 11-28 • Non-EXE Application Attributes

Attributes	Description
ID	The unique identifier of the application within the package. This value is sometimes referred to as the package-relative app identifier (PRAID). The ID is unique within the package but not globally. There may be another package on the system that uses the same ID. The same ID cannot be used more than once in the same package. This string contains alpha-numeric fields separated by periods. Each field must begin with an ASCII alphabetic character. You cannot use these as field values: "CON", "PRN", "AUX", "NUL", "COM1", "COM2", "COM3", "COM4", "COM5", "COM6", "COM7", "COM8", "COM9", "LPT1", "LPT2", "LPT3", "LPT4", "LPT5", "LPT6", "LPT7", "LPT8", and "LPT9".
File	Click ellipses(…) and select the file (.bat, .cmd, .txt, .html, .docx) that you want the application to point to launch while launching. This is a mandatory property.
Argument	Specify the command-line arguments that you wish to pass to the file specified in the File property. The Argument is property is applicable only for the .bat and .cmd file types. This is a mandatory property.

Visual Assets View

Visual Assets comprise of different tile sizes, application icon, package logo, splash screen and badge logo. Upon its installation, MISX surfaces on a Windows machine through these visual assets, for example as tiles and/or application icon on the start menu. Unlike traditional MSI icons, Visual Assets provide flexibility to specify different sizes of tiles and icons with different images and scales. The Visual Assets view enables you to view, edit and generate new visual assets for an MSIX package.

The Visual Assets view has the following configurations:

Table 11-29 • Visual Assets Details

Asset Name	Description	Asset File Name
Display Settings	Describes the visual aspects of the Windows app package like Display Name, Short Name, Show Name, Description, Tile Background, Splash Screen Background, Supported Rotations, Lock Screen Notification, Recurrence, and Url Template on Tiles.	
Small Tile	Specify a relative path to a small-sized (71x71) tile background image.	Square71x71Logo.png
Medium Tile	Specify a relative path to a medium-sized (150x150) tile background image.	Square150x150Logo.png
Wide Tile	Specify a 310x150 image file to be used for a wide tile, only when pinned and resized. If specified, this will allow the tile to be set to Wide.	Wide310x150Logo.png
Large Tile	Specify a 310x310 image file to be used for a large tile, only when pinned and resized. If specified, this will allow the tile to be set to Large.	Square310x310Logo.png
App Icon	Specify a 44x44 image file to use as a small logo icon that is displayed in the left column of the Start Menu's All Apps list.	Square44x44Logo.png
Splash Screen	Specify a image that defines the appearance of the splash screen, which is displayed by the app during launch.	SplashScreen.png
Badge Logo	Specify a logo image that is shown next to the badge to identify the app. This image must be monochromatic, of type .png, and measure 24x24 pixels.	BadgeLogo.png
Package Logo	Specify a image appears in App installer, Partner Center, the "Report an app" option in the Store, the "Write a review" option in the Store.	StoreLogo.png

Visual Asset Generator

In the Visual Asset Generator, you can see the following details:

Table 11-30 • Visual Asset Generator Details

Field	Description
Source	Browse and specify the source path of the file.
Visual Assets	Specify visual assets for the selected source file.
Tile Background	Specify the hex value or a color name for the tile background color.
Splash screen Background	Specify the hex value or color name for the splash screen background. ***Note*** • *If you want to specify Splash screen Background color, Splash screen image should be provided.*

Display Settings

In the Display Settings, you can see the following details:

Table 11-31 • Display Settings Details

Field	Description
Display Name	Specify name of the Application. This is a mandatory field.
Short Name	Specify short name of the Application.
Show Name	Specify the short name to be displayed on one of the following: • Medium • Wide • Large ***Note*** • *Respective tile image should be specified before selecting these options.* ***Note*** • *Multiple selections are allowed.*
Description	Specify description of the Application. This is a mandatory field.

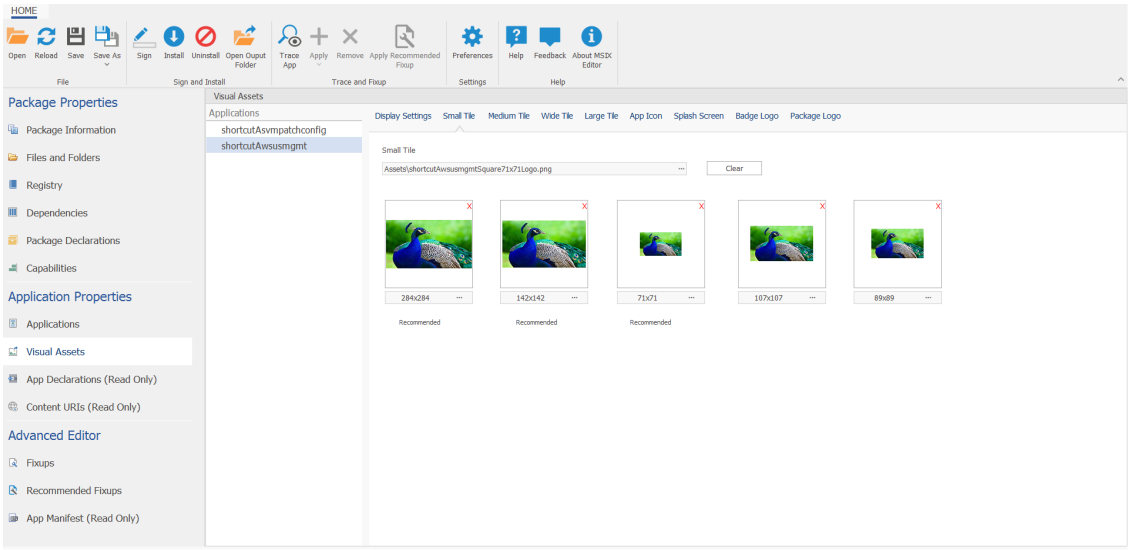
Table 11-31 • Display Settings Details

Field	Description
Supported Rotations	Displays in one of the following supported rotations. Select the desired options: • Landscape • Portrait • Landscape-flipped • Portrait-flipped ***Note*** • *Multiple selections are allowed.*
Lock Screen Notification	The lock screen notification setup for the application. Select one of the following: • Badge • Badge And Tile Text ***Note*** • *If you select **Badge** option then **Badge Logo** image should be provided, and vise versa.* *If you select **Badge And Tile Text** option then both **Badge Logo** and **Wide Tile** images should be provided.*
Recurrence	Shows the recurrence of the application. Select one of the following: • Half Hour • Hour • Six Hours • Twelve Hour • Daily
Url Template	Specify the valid Url template. ***Note*** • *This field will be enabled only when you select **Recurrence** type.*

Small Tile

The small square version of the logo image.

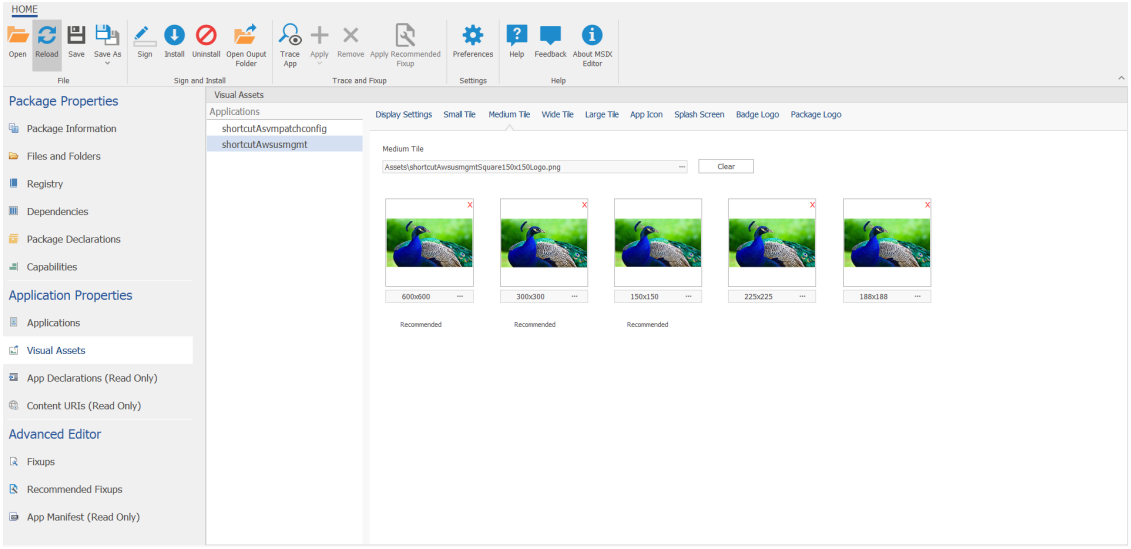
Click the ellipses (...) to browse and select a image to a small-sized tile background. You can also select image for the recommended size type at the image-level.



Medium Tile

A medium-size image used as the app's square tile.

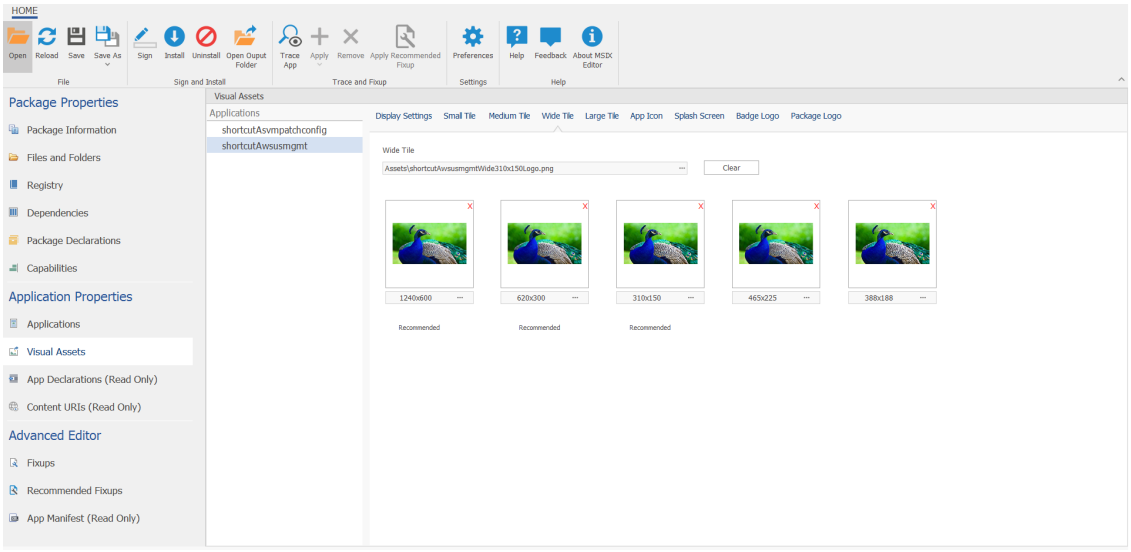
Click the ellipses (...) to browse and select a image to a medium-sized tile background. You can also select image for the recommended size type at the image-level.



Wide Tile

These images are displayed when the tile is rendered in its wide format.

Click the ellipses (...) to browse and select a image to a wide-sized tile background. You can also select image for the recommended size type at the image-level.

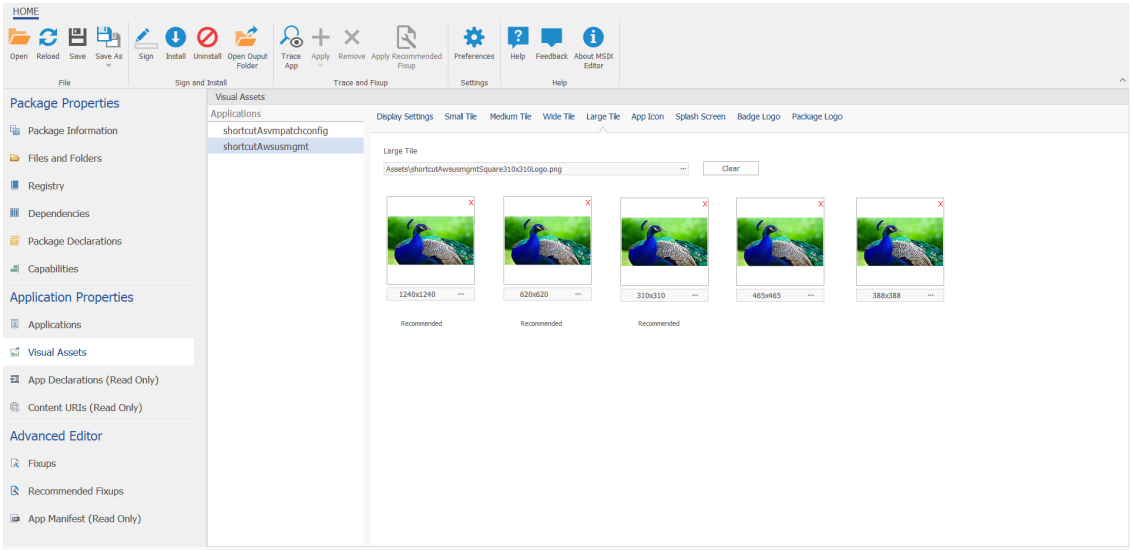


Large Tile

The large square version of the logo image.

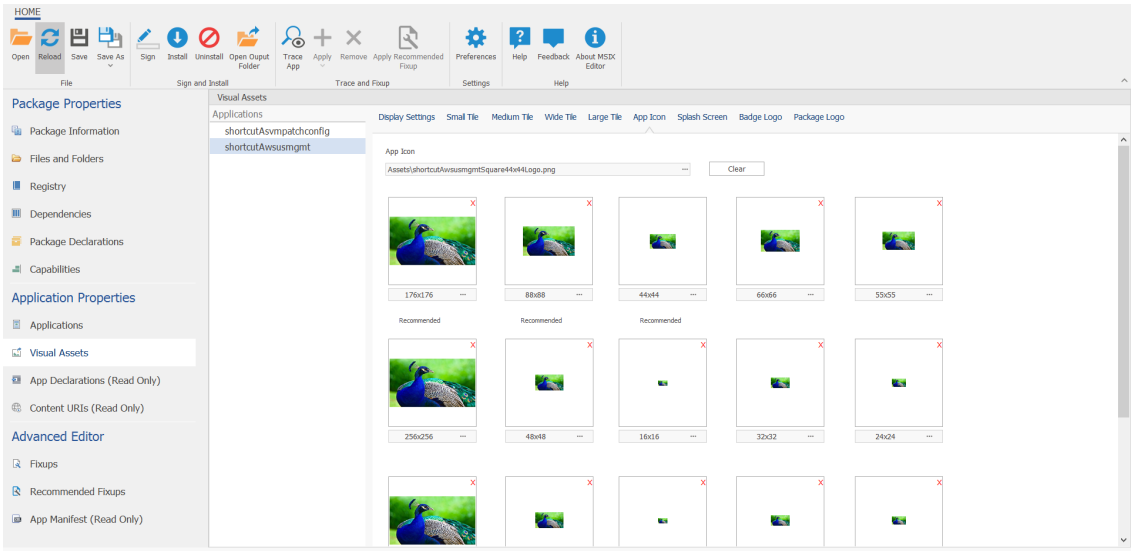
Click the ellipses (...) to browse and select a image to a large-sized tile background. You can also select image for the recommended size type at the image-level.

Note • *If you select **Large Tile** logo image then **Wide Tile** image is mandatory.*



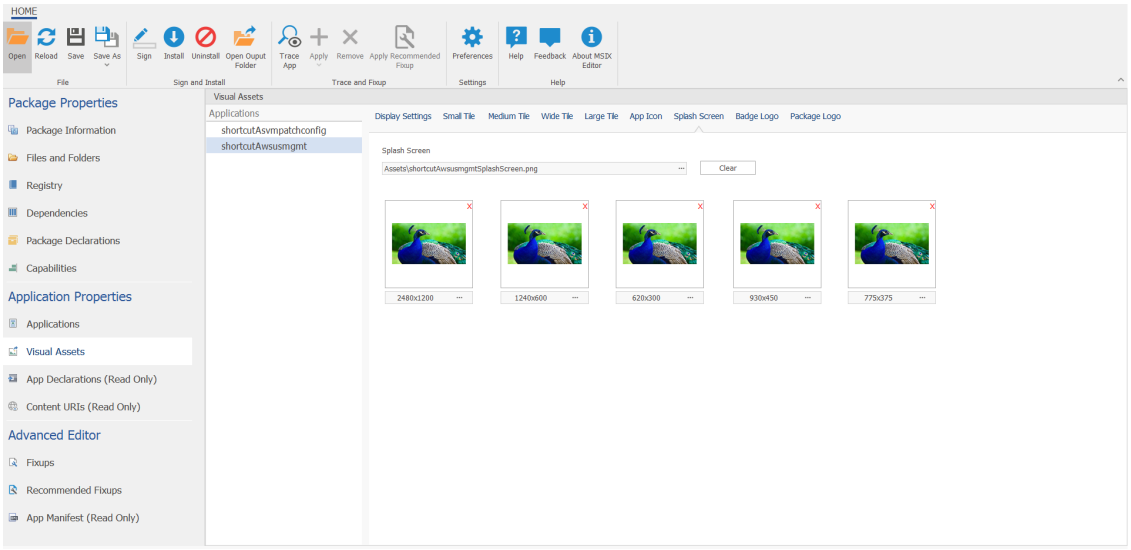
App Icon

Click the ellipses (...) to browse and select a image to app icon background. You can also select image for the recommended size type at the image-level.



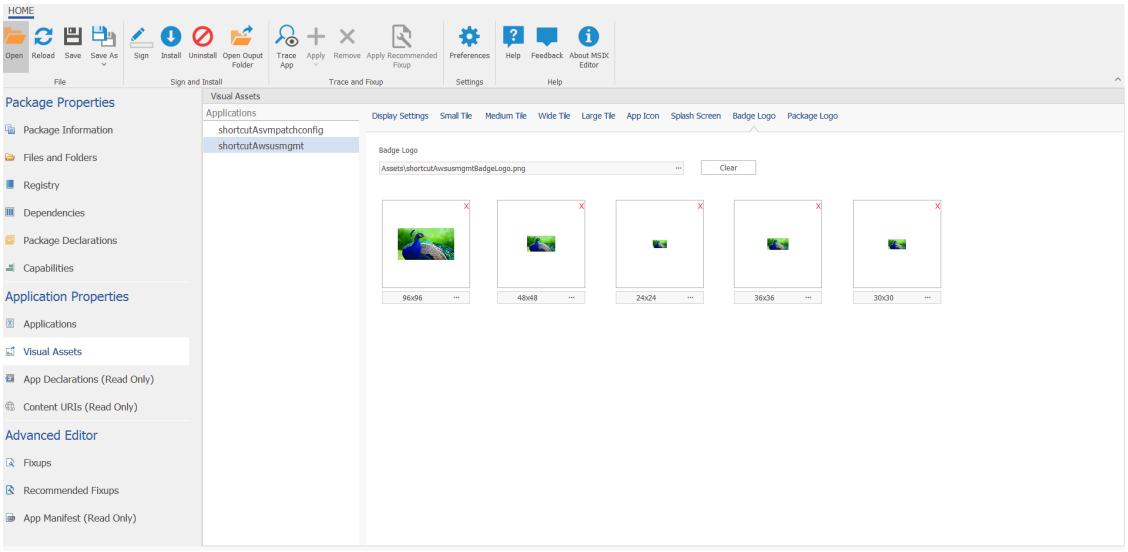
Splash Screen

Click the ellipses (...) to browse and select a image to Splash Screen background. You can also select image for the recommended size type at the image-level.



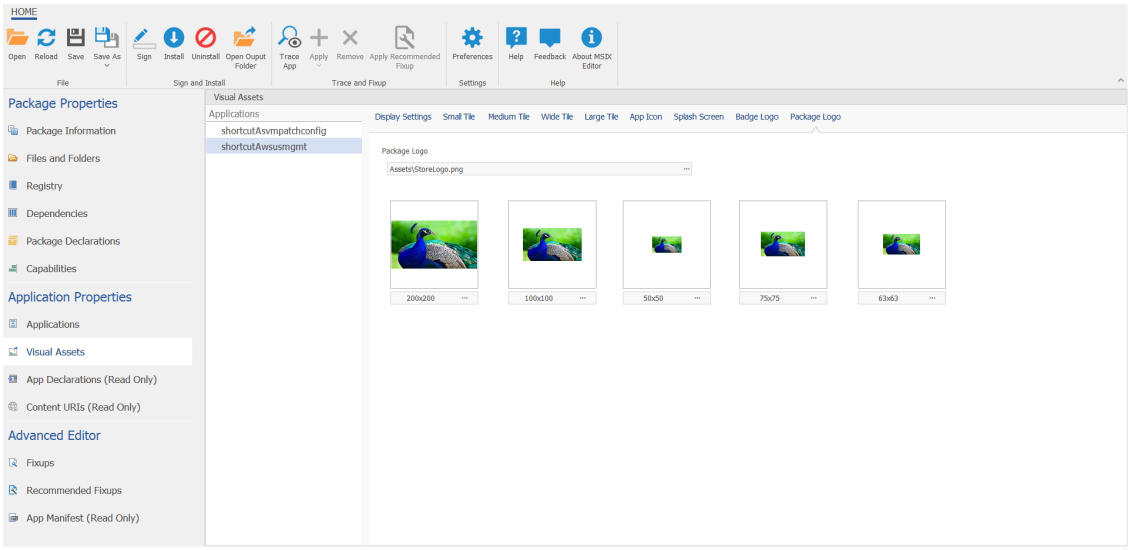
Badge Logo

Click the ellipses (...) to browse and select a image to Badge logo. You can also select image for the recommended size type at the image-level.



Package Logo

Click the ellipses (...) to browse and select a image to Package logo. You can also select image for the recommended size type at the image-level.



App Declarations View

A Declaration (also referred to as extension) is like an agreement between an app and the Operating System. Declaration is a way to add extensibility points to an app to interact with Operating System or other apps. In the App Declarations view you can see the all the declarations defined in your MSIX package.

Note • *This view is currently read-only.*

Content URIs View

Use this view to specify the URIs that can use window.external.notify to send a Script-Notify event to the app. URIs can include wild-card characters in sub-domain names (for example, https://*.flexera.com or https://*.*.flexera.com)

The Content URIs view specifies the pages in the web context that have access to the system's geolocation devices (if the app has permission to access this capability) and access to the clipboard.

If more than one rule is defined, then the order of the rules is important.

To define the Match attribute with an IRI for a web resource, you can specify only secure "https:" sites - unsecure "http:" sites are not allowed.

Note • *This view is currently read-only.*

Advanced Editor

Advanced Properties view facilitates advanced editing of your MSIX package with the following views:

- Fixups View
- Recommended Fixups View
- App Manifest View

Fixups View

Package Support Framework (PSF) provides a way to identify and fix certain run time issues of your MSIX package when you do not have access to this source code. PSF is a light weight, open source kit which helps your application follow the best practices for modern runtime environment.

A fix offered by Package Support Framework (PSF) for commonly seen run time issues is generally referred as a Fixup.

MSIX package must be traced using MSIX Editor to identify run time issues. The Fixup view provide an easy way to apply fixups to your MSIX package to fix the run time issues identified during trace.

The following fixups can be applied to an MSIX package using MSIX Editor.

File Redirection Fixup (FRF)

For more information on Package Support Framework, see here.

For more information on applying fixups, see here.

Recommended Fixups View

The run time issues identified during the trace will be displayed in the Recommended Fixup view. Apply recommended fixups to your MSIX package to fix the run time issues automatically which were identified during the trace.

App Manifest View

The Appxmanifest.xml is the most important file of the MSIX package. This is an XML file which defines all the important aspects of the package and its features. Appxmanifest.xml dictates the installation and run time behavior of the package. The App Manifest view lets you make edits to the Appxmanifest.xml file of the MSIX package.

Note • *This view is currently read-only.*

12

Using the Virtual Package Editor

The Virtual Package Editor is a powerful tool that lets you edit App-V packages and perform tasks such as the following:

- Customize your App-V applications.
- Resolve virtualization best practice issues and application conflicts.
- Fix run-time problems.

The Virtual Package Editor documentation contains the following sections:

Table 12-1 • AdminStudio Virtual Package Editor Documentation Sections

Section	Description
About Virtualization	Provides background information about virtualization.
About the Virtual Package Editor	Introduces some basic concepts to help you get started with editing a virtual package.
Getting Started with the Virtual Package Editor	Contains information to help you become familiar with the Virtual Package Editor, begin editing a virtual package, and customize the Virtual Package Editor user interface.
Editing Virtual Packages	Explains how to edit virtual packages and guides you through every step of the process.
Virtual Package Editor Reference	Contains comprehensive reference information for the Virtual Package Editor user interface.

Contacting Us

Flexera is headquartered in Itasca, Illinois, and has offices worldwide. To contact us or to learn more about our products, visit our website at:

http://www.flexera.com

About Virtualization

Virtualization enables you to isolate an application in its own environment so that it does not conflict with existing applications or modify the underlying operating system.

Limitations of a Standard Installation Environment

A typical Windows-based application has dependencies on components that are shared by multiple applications. Applications access these shared system resources, such as the registry or Windows system files. When an installation author recognizes that their application references a shared system component, they include a merge module to install that component.

When one of these shared components is installed, it is possible that a previously installed version of the same component could be overwritten; this may cause the existing application to break. A similar problem could occur when one of these applications containing a shared component is uninstalled. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment.

The following diagram provides an example of two conflicting installed applications.

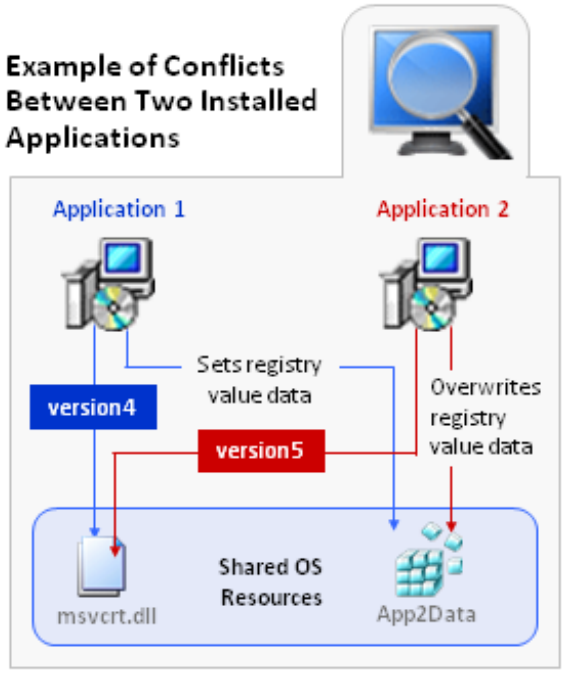


Figure 12-1: Example of Conflicts Between Two Installed Applications

Benefits of Application Virtualization

Virtual applications run in virtual environments that keep each application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

Even though virtual applications are not installed on the local machine, they exhibit the same functionality and access to local services as locally installed applications, and also nearly the same performance characteristics.

The following diagram provides an example of how application virtualization would solve the conflicts that are shown in the previous example.

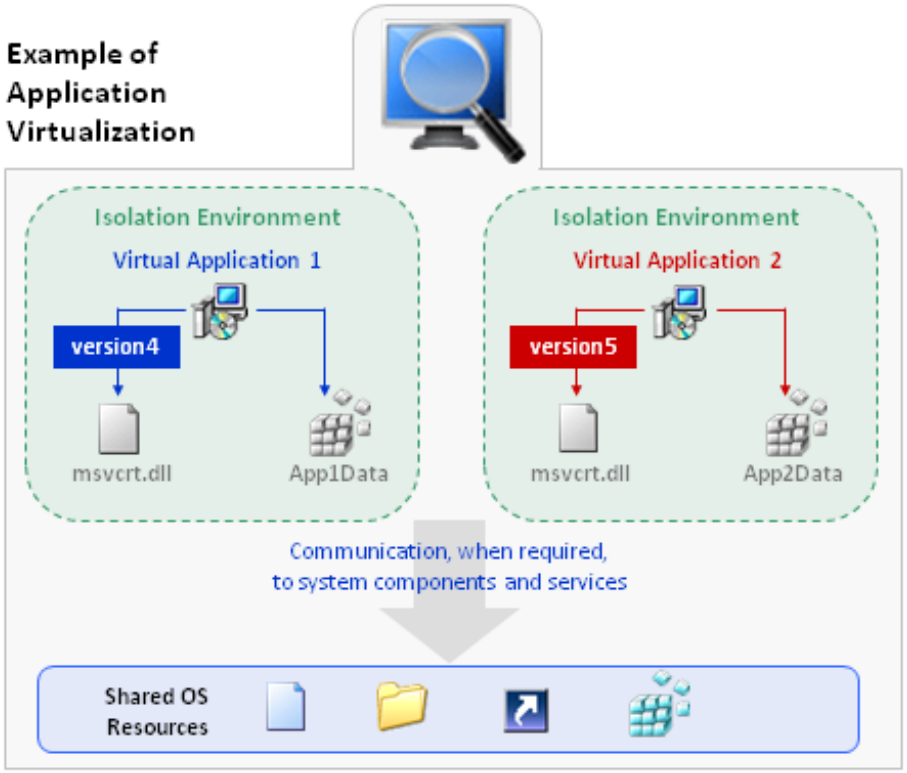


Figure 12-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files, and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies regression testing.

About the Virtual Package Editor

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally; only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

The Virtual Package Editor is a powerful tool that lets you edit App-V packages to customize your App-V applications, resolve virtualization best practice issues and application conflicts, and fix run-time problems. You can save your App-V packages as new packages that can be deployed alongside earlier versions of the virtual package in the same virtual environment; you can also create update packages that can upgrade earlier versions of your virtual applications.

Components of an App-V Package

The files that comprise an App-V package depend on the version of the App-V package.

Components of an App-V 5 Package (.appv)

The following table describes the main components of an App-V 5 package (.appv):

Table 12-2 • Components of an App-V 5 Package

File	Description
.appv	The .appv file is the compressed package file that contains all of the other parts of the package.
[Content_Types].xml	This file contains a list of file extensions that the package supports and the type of content to which each extension type maps.
AppxBlockMap.xml	This file contains a list of files with details such as header size and file size.
AppxManifest.xml	This file contains metadata about the package.
FilesystemMetadata.xml	This file contains information such as short file names, the directory-file hierarchy, and the mapping between the root folder and INSTALLDIR.
Registry.dat	This file contains registry data for the package.
StreamMap.xml	This file contains feature block 1 information.

Components of an App-V 4.x Package (.sft)

The following table describes the main components of an App-V 4.x package (.sft):

Table 12-3 • Components of an App-V 4.x Package

File	Description
.sft	The .sft file contains all of the files, registry information, and other configuration details of the package.
Manifest file	This file is an XML file that lists all of the .osd files in an App-V application.
.osd	The .osd files are XML-based files that describe the package's individual targets (or applications) that can be run.

Table 12-3 • Components of an App-V 4.x Package (cont.)

File	Description
.ico	The .ico files are icons files that are used for published shortcuts and file type associations.
.sprj	This file is the Microsoft App-V Sequencer project file. It contains references to the .sft and .osd files, and to a large number of settings related to the sequencing process.

Getting Started with the Virtual Package Editor

The Virtual Package Editor provides powerful features that make editing virtual packages easy. This section of the documentation contains information to help you become familiar with the Virtual Package Editor, begin editing a virtual package, and customize the Virtual Package Editor user interface.

Starting the Virtual Package Editor

Task ***To open the Virtual Package Editor, do one of the following:***

- On the **Tools** tab in AdminStudio, right-click **Virtual Package Editor** and then click **Launch Tool**.
- Launch **Application Manager**. On the **Products** tab, right-click an existing App-V package that you want to open, and then click **Edit with Virtual Package Editor**.

When you launch the Virtual Package Editor through the Tools tab, the Start Page opens. The Start Page provides access to product information, recently opened virtual packages, and product resources.

If you launch the Virtual Package Editor by opening a virtual package in the Application Manager, the Virtual Package Editor displays one of the views for the virtual package.

Opening an Existing Virtual Package

The Virtual Package Editor offers several ways to open an existing virtual package (.appv or .sft).

Task ***To open an existing virtual package:***

1. Do one of the following:
   - On the **File** menu, click **Open**.
   - Press CTRL+O.
   - On the toolbar, click the **Open** button.
   - On the **Start Page** in the **Package Tasks** area, click the **Open an Existing Package** link.

   The **Open** dialog box opens.
2. Browse to the virtual package (.appv or .sft), and then click the **Open** button.

The Virtual Package Editor opens the virtual package, enabling you to edit it as needed.

Tip *• As an alternative, you can open a recently opened virtual package. To do so, perform one of the following tasks:*

- *On the **Start Page** in the **Package Tasks** area, click the **Open an Existing Package** link.*
- *On the **File** menu, click a recently opened .appv or .sft file name.*

*You can also open a virtual package from within Application Manager: On the **Products** tab, right-click an existing App-V package that you want to open, and then click **Edit with Virtual Package Editor**. Application Manager lets you check out the file. When you save the virtual package in the Virtual Package Editor, your changes are saved in a temporary location. The version that is stored in Application Manager is updated when you check your changes in to the Application Manager.*

Saving a Virtual Package

The Virtual Package Editor offers several machine-wide, user-specific options for saving a virtual package (.appv or .sft). Before you save your virtual package as either a new package or an upgrade package, select the appropriate options.

Selecting the Appropriate Save Options

Task ***To select the appropriate save options:***

On the **File** menu, point to **Save Options**, and then click the appropriate command to select or clear an option. The following table describes each option. If an option is selected, its menu command includes a check mark.

Table 12-4 • Save Options

Command	Description
Include App-V Launcher	If you want to use the AdminStudio App-V Application Launcher to test a newly built App-V package locally before moving it to a deployment server, select this command. To learn more, see Using the App-V Application Launcher to Test the Virtual Package. This command is selected by default.
Append Package Version	If you want the Virtual Package Editor to append the package version number to the name of the file whenever you save an App-V package, select this command. This command is selected by default.

Table 12-4 • Save Options (cont.)

Command	Description
Build Wrapper MSI	If you want to build a Windows Installer package to assist in the distribution of each App-V package that you save in the Virtual Package Editor, select this command. If you enable this option, the Virtual Package Editor creates an InstallShield project (.ism file) and uses it to build an .msi package. If you run the .msi package, it "installs" the App-V application files in the local App-V client system cache. The wrapper .msi package can optionally include the App-V package, depending on whether the Include SFT in Wrapper MSI command is selected. ***Note*** *• The Microsoft Application Virtualization Client must be installed on the local machine before you can install an App-V application through a wrapper .msi package. The installation determines whether the App-V client is present; if it is not, the installation displays an error message and exits.* Building a wrapper .msi file simplifies the deployment of an App-V application by enabling you to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management. This command is cleared by default.
Include SFT in Wrapper MSI	If you want to include the App-V package in the Windows Installer package that you save in the Virtual Package Editor, select this command. If you enable this option, the Virtual Package Editor includes the .appv or .sft file in the wrapper .msi package that it builds when you save the open App-V package. If you run the .msi package, it "installs" the App-V application files, including the .appv or .sft file, in the local App-V client system cache. If you disable this option, the contents of the .appv or .sft file are streamed from the App-V server as requested by the client. This command is available only if the Build Wrapper MSI command is selected. This command is cleared by default.
Compress Wrapper MSI	If you want to compress the App-V package files into the wrapper .msi package, select this command. This command is available only if the Build Wrapper MSI command and the Include SFT in Wrapper MSI option are selected. If you enable the former command but disable the latter command, the .msi package that is generated is always compressed. This command is selected by default.

You can save a virtual package in either of the following ways:

- Save the package as a new package. You can deploy a new package alongside earlier versions of the virtual package in the same virtual environment.
- Save the package as an update package. An update package can upgrade earlier versions of the virtual application.

Tip • *If you are using the Virtual Package Editor to edit a virtual package that is part of your application catalog, the Virtual Package Editor saves your changes in a temporary location. To update your application catalog with the latest changes to your virtual package, use Application Manager to check in your virtual package.*

Saving a Virtual Package as a New Package

Task ***To save a virtual package as a new package, do one of the following:***

- On the **File** menu, click **Save**.
- Press CTRL+S.
- On the toolbar, click the **Save** button.

The Virtual Package Editor saves your virtual package as a new package. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Important • *If you want to deploy two copies of a package side by side, you must do some additional work before saving the package:*

- *In the General Information view, change the value of the Root Folder Name setting. This value must be unique because two packages with the same root folder name cannot be deployed simultaneously.*
- *In the General Information view, change the value of the Name setting. It is recommended that this value be different for each package.*
- *In the Shortcuts view, change the value of the Name setting, the Target Version setting, or both of those settings for each target in your package. The combination of the name and version must be unique for the targets in each new package; otherwise, the two packages cannot be deployed simultaneously.*

Saving a Virtual Package as an Update Package

Task ***To save a virtual package as an update package:***

1. On the **File** menu, click **Save As**. The **Save As** dialog box opens.
2. Click the **Save as a new package** option.

The Virtual Package Editor saves your virtual package as an update package. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Saving a Virtual Package (an Update Package or a New Package) with a New Name and Location

1. On the **File** menu, click **Save As**. The **Save As** dialog box opens.
2. In the **Virtual Package** box, enter the path and file name that you want to use for the .appv or .sft file. As an alternative, you can click the ellipsis button (**...**) to browse to the file.
3. Click the **Save as an update package** option or the **Save as a new package** option.

The Virtual Package Editor saves your virtual package as specified. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Closing a Virtual Package

Task ***To close a virtual package in the Virtual Package Editor:***

1. Select the tab of the file that you want to close.
2. Do one of the following:
   - On the **File** menu, click **Close**.
   - Click the tab's **Close** button.

Working with the Virtual Package Editor Interface

The Virtual Package Editor interface is a graphical user interface with conventional Windows-based elements such as a menu bar, a toolbar, and dialog boxes. This section includes topics that explain how to perform basic tasks using these elements and how to customize the interface.

Configuring the Value of a Setting for More Than One Item at a Time

In many views of the Virtual Package Editor, you can select multiple items—such as files, registry keys, file extensions, or virtual services—and then change the value for one of the settings. The Virtual Package Editor lets you use the same value in that setting for all of the selected items in that view. This feature may save you time by enabling you to make extensive changes to multiple items simultaneously, instead of requiring you to edit the setting for each item individually.

For example, in the Files and Folders view, you may want to change the value of the **Isolation** setting for a large number of folders. If all of the folders need to be configured the same way, you can simply select all of the pertinent folders, and then change the value of the Isolation setting as needed. Therefore, it is not necessary to separately select and configure each folder that you want to modify.

Note that the values of some settings may not be equivalent for each selected item. For example, your virtual package may contain one folder whose Isolation setting is **Override**, and another file whose Isolation setting is **Merge**. If you select both of those folders in the Files and Folders view, you will see the following unequal sign as the value of the Isolation setting, indicating that the selected items have different values:

≠

In this example, you can select both folders and select the appropriate value—**Override** or **Merge**—to have the Virtual Package Editor use the same value for both folders.

In some cases, the Virtual Package Editor does not allow you to change unequal values for more than one selected item. For example, if you select two files that are in the same folder, you cannot change the value of the **Name** setting for both of those files simultaneously, since each file in a folder must have a different file name.

Note • *If you select two or more items and you want to delete the entry in a setting that shows the unequal sign to indicate different values, you must first enter a value in the setting; then you can delete that value. For example, if you want to delete the value of the* ***Description*** *setting for two selected file extensions, and those file extensions have different values in the* ***Description*** *setting, you must first enter a value, so that they both match. Then you can delete that value for both file extensions at the same time.*

The Virtual Package Editor provides several methods for selecting multiple items in a view.

Task ***To select multiple items in a view so that you can configure some of their settings simultaneously, do one of the following:***

- To select multiple consecutive items that are near each other, drag your mouse pointer to create a box that surrounds each item that you want to select. When you do this, ensure that you start dragging your mouse pointer in empty space; otherwise, you may drag one or more item to a new location.
- To select multiple consecutive files or folders, select the first file or folder, press and hold SHIFT, and select the last file or folder.
- To select multiple nonconsecutive files or folders, select one file or folder, press and hold CTRL, and select each additional file or folder.

Showing or Hiding the Start Page in the Virtual Package Editor

The Virtual Package Editor Start Page is a tab that provides quick access to product information, to recently opened projects, and to Virtual Package Editor resources. You can show or hide this tab as necessary.

Task ***To show the Start Page:***

On the **File** menu, click **Start Page**.

Task ***To hide the Start Page, do one of the following:***

- On the **Start Page** tab, click the **Close** button.
- Click the **Start Page** tab. On the **File** menu, click **Close**.

Rearranging the Start Page and Virtual Package Tabs

Each virtual package that you have open in the Virtual Package Editor is displayed on a separate tab. The Start Page is also displayed on a separate tab. The Virtual Package Editor lets you change the order of these tabs.

Task ***To change the order of the open tabs:***

Drag the tab that you want to move to the new location in the rows of tabs.

Showing or Hiding the Settings and Output Windows

The Settings window in the Virtual Package Editor contains a grid that lists information about the item that is selected in an open view. The Output window displays task-specific information such as details about the virtual package that you are opening. It also shows save information.

The Settings window and the Output window can be shown or hidden as necessary.

Task ***To show or hide the Output window or the Settings window:***

On the **View** menu, click **Output Window** or click **Settings**.

If the window was visible, the Virtual Package Editor hides it. If the window was hidden, the Virtual Package Editor shows it.

Note that closing the Output window clears its contents. The Virtual Package Editor automatically shows the Output window whenever a task—such as saving or opening a virtual package—generates output.

Moving the Settings, Output, and Script Windows

The Settings window, the Output window, and the Script window can be moved to any side of the workspace in the Virtual Package Editor.

If you drag the Settings, Output, or Script window to the edge of a different side of the Virtual Package Editor interface, it becomes a docked window in that location.

Task ***To move the Settings window, Output window, the or the Script window:***

Drag the title bar of the **Settings** window, the **Output** window, or the **Script** window to the new location. Resize the window as needed.

Showing or Hiding Toolbars

Task ***To show or hide a toolbar, do one of the following:***

- Right-click a toolbar and select the toolbar that you want to be displayed or hidden.
- On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens. Select the check box for each toolbar that you want to be displayed. Clear the check box for each toolbar that you want to be hidden.

Adding Buttons and Menus to a Toolbar

Task ***To add a button or menu to a toolbar:***

1. Ensure that the toolbar that you want to change is visible.
2. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.

3. Click the **Commands** tab.
4. In the **Categories** box, click the category for the button or menu that you want to add.
5. Drag the button or menu from the **Commands** box to the appropriate toolbar.

Tip • *To create your own custom toolbar, drag the button or menu to the empty gray area near the toolbars.*

Removing Buttons and Menus from a Toolbar

Task ***To remove a button or menu from a toolbar:***

1. Ensure that the toolbar that you want to change is visible.
2. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.
3. Right-click the button or menu that you want to remove, and then click **Delete**.

Creating a Custom Toolbar

Task ***To create a custom toolbar:***

1. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.
2. Click the **Tools** tab.
3. Click the **New** button. The **New Toolbar** dialog box opens.
4. In the **Toolbar name** box, enter a descriptive name for the toolbar, and click **OK**.
5. Customize the new toolbar by adding menus or buttons.

Editing Virtual Packages

Editing a virtual package involves performing some or all of the following tasks.

Table 12-5 • Editing Virtual Packages

Task	Description
Specifying Virtual Package Information	Basic information that you enter in the General Information view is used by the Microsoft Application Virtualization Client and the App-V server. The Dependencies view is where you specify other App-V packages that your App-V package requires. ***Version*** • *The Dependencies view is available for App-V 4.x packages.*
Organizing Virtual Application Data	The Virtual Package Editor lets you manage the folders and files that will be available in the virtual environment. It also lets you define registry keys, values, and data for your virtual package. In addition, you can use the Virtual Package Editor to define the targets for your virtual application, and define entry points such as shortcuts for each target. These entry points enable end users to launch an App-V application from within the virtual environment.
Configuring Virtual Services	The Virtual Package Editor enables you to configure services that you want to include in your virtual package so that they are available in the virtual environment.
Testing and Troubleshooting Virtual Packages	Once you have made the necessary changes for the files, folders, shortcuts, services, and other elements of your virtual package, you are ready to test the virtual package, and identify potential conflicts and best practice violations between different App-V packages, and between App-V packages and Windows Installer–based installations. The Virtual Package Editor lets you add to a virtual package shortcuts that launch the Command Prompt window and the registry editor in order to debug issues in the virtual environment.

Specifying Virtual Package Information

When you open an existing package in the Virtual Package Editor, you may need to view or specify important package information. This includes basic information such as the name of the virtual package and details such as the package GUID and version number. You may also want to see history information such as each date on which the package was saved.

Viewing History for a Virtual Package

Version • *This information applies to App-V 4.x packages.*

The Virtual Package Editor shows read-only history information such as the following:

- The date and time when the package was saved

- The GUID of each saved package
- The user name of the person who saved the package
- The name of the machine on which the package was saved
- The version of the Virtual Package Editor that was used to save the package
- The version of App-V that was used when saving the package
- The operating system of the machine on which the package was saved

Each time that you save your file, the Virtual Package Editor adds a new history entry to the History pane and shows such details.

Task ***To view history for your virtual package:***

1. In the View List under **Package Information**, click **General Information**.
2. Review the information in the **History** pane.

Configuring General Information for a Virtual Package

In the General Information view, you can view and, if appropriate, edit basic information about your virtual package.

Task ***To configure general information for your virtual package:***

1. In the View List under **Package Information**, click **General Information**.
2. In the **Settings** window, configure the settings as needed. For details about each setting, see General Information View.

Specifying a Virtual Package's Dependencies

Version • *This information applies to App-V 4.x packages.*

A virtual package may rely on one or more other virtual packages in order to function properly. The Virtual Package Editor lets you specify other App-V packages that the open App-V package (the primary package) requires. This capability, called *Dynamic Suite Composition*, enables your virtual package to interact with the other virtual applications in the virtual environment. Dynamic Suite Composition enables you to deploy common system components once on each client system, making them available for use by many App-V applications, rather than requiring you to include them with each of the App-V applications that are dependent on them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary App-V application.

If you add a new dependency to your primary package, the Virtual Package Editor automatically associates each of the targets that are defined in the Shortcuts view with that new dependency. Similarly, if you add a new target to your primary package, the Virtual Package Editor automatically associates that target with each dependency that is defined in the

Dependencies view. Each .osd file that defines a target contains a list of the other .sft files on which it depends. The Application Virtualization Client may cache this list; therefore, in most cases all of the primary package's targets should be associated with each dependency.

Adding a Dependency to a Virtual Package

Version • *This information applies to App-V 4.x packages.*

The Virtual Package Editor lets you specify other App-V packages that the open App-V package (the primary package) requires.

Task ***To add a dependency to your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. Right-click the **Dependencies** explorer and then click **Add Dependency**. The **Open** dialog box opens.
3. Browse to the .sft or .osd file for the required App-V package, and then click **Open**.

The Virtual Package Editor adds an .sft item to the Dependencies explorer. The .sft item may contain one or more targets. The targets are defined in the Shortcuts view of the primary package.

When a target with an associated dependency is launched, the Application Virtualization Client loads the dependency's environment and makes it available as part of the virtual environment of the target's package.

Configuring a Dependency in a Virtual Package

Version • *This information applies to App-V 4.x packages.*

The Virtual Package Editor lets you view and configure settings for the dependencies in your virtual package. The settings display information such as the GUID and the server URL for the dependency.

Task ***To configure a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, click the dependency that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Dependencies View.

Associating a Package's Targets with a Dependency in a Virtual Package

Version • *This information applies to App-V 4.x packages.*

It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies. If you add a new dependency to your primary package, the Virtual Package Editor automatically associates each of the targets that are defined in the Shortcuts view with that new dependency. Similarly, if you add a new target to your primary package, the Virtual Package Editor automatically associates that target with each dependency that is defined in the Dependencies view. If you remove a target from a dependency in the Dependencies view, you may want to add it back.

Task ***To associate a target with a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the .sft file with which you want to associate a target, and then click **Associate Target**.

   If one or more targets in the package are not associated with the dependency, the **Associate Targets with a Dependency** dialog box opens. Select the targets that you want to associate with the dependency.

   If all of the targets in the package are associated with the dependency, the Virtual Package Editor displays a message box informing you that all of the package's targets are already associated with the dependency.

The Virtual Package Editor adds one or more targets to the dependency if appropriate.

Specifying Whether a Dependency is Mandatory for a Target in a Virtual Package

Version • *This information applies to App-V 4.x packages.*

The Virtual Package Editor lets you specify whether a dependency is mandatory in order for target in the primary package (the App-V package that you are editing in the Virtual Package Editor) to run properly. If the dependency is mandatory, the primary package cannot run without loading the required package. For example, a system DLL such as an MFC DLL is likely to be mandatory, but a reference to a document viewer such as Adobe Reader may not be mandatory.

Task ***To specify whether a dependency is mandatory for a target in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, click the target that you want to configure.
3. In the **Settings** window, configure the **Mandatory** setting as needed.

Tip • *The Virtual Package Editor lets you configure the settings for more than one target at a time. To learn more, see* *Configuring the Value of a Setting for More Than One Item at a Time**.*

Removing a Target from a Dependency in a Virtual Package

Version • *This information applies to App-V 4.x packages.*

Important • *It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.*

Task ***To remove a target from a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the target that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the target from the dependency.

Removing a Dependency from a Virtual Package

Version • *This information applies to App-V 4.x packages.*

The Virtual Package Editor lets you remove a dependency from an App-V package.

Task ***To remove a dependency from your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the .sft file dependency that you want to delete, and then click **Remove**.

The Virtual Package Editor removes the dependency from your App-V package.

Configuring Asset Intelligence Information

Asset intelligence is used to enhance the inventory capabilities of Microsoft System Center 2012 Configuration Manager by extending hardware inventory and adding license management functionality. The System Center 2012 Configuration Manager asset intelligence features can report application data such as digital PID, MSI product codes, and publisher names for each virtual application registered on a client computer.

In App-V 5 packages, asset intelligence information is incorporated into the package itself, with the information being captured during sequencing.

You can view and edit information identifying an App-V 5.0 application on the **Asset Intelligence** view. Typically these values are read in from the Add/Remove Programs Uninstall registry key.

In the **Asset Intelligence** view, you can view and, if appropriate, edit intelligence information about your App-V 5.0 virtual package.

Task ***To configure asset intelligence information for your App-V 5.0 virtual package:***

1. Open an App-V 5.0 package in Virtual Package Editor.
2. In the View List under **Package Information**, click **Asset Intelligence**.

3. In the **Settings** window, configure the settings as needed. For details about each setting, see Asset Intelligence View.

Organizing Virtual Application Data

The primary objective of a virtual package is to keep the application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

The Virtual Package Editor lets you manage the folders and files that will be available in the virtual environment. It also lets you define registry keys, values, and data for your virtual package. In addition, you can use the Virtual Package Editor to define the targets for your virtual application, and define entry points such as shortcuts for each target. These entry points enable end users to launch an App-V application from within the virtual environment.

Including Files and Folders

The Virtual Package Editor lets you manage the files and folders that are in your virtual package. This includes the files and folders in the root folder, the virtual file system (VFS) folder, and—if applicable—the SoftGridUserSettings folder. The Virtual Package Editor also lets you extract folders and files from the App-V package file (.appv or .sft) to a location that you specify.

Adding a Predefined Folder to the VFS Folder in an App-V Package

The Virtual Package Editor lets you add various folders that use a constant to the VFS folder. App-V 5 packages use system constants. App-V 4.x packages use CSIDL constants and SFT constants (such as CSIDL_APPDATA and SFT_PROGRAM_FILES_X64). At run time, the folder is mapped to the appropriate location in the virtual environment.

Task ***To add a predefined folder to the VFS folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the **VFS** folder, point to **Add Predefined Folder**, and then click the appropriate folder.

The Virtual Package Editor adds the predefined folder to the VFS folder.

Adding a Folder to an App-V Package

The Virtual Package Editor lets you add folders to your virtual package.

Task ***To add a folder to an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. Do one of the following:

- To add an existing folder and all of its contents to the package, in the **Files and Folders** explorer, right-click the location where you want to add a new folder and click **Add Folder**. The **Browse For Folder** dialog box opens, enabling you to select the folder that you want to add.
- To add a new empty folder to the package, in the **Files and Folders** explorer, right-click the location where you want to add a new folder and click **Add New Folder**. The Virtual Package Editor adds a new folder.

The Virtual Package Editor adds the folder to your virtual package.

Tip • *To change the name of the new folder, do one of the following:*

- *In the **Files and Folders** explorer, click the name of the new folder and then press F2. The Virtual Package Editor highlights the name of the folder, enabling you to edit it as needed.*
- *In the **Files and Folders** explorer, right-click the name of the new folder and then click **Rename**. The Virtual Package Editor highlights the name of the folder, enabling you to edit it as needed.*
- *Select the new folder, and in the **Settings** window, change the value of the **Name** setting.*

Adding a File to an App-V Package

The Virtual Package Editor lets you add files to your virtual package.

Task ***To add a file to an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the location where you want to add a new file and click **Add Files**. The **Select files to add to the virtual package** dialog box opens.
3. Select the file that you want to add and then click **Open**.

Tip • *To select multiple files in a folder, hold down the CTRL key while clicking files.*

The Virtual Package Editor adds the file or files that you selected to the virtual package.

Configuring a File or Folder in an App-V Package

The Virtual Package Editor lets you configure settings for the files and folders in your virtual package. The settings set information such as file attributes, whether a file is part of feature block 1, and the file or folder data type (application data or user data).

Task ***To configure the settings for a file or folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, click the file or folder that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Files and Folders View.

Tip • *The Virtual Package Editor lets you configure the settings for more than one file or folder at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Setting the VFS Path for the Contents of a Predefined Folder in an App-V Package

If a file or folder should exist outside the App-V package's root folder on the virtual file system, the Virtual Package Editor lets you modify the VFS path for that file or folder.

You may want to specify a VFS path if your virtual application tries to access the files in a folder by referring to a system folder to find the files (for example, looking up the Programs File folder) instead of using a relative path. You may also want to specify a VFS path if end users need to be able to find files when using a file browse dialog box (for example, for templates that are stored in a common file folder).

You can modify a folder's **VFS Path** setting by changing its **Isolation** setting. Using the **Isolation** setting, you can specify whether you want the selected folder in the App-V package to override the corresponding folder on the client system. The available options are:

- **Override**—The App-V application sees only the file content of the folder that is inside the App-V package. For an App-V 5.x package, this setting is inherited by all subfolders. For App-V 4.x packages, selecting this option automatically sets the read-only **VFS Path** setting.

  Note • *Overriding the isolation setting is also referred to as "fully virtualized".*

- **Merge**—The App-V application sees a merged view of the file content inside the App-V package and of the file content of the corresponding folder on the physical client system. For App-V 4.x packages, selecting this option automatically clears the **VFS Path** setting.

To modify the VFS path for the contents of a folder in an App-V package, perform the following steps:

Task ***To modify the VFS path for the contents of a folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, select the predefined folder that contains the files and folders whose VFS path you want to configure.
3. In the Settings window, set the folder's **Isolation** setting to **Override**. For App-V 4.x packages, the **VFS Path** setting will then be automatically set.

Moving a File or Folder in an App-V Package

The Virtual Package Editor lets you move files and folders in your virtual package from one location to another using drag and drop functionality.

Task ***To move a file or folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, drag a file or folder that you want to move to the appropriate location.

Extracting Files and Folders from the App-V Package

When you are editing an App-V package in the Virtual Package Editor, you may want to extract one or more files and folders from the package and save them to a local or network location. Doing so enables you to view the physical files that are streamed within the App-V package. If you extract a folder that contains subfolders and files, the Virtual Package Editor uses the same folder structure when saving the folder and its contents.

Task ***To extract a file from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the file that you want to extract, and then click **Extract**. The **Save As** dialog box opens.
3. Browse to the location where you want to save the file.
4. In the **File name** setting, enter a new name for the file if you want to use a different one.
5. Click the **Save** button.

The Virtual Package Editor saves the file in the location that you specified.

Task ***To extract a folder and its contents from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the folder that you want to extract, and then click **Extract**. The **Browse for Folder** dialog box opens.
3. Select the folder that you want to contain the extracted folder and its contents.

The Virtual Package Editor saves the folder, its subfolders, and its files in the location that you specified.

Removing a File or Folder in an App-V Package

The Virtual Package Editor lets you remove files and folders from your App-V package. If you remove a folder, all of its contents—including any subfolders and files—are also removed.

Task ***To remove a file or folder from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the file or folder that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the file or folder from your virtual package.

Editing the Virtual Registry

The Registry view enables you to define registry keys, values, and data for your App-V package. This view also lets you configure isolation options for selected registry keys. Isolation options indicate how the isolation environment provides access to system resources that the application needs: you can choose to override one or more keys on the client system, or you can choose to create a merged view of one or more keys for the virtual environment.

Note that the registry entries that are configured in the Registry view affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Keys, Value Names, and Value Data

The registry consists of machine data and user data. A key is a named location in the registry. A key can contain subkeys, a default value, and named values. A default value is a value without a name. All other values associate a name with some data: the value name identifies where to store it, and the value data is the data in that storage.

Note that the terms *key* and *subkey* are relative. In the registry, a key that is below another key can be referred to as a *subkey* or as a *key*, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding a Registry Key to a Virtual Package

The Virtual Package Editor enables you to add registry keys to your App-V package so that they are available in the virtual environment.

Task ***To add a registry key to your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. Do one of the following:
   - To add machine registry data, expand the **MACHINE** node.
   - To add user registry data, expand the **USER** node. If your App-V package does not contain any user registry data, you may need to add the **USER** node. To do so, right-click the **Registry** explorer, point to **Add Predefined Key**, and click **USER**.
3. In the **Registry** explorer, right-click the registry entry that you want to contain the new key, and then click **Add Key**.

The Virtual Package Editor adds a registry key with a default REG_SZ value.

Tip • *To change the name of the new registry key, do one of the following:*

- *In the **Registry** explorer, click the name of the new registry key and then press F2. The Virtual Package Editor highlights the name of the key, enabling you to edit it as needed.*
- *In the **Registry** explorer, right-click the name of the new registry key and then click **Rename**. The Virtual Package Editor highlights the name of the key, enabling you to edit it as needed.*
- *Select the new registry key, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Registry Key in a Virtual Package

If your virtual package includes one or more registry keys, you can configure each key's settings to specify information such as the value data, as well as whether the key in the App-V package should override the corresponding key on the client system.

Task ***To configure a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry key that you want to configure.
3. In the **Settings** window, configure the settings for the registry key as needed. For details about each setting, see Registry View.

Tip • *The Virtual Package Editor lets you configure the settings for more than one registry entry at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

*The **Isolation** setting lets you specify whether you want the selected registry key in the App-V package to either see only the registry content that is inside the App-V package for a key and all its subkeys, or see a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If you want to change the value of this setting for all of a registry key's subkey simultaneously, see Configuring the Isolation Setting for All of the Subkeys Under One or More Keys.*

Configuring the Isolation Setting for All of the Subkeys Under One or More Keys

If your virtual package includes a registry key that has multiple subkeys whose **Isolation** setting should be configured with the same value, you can quickly change the value of the **Isolation** setting for all of that key's subkeys simultaneously.

You can also quickly configure the **Isolation** setting for all of the subkeys that belong to multiple parent keys.

Task ***To configure the Isolation setting for all of the subkeys under one or more keys:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry key that you want to configure.

   If you want to configure the **Isolation** setting for all of the subkeys under multiple parent keys, select all of the applicable parent keys. To select multiple consecutive keys, select the first registry key, press and hold SHIFT, and select the last key. To select multiple nonconsecutive keys, select one key, press and hold CTRL, and select each additional key.
3. Right-click the selected key or keys and then click the appropriate command:
   - **Override Child Keys**—If you want to select **Override** for the **Isolation** setting of each subkey under the selected keys, select this option.

     The App-V application sees the registry content that is inside the App-V package for this key and all subkeys. Thus, the application does not see any registry content from the physical client system.

- **Merge Child Keys**—If you want to select **Merge** for the **Isolation** setting of each subkey under the selected keys, select this option.

  The App-V application sees a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If the registry key has subkeys on the physical client system but not in the App-V package, these keys are merged into the registry view that is available to the App-V application. However, registry values that are on the physical client system and that are in registry keys that also exist in the App-V package are not merged into the App-V application's registry view.

Adding a Registry Value to a Registry Key in a Virtual Package

The Virtual Package Editor enables you to add registry values to registry keys to your App-V package so that they are available in the virtual environment.

Task ***To add a registry value to a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry key that you want to contain the new value, and then click **Add Value**.

The Virtual Package Editor adds a registry value.

Tip • *To change the name of the new registry value, do one of the following:*

- *In the **Registry** explorer, click the name of the new registry value and then press F2. The Virtual Package Editor highlights the name of the value, enabling you to edit it as needed.*
- *In the **Registry** explorer, right-click the name of the new registry value and then click **Rename**. The Virtual Package Editor highlights the name of the value, enabling you to edit it as needed.*
- *Select the new registry value, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Registry Value and Its Value Data in a Virtual Package

If your virtual package includes one or more registry values, you can configure each value's settings to specify information such as the value data and the value type.

Task ***To configure a registry value and its value data in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry value that you want to configure.
3. In the **Settings** window, configure the settings for the registry key as needed. For details about each setting, see Registry View.

Tip • *The Virtual Package Editor lets you configure the settings for more than one registry entry at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Registry Value from a Registry Key in a Virtual Package

Task ***To remove a registry value a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry value that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the registry value from the registry key in your virtual package.

Removing a Registry Key from a Virtual Package

The Virtual Package Editor lets you remove registry keys from your App-V package. If you remove a registry key, all of its subkeys and values are also removed.

Task ***To remove a registry key from your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry key that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the registry key from your virtual package.

Defining Targets in a Virtual Application

The Virtual Package Editor lets you define each of the targets in your virtual package. Each target in your virtual package can contain one or more entry points, such as shortcuts, for each target. Entry points enable end users to launch each target in an App-V package from within the virtual environment.

Adding a Target to a Virtual Package

The Virtual Package Editor enables you to add one or more targets for your App-V package.

Task ***To add a target to your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. Right-click the **Targets** explorer and then click **Add Target**.

The Virtual Package Editor adds a new target.

Tip • *To change the name of the new target, do one of the following:*

- *In the **Targets** explorer, click the name of the new target and then press F2. The Virtual Package Editor highlights the name of the target, enabling you to edit it as needed.*

- *In the **Targets** explorer, right-click the name of the new target and then click **Rename**. The Virtual Package Editor highlights the name of the target, enabling you to edit it as needed.*
- *Select the new target, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Target in a Virtual Package

The Virtual Package Editor lets you configure settings for a target in your App-V package to specify information such as the file name and version number of the target file.

Task ***To configure a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the target that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Target Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one target at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Target from a Virtual Package

Task ***To remove a target from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the target that you want to remove, and then click **Remove**.

Creating Shortcuts to the Virtual Application on the Client System

Shortcuts offer quick access to a virtual application. You can configure your virtual package so that it adds shortcuts for your virtual application on the desktop, the Start menu, and various other locations on the client system.

Each shortcut that you create is part of a target in your virtual package. Each target in your virtual package can contain one or more entry points, such as shortcuts, for each target. At the target level, the Virtual Package Editor enables you to configure information such as the file in your virtual application that you want to launch, the icon that should be used for the target, and the command-line arguments that should be used to launch the file. For a shortcut, the Virtual Package Editor enables you to configure the display name and location of the shortcut.

Adding a Shortcut for a Virtual Package

The Virtual Package Editor enables you to add to your App-V package a shortcut that points to your App-V application.

You can add a shortcut to any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task ***To add a shortcut to a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new shortcut, right-click the **Shortcuts** folder, and then click **Add Shortcut**.

The Virtual Package Editor adds the shortcut to the Targets explorer.

Tip • *To change the display name of the new shortcut, do one of the following:*

- *In the **Targets** explorer, click the name of the new shortcut and then press F2. The Virtual Package Editor highlights the name of the shortcut, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new shortcut and then click **Rename**. The Virtual Package Editor highlights the name of the shortcut, enabling you to edit it as needed.*
- *Select the new shortcut, and in the **Settings** window, change the value of the **Display Name** setting.*

Configuring a Shortcut in a Virtual Package

The Virtual Package Editor lets you configure the display name of a shortcut. It also lets you configure the shortcut's location, such as on the desktop, the Start menu, or various other locations on the client system.

Tip • *To configure information such as the file in your virtual application that you want to launch, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the shortcut. To learn more, see Configuring a Target in a Virtual Package.*

Task ***To configure a shortcut in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the shortcut that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Shortcut Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one shortcut at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Shortcut from a Virtual Package

Task ***To remove a shortcut from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the shortcut that you want to remove, and then click **Remove**.

Using Environment Variables in a Virtual Environment

Environment variables are name and value pairs that can be accessed by your virtual application. Environment variables in a virtual package are stored in the virtual registry.

The Virtual Package Editor enables you to create environment variables that you want to be available to your virtual application in the virtual environment.

Note that the environment variables that are configured in an App-V package affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Setting an Environment Variable in a Virtual Package

Version • *The procedure for setting an environment variable varies, depending on the version of the App-V package.*

The Virtual Package Editor enables you to add to your App-V package one or more environment variables.

Setting an Environment Variable in an App-V 5 Package

Task ***To set an environment variable in your App-V 5 package:***

1. In the View List under **System Configuration**, click **Environment Variables**.
2. Right-click the **Environment Variables** explorer, and then click **Add Variable**.

The Virtual Package Editor adds the environment variable to the Environment Variables explorer.

Tip • *To change the name of the new environment variable, do one of the following:*

- *In the **Environment Variables** explorer, click the name of the new environment variable and then press F2. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *In the **Environment Variables** explorer, right-click the name of the new environment variable and then click **Rename**. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *Select the new environment variable, and in the **Settings** window, change the value of the **Name** setting.*

Setting an Environment Variable in an App-V 4.x Package

The Virtual Package Editor enables you to associate an environment variable with a target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task ***To set an environment variable in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to be associated with the new environment variable, right-click the **Environment Variable** folder, and then click **Add Variable**.

The Virtual Package Editor adds the environment variable to the Targets explorer.

Tip • *To change the name of the new environment variable, do one of the following:*

- *In the **Targets** explorer, click the name of the new environment variable and then press F2. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new environment variable and then click **Rename**. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *Select the new environment variable, and in the **Settings** window, change the value of the **Name** setting.*

Configuring an Environment Variable in a Virtual Package

Version • *The procedure for configuring an environment variable varies, depending on the version of the App-V package.*

The Virtual Package Editor lets you set the name and value of an environment variable in your App-V package.

Tip • *The Virtual Package Editor lets you configure the settings for more than one environment variable at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Configuring an Environment Variable in an App-V 5 Package

Task ***To configure an environment variable in your App-V 5 package:***

1. In the View List under **System Configuration**, click **Environment Variables**.
2. In the **Environment Variables** explorer, click the environment variable that you want to set.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Environment Variables View.

Configuring an Environment Variable in an App-V 4.x Package

Task ***To configure an environment variable in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the environment variable that you want to set.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Environment Variables View.

Removing an Environment Variable from a Virtual Package

Version • *The procedure for removing an environment variable varies, depending on the version of the App-V package.*

Removing an Environment Variable from an App-V 5 Package

Task ***To remove an environment variable from your App-V 5 package:***

1. In the View List under **System Configuration**, click **Environment Variables**.
2. In the **Environment Variables** explorer, right-click the environment variable that you want to remove, and then click **Remove**.

Removing an Environment Variable from an App-V 4.x Package

Task ***To remove an environment variable from your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the environment variable that you want to remove, and then click **Remove**.

Configuring File Extension Associations for the Virtual Application

The Virtual Package Editor enables you to set up file extensions in your virtual package. Once you have added a file extension, you can set up one or more verbs, such as Open or Print, for the file extension. When an end user double-clicks a file with that file extension in the virtual environment, the file opens in your virtual application. If an end user right-clicks a file with that file extension in the virtual environment, the context menu shown by Windows Explorer includes the display names of the verbs that are set up for the file extension.

Adding a File Extension to a Virtual Package

Version • *The procedure for adding a file extension varies, depending on the version of the App-V package.*

*Also note that for App-V 5 packages, the file extension that you enter must include a dot—for example, **.txt**. However, for App-V 4.x packages, the file extension that you enter should not include a dot—for example, **txt**.*

The Virtual Package Editor enables you to add a file extension to your App-V application.

Adding a File Extension to an App-V 5 Package

Task ***To add a file extension to your App-V 5 package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. Right-click the **File Extensions** explorer, and then click **Add File Extension**.

The Virtual Package Editor adds the file extension to the File Extensions explorer.

Tip • *To specify the file extension, do one of the following:*

- *In the **File Extensions** explorer, click the name of the new file extension and then press F2. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *In the **File Extensions** explorer, right-click the name of the new file extension and then click **Rename**. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *Select the new file extension, and in the **Settings** window, change the value of the **Extension** setting.*

*For App-V 5 packages, it is necessary to enter the dot—for example, enter **.txt** instead of **txt**.*

Adding a File Extension to a Target in an App-V 4.x Package

You can associate a file extension with any target in an App-V 4.x package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task ***To add a file extension to a target in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new file extension, right-click the **File Extensions** folder, and then click **Add File Extension**.

The Virtual Package Editor adds the file extension to the Targets explorer.

Tip • *To specify the file extension, do one of the following:*

- *In the **Targets** explorer, click the name of the new file extension and then press F2. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *In the **Targets** explorer, right-click the name of the new file extension and then click **Rename**. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *Select the new file extension, and in the **Settings** window, change the value of the **Extension** setting.*

*It is not necessary to enter the dot—for example, enter **txt** instead of **.txt**.*

Configuring a File Extension in a Virtual Package

Version • *The procedure for configuring a file extension varies, depending on the version of the App-V package.*

The Virtual Package Editor lets you configure information such as the MIME type and the ProgId of the file extension.

Configuring a File Extension in an App-V 5 Package

Task ***To configure a file extension in your App-V 5 package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. In the **File Extensions** explorer, click the file extension that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see File Extension Settings.

Configuring a File Extension in an App-V 4.x Package

Tip • *To configure information such as the file in your virtual application that you want to launch for the file extension, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the file extension. This is applicable to App-V 4.x packages. To learn more, see Configuring a Target in a Virtual Package.*

Task ***To configure a file extension in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the file extension that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see File Extension Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one file extension at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Adding a Verb to a File Extension in a Virtual Package

Version • *The procedure for adding a verb to a file extension varies, depending on the version of the App-V package.*

The Virtual Package Editor enables you to add a verb to a file extension to your App-V application.

Configuring a File Extension in an App-V 5 Package

Task ***To add a verb to a file extension in your App-V 5 package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. In the **File Extensions** explorer, right-click the file extension that you want to be associated with the new verb, and then click **Add Verb**.

The Virtual Package Editor adds the verb to the File Extensions explorer.

Tip • *To change the verb name, do one of the following:*

- *In the **File Extensions** explorer, click the name of the new verb and then press F2. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *In the **File Extensions** explorer, right-click the name of the new verb and then click **Rename**. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *Select the new verb, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a File Extension in an App-V 4.x Package

Task ***To add a verb to a file extension in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the appropriate target, right-click the file extension that you want to be associated with the new verb, and then click **Add Verb**.

The Virtual Package Editor adds the verb to the Targets explorer.

Tip • *To change the verb name, do one of the following:*

- *In the **Targets** explorer, click the name of the new verb and then press F2. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new verb and then click **Rename**. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *Select the new verb, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Verb for a File Extension in a Virtual Package

Version • *The procedure for configuring a verb for a file extension varies, depending on the version of the App-V package.*

The Virtual Package Editor lets you configure information such as the display name and dynamic data exchange (DDE) settings of a file extension's verb.

Configuring a File Extension in an App-V 5 Package

Task ***To configure a verb in your App-V 5 package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. In the **File Extensions** explorer, click the verb that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Verb Settings for a File Extension.

Configuring a File Extension in an App-V 4.x Package

Tip • *To configure information such as the file in your virtual application that you want to launch with the verb, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the verb. To learn more, see Configuring a Target in a Virtual Package.*

Task ***To configure a verb in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the verb that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Verb Settings for a File Extension.

Tip • *The Virtual Package Editor lets you configure the settings for more than one verb at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Verb from a File Extension in a Virtual Package

Version • *The procedure for removing a verb from a file extension varies, depending on the version of the App-V package.*

Removing a Verb from a File Extension in an App-V 5 Package

Task ***To remove a verb from a file extension in your App-V 5 package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. In the **File Extensions** explorer, right-click the verb that you want to remove, and then click **Remove**.

Removing a Verb from a File Extension in an App-V 4.x Package

Task ***To remove a verb from a file extension in your App-V 4.x package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the verb that you want to remove, and then click **Remove**.

Removing a File Extension from a Virtual Package

Version • *The procedure for removing a file extension varies, depending on the version of the App-V package.*

Removing a File Extension from an App-V 5 Package

Task ***To remove a file extension from your virtual package:***

1. In the View List under **Application Data**, click **File Extensions**.
2. In the **File Extensions** explorer, right-click the file extension that you want to remove, and then click **Remove**.

Removing a File Extension from an App-V 4.x Package

Task ***To remove a file extension from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the file extension that you want to remove, and then click **Remove**.

Creating Scripts that Run Before or After the App-V Application Is Streamed or Launched

The Virtual Package Editor lets you add to your App-V package scripts that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched. You can create scripts that make changes that your application requires, either in the App-V environment or on the client system outside the virtual environment. For example, you may want to launch a script that ensures that a particular file or registry entry exists, or that synchronizes data inside the virtual environment with data outside the virtual environment.

Two different types of scripting are available:

- **Single command (HREF)**—The App-V package references an external script or an executable file. The contents of the script are launched directly on the client system. The Command Prompt window is not displayed unless the process that is being called opens it.
- **Command script (SCRIPTBODY)**—The contents of the script are stored in the App-V package and copied to a temporary .bat file in the root folder (typically under the Q drive) of the App-V package on the client system. The .bat file is launched from a visible Command Prompt window.

You can use either type of scripting to call an executable file that exists in the folder on the virtual application server where the App-V package is stored.

Each script that you create is associated with a target in your virtual package. Each target in your virtual package can contain one or more entry points, such as scripts and shortcuts. At the target level, the Virtual Package Editor enables you to configure information such as the file in your virtual application that you want to launch and the command-line arguments that should be used to launch the file. For a script, the Virtual Package Editor enables you to configure information such as when you want the script to be launched.

Adding a Script to a Target in a Virtual Package

The Virtual Package Editor enables you to add to your virtual package a script that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched.

You can associate a script with any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task ***To add a script to a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new script, right-click the **Scripting** folder, and then click **Add Script**.

The Virtual Package Editor adds the script to the Targets explorer.

Configuring a Script in a Virtual Package

For App-V 4.x packages, you can use Virtual Package Editor to configure script information, such as when you want a script to be launched.

Tip • *To configure information such as the file in your virtual application that you want to be associated with the script, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the shortcut. To learn more, see Configuring a Target in a Virtual Package.*

Task ***To configure a script in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the script that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Scripting Settings.

   To learn how to trigger the appropriate behavior if the script fails, see Causing the App-V Application to Close After a Script Failure.
4. In the **Script** window, enter your script: either a one-line command or the body of the script.

Tip • *The Virtual Package Editor lets you configure the settings for more than one script at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Guidelines for Entering Script

Note the following guidelines when you are entering script in the Shortcuts view.

- If you are entering script for the command script (Scriptbody) type of script, you can use any script language that the client operating system supports.
- If you are entering script for the command script (Scriptbody) type of script, you can use command processor commands such as CHDIR and MOVE in your script.

  If you are entering script for the single command (HREF) type of script, command processor commands cannot be used, unless you launch `cmd.exe` to run the script.
- The Virtual Package Editor automatically adds the proper escape sequence for a newline character (\n), if appropriate, to the App-V package that it generates. Thus, to end a line and start a new one, simply press Enter; avoid entering a newline character (\n).

  The Virtual Package Editor also automatically adds the backslash character (\) if you enter a backslash, resulting in a double backslash (\\) in the App-V package. Thus, if you are specifying a path, do not use the escape character.
- If you are entering script for the single command (HREF) type of script, ensure that you enter only one line of script. If you enter more than one line, the Virtual Package Editor ignores all of the lines after the first line.

  Note that if you are using the command script (Scriptbody) type of script, you can enter more than one line of script.

Causing the App-V Application to Close After a Script Failure

If you have added a script that you want to be run for your App-V package, you can also specify the conditions under which the App-V package should be closed or the App-V package streaming should be stopped.

Note • *If you specify Post-shutdown for the Event setting of the script, any values that you specify for the Success Result setting and the Abort Result setting are ignored.*

Task ***To specify success and abort behavior for a script in your App-V package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the script that you want to configure.
3. Enter the appropriate value in the **Success Result** setting or the **Abort Result** setting, as appropriate.

Note • *In some versions of App-V, unexpected results could occur if you specify values for both the Success Result setting and the Abort Result setting.*

Table 12-6 • Client Behavior for Various App-V Package Settings and Script Return Codes

Script Return Code	Value for the Success Result Setting	Value for the Abort Result Setting	Behavior of the Application Virtualization Client
0	Any value or empty	0	The Application Virtualization Client silently aborts the application startup.
1	1	0 or null	The Application Virtualization Client proceeds with the next part of the App-V application streaming or launching.
1	Null	0	The Application Virtualization Client proceeds with the next part of the App-V application streaming or launching.
2	1	1 or null	The Application Virtualization Client fails to stream the package or start the application, and it displays an error message. The message includes an error code in the format of xxxxxx-xxxxxx18-0000000*n*, where *n* represents the return code of the script. The error code is written to the log file.

Removing a Script from a Virtual Package

Task ***To remove a debug tool from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the script that you want to remove, and then click **Remove**.

Specifying the Application Path for a File in a Virtual Package

Version • *This information applies to App-V 5 packages.*

An application path is a list of directories for the search path that the system should use to load DLLs and other files for the application. These directories are added to the beginning of the system PATH environment variable.

The Virtual Package Editor enables you to create application paths for your virtual application in the virtual environment.

Note that the application paths that are configured in an App-V package affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Adding an Application Path to a Virtual Package

Version • *This information applies to App-V 5 packages.*

The Virtual Package Editor enables you to add to your App-V package one or more application paths.

Task ***To set an environment variable in your virtual package:***

1. In the View List under **System Configuration**, click **Application Paths**.
2. Right-click the **Application Paths** explorer, and then click **Add App Path**.

The Virtual Package Editor adds the application path to the Application Paths explorer.

Tip • *To change the name of the new application path, do one of the following:*

- *In the **Application Paths** explorer, click the name of the new application path and then press F2. The Virtual Package Editor highlights the name of the application path, enabling you to edit it as needed.*
- *In the **Application Paths** explorer, right-click the name of the new application path and then click **Rename**. The Virtual Package Editor highlights the name of the application path, enabling you to edit it as needed.*
- *Select the new application path, and in the **Settings** window, change the value of the **Name** setting.*

Configuring an Application Path in a Virtual Package

Version • *This information applies to App-V 5 packages.*

The Virtual Package Editor lets you specify details for each application path in your App-V package.

Tip • *The Virtual Package Editor lets you configure the settings for more than one application path at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Task ***To configure an application path in your virtual package:***

1. In the View List under **System Configuration**, click **Application Paths**.
2. In the **Application Paths** explorer, click the environment variable that you want to set.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Application Paths View.

Removing an Application Path from a Virtual Package

Version • *This information applies to App-V 5 packages.*

Task ***To remove an application path from your virtual package:***

1. In the View List under **System Configuration**, click **Application Paths**.
2. In the **Application Paths** explorer, right-click the application path that you want to remove, and then click **Remove**.

Configuring Virtual Services

Windows services are executable files that Windows–based systems run in the background to manage various system tasks. A service is an executable file, but it must be designed as a service; you cannot automatically use an arbitrary executable file as a service. Windows services can be configured to run every time that the system starts or on demand when needed. The Virtual Package Editor enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.

Adding a Virtual Service to a Virtual Package

The Virtual Package Editor enables you to add virtual services to your App-V package so that they are available in the virtual environment.

Task ***To add a virtual service to your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. Right-click the **Virtual Services** explorer and then click **Add Virtual Service**.

The Virtual Package Editor adds a new service.

Tip • *To change the name of the new service, do one of the following:*

- *In the **Targets** explorer, click the name of the new service and then press F2. The Virtual Package Editor highlights the name of the service, enabling you to edit it as needed.*

- *In the **Targets** explorer, right-click the name of the new service and then click **Rename**. The Virtual Package Editor highlights the name of the service, enabling you to edit it as needed.*
- *Select the new service, and in the **Settings** window, change the value of the **Display Name** setting.*

Configuring a Virtual Service in a Virtual Package

If your virtual package includes one or more virtual services, you can configure each service's settings to specify information such as the service name, the path to the executable file, and the type of service.

Task ***To configure a virtual service in your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. In the **Virtual Services** explorer, click the service that you want to configure.
3. Configure the settings for the virtual service as needed. For details about each setting, see Virtual Services View.

Removing a Virtual Service from a Virtual Package

Task ***To remove a virtual service from your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. In the **Virtual Services** explorer, right-click the service that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the service from your virtual package.

Testing and Troubleshooting Virtual Packages

Once you have made the necessary changes for the files, folders, shortcuts, services, and other elements of your virtual package, you are ready to test the package. The Virtual Package Editor lets you optionally include the AdminStudio App-V Application Launcher with your App-V package if you want to test a newly saved App-V package locally before moving it to a deployment server.

Application Manager Analyze includes several Application Conflict Evaluators (ACEs) that may help you identify potential conflicts between different App-V packages, and between App-V packages and Windows Installer–based installations. Part of your testing strategy may involve using Analyze to detect potential conflicts.

If you encounter issues when running the App-V package, you can add to the package shortcuts that launch the Command Prompt window and the registry editor. These types of shortcuts may help you debug problems with an App-V package, since they enable you to examine the file system and view the registry while the virtual application is running in the virtual environment.

Using the App-V Application Launcher to Test the Virtual Package

You can use the AdminStudio App-V Application Launcher to test a newly saved App-V package on a test machine before moving it to a deployment server.

If you want the Virtual Package Editor to include the App-V Application Launcher whenever you save the App-V package, enable the App-V Launcher save option. To learn more, see Saving a Virtual Package.

Requirements for Using the App-V Application Launcher

The machine on which you use the App-V Application Launcher to test an App-V package must meet the following requirements:

- The Microsoft Application Virtualization Client must be installed.
- The version of the Microsoft Application Virtualization Client that is present should be equal to or newer than the minimum client version of the App-V package. The Virtual Package Editor displays the minimum client version of the App-V package in the General Information view.
- File streaming must be enabled because the App-V Application Launcher publishes the App-V package from a local file path. If file streaming is not enabled, the App-V Application Launcher displays an informative message asking if it can enable this functionality.

Starting the App-V Application Launcher

When you save an App-V package in the Virtual Package Editor and the App-V Launcher save option is enabled, the Virtual Package Editor adds the App-V Application Launcher (`AppVLauncher.exe`) to the same folder as the App-V package every time that you save an App-V package.

Task ***To use the App-V Application Launcher for testing a virtual package:***

1. In the Virtual Package Editor, open the App-V package that you want to test.
2. Do one of the following:
   - On the **View** menu, click **Show in Explorer**.
   - Press CTRL+E.
   - On the toolbar, click the **Explore** button.

   A Windows Explorer window opens. It shows the folder that contains the .appv or .sft file, the .xml files, the `AppVLauncher.exe` file, possibly one or more .osd files, possibly a Registry.dat file, and possibly an icon folder.

   If you have saved the App-V package as a new version one or more times, the folder may also contain a subfolder for each earlier version. The subfolders are named bkup_*N*, where *N* represents the version number of the App-V package.
3. Copy the contents of the folder (except for the bkup_*N* folders) to a test machine that meets the aforementioned App-V Application Launcher requirements. The `AppVLauncher.exe` file should be in the same folder as the .appv or .sft file.
4. Double-click the `AppVLauncher.exe` file.

If the App-V package has one target defined in the Shortcuts view (that is, if the App-V package has only one .osd file), the App-V Application Launcher starts the App-V application.

If the App-V package has more than one target defined in the Shortcuts view (that is, if the App-V package has two or more .osd files), the App-V Application Launcher displays a dialog box that lists each target, and it lets you select the one that you want to launch.

Note • *The first time that you use the App-V Application Launcher to run an application in an App-V package, the entire package is published to that machine; this includes all of the package's shortcuts and file extension associations in the package. If you then use the App-V Application Launcher to run any application in the App-V package again, the App-V Application Launcher unpublishes the package (and its shortcuts and file extension associations) before republishing the package.*

Also note that the AppVLauncher.exe file requires elevation. If you want to be able to test your App-V package in a locked-down environment where end users will not have elevated privileges, you may want to use the App-V Application Launcher once to launch and publish your App-V package with elevated privileges. Once you have done that, you can use the published shortcuts and file extension associations to start your application.

Using Debug Tools with a Virtual Package

The Virtual Package Editor lets you incorporate the following tools in a virtual package:

- **Cmd.exe (x86)**—The 32-bit version of Cmd.exe on the local machine runs, and it has access to the virtual environment.
- **Cmd.exe (x64)**—The 64-bit version of Cmd.exe on the local machine runs, and it has access to the virtual environment. This requires Microsoft Application Virtualization Client 4.6 or later, and it also require that the App-V package be published in a 64-bit environment.
- **Regedit.exe**—Regedit.exe on the local machine runs, and it has access to the virtual environment.

These debug tools may help you troubleshoot issues in the virtual package.

Important • *It is recommended that you use the debug tools only for testing. Before you release your virtual package, remove these tools from the package.*

Task ***To add a debug tool to your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. Right-click the **Targets** explorer, point to **Add Debug Tool**, and then point to the appropriate command:
   - Cmd.exe (x86)
   - Cmd.exe (x64)
   - Regedit.exe

The Virtual Package Editor adds the debug tool to the Targets explorer. The debug tool includes a shortcut that you can use to launch the tool in the virtual environment.

Adding a debug tool to a virtual package is similar to adding a target. Therefore, if appropriate, you can perform other tasks for the debug tool, just as you can for a target. For example, if you want to launch a script whenever you launch the Command Prompt window to simulate run-time behavior, you can add a script to the debug tool.

Task ***To remove a debug tool from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the debug tool that you want to remove, and then click **Remove**.

Using the Virtual Package Editor to Resolve Application Conflict Evaluators (ACEs) in App-V Packages

Version • *Some ACEs apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

You can use Application Manager to run Application Conflict Evaluators (ACEs) and identify potential conflicts between different App-V packages, and between App-V packages and Windows Installer packages. The following table lists ACE tests that pertain to App-V packages, as well as troubleshooting tips for resolving the issues through the Virtual Package Editor.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages

ACE Test	Test Group/Test Category	Description
ACE200	Inter-Application Conflicts/Microsoft App-V Conflict Tests	This ACE indicates that two or more packages contain a shortcut with the same display name and location. To resolve this issue in an App-V package, use the Shortcuts view to do one of the following: • Select the shortcut, and then modify the value in the Display Name setting or the Location setting. • Remove the shortcut from the App-V package.
ACE201	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE indicates that a target in the package has a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this issue in an App-V package, change the path of the target to use a variable instead of a hard-coded path: 1. In the View List under **Application Data**, click **Shortcuts**. 2. In the **Targets** explorer, select the target that contains the hard-coded path. 3. In the **Target** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Test	Test Group/Test Category	Description
ACE202	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE indicates that the command-line arguments for a target in the package include a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this issue in an App-V package, change the path to use a variable instead of a hard-coded path: **1.** In the View List under **Application Data**, click **Shortcuts**. **2.** In the **Targets** explorer, select the target that contains the hard-coded path. **3.** In the **Arguments** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.
ACE203	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE indicates that the working directory for a target in the package include a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this issue in an App-V package, change the path to use a variable instead of a hard-coded path: **1.** In the View List under **Application Data**, click **Shortcuts**. **2.** In the **Targets** explorer, select the target that contains the hard-coded path. **3.** In the **Working Directory** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.
ACE204	Inter-Application Conflicts/Microsoft App-V Conflict Tests	This ACE indicates that two or more packages have the same package GUID; therefore, the two packages cannot be deployed simultaneously as separate packages. If you are editing an update package that can upgrade earlier versions of the virtual package, the package GUID should stay the same. If you are editing a new package that can be deployed simultaneously as another package, the package GUID in one of the packages must be changed. To change the package GUID, save the package as a new package. To learn more, see Saving a Virtual Package.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Test	Test Group/Test Category	Description
ACE205	Inter-Application Conflicts/Microsoft App-V Conflict Tests	This ACE indicates that two or more packages have the same name. This is not advisable from a best practice perspective, and it may cause some issues if you try to simultaneously deploy the App-V packages. To resolve this issue: **1.** In the View List under **Package Information**, click **General Information**. **2.** In the **Name** setting, replace the duplicate name with a unique name.
ACE206	Inter-Application Conflicts/Microsoft App-V Conflict Tests	This ACE indicates that two or more packages have support for the same file extension. However, a file extension can be registered with only one application at a time. To resolve this issue, you may need to decide which package should contain the file extension association and which should not. Then you can use the Virtual Package Editor to remove the appropriate file extension.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Test	Test Group/Test Category	Description
ACE207	Inter-Application Conflicts/Microsoft App-V Conflict Tests	***Note*** *• This ACE applies to App-V 4.x packages.* This ACE indicates that two or more packages have the same long or short name for the root folder. These names must be unique because two packages with the same root folder name cannot be deployed simultaneously. To resolve this issue: 1. In the View List under **Package Information**, click **General Information**. 2. In the **Root Folder Name** setting, replace the duplicate folder name with a unique folder name. Note that instances of the old package's root folder name may still exist in location-related configuration data, such as in registry entries, .ini files, or XML files in the App-V package. The root folder name is not updated in those areas automatically if you change the root folder name in the General Information view. Therefore, if you know that the old package contains configuration data, you may need to identify where it is. Then you can use the Virtual Package Editor to update the root folder name as necessary. For example, you may want to use the Virtual Package Editor to extract a configuration file from the package. Next, you can update the root folder name in the file. In the Virtual Package Editor, you would then delete the old file from the App-V package, and add the updated file.
ACE208	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE indicates that an App-V package does not contain any shortcuts. You can ignore this ACE if one of the following is true: • This package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package and select this App-V package as a dependency in the Dependencies view. • This package is intended to be used as a plug-in. In this case, you need to create a shortcut to the application for which this is a plug-in. Some common examples include Office and Internet Explorer. If end users need to be able to launch this App-V package independently, consider adding a target to the App-V package if necessary, and then adding a shortcut to the target.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Test	Test Group/Test Category	Description
ACE215	Inter-Application Conflicts/Microsoft App-V Conflict Tests	This ACE indicates that the App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package. The combination of the name and version should be unique for shortcuts in different packages, since only one application is published and available at any given time. To resolve this issue in an App-V package, use the Shortcuts view to do one of the following: • Select the target that contains the shortcut, and then modify the value in the Name setting or the Target Version setting. • Remove the shortcut from the App-V package.
ACE216	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	 ***Note*** • *This ACE applies to App-V 4.x packages.* This ACE checks whether an App-V package's file name contains more than 56 characters. To resolve this issue in an App-V package, rename the .sft file with a name that contains fewer than 56 characters.
ACE217	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE checks whether the App-V package contains a WMI Provider component. If the WMI Provider is not an important part of the application, or if it can be separately installed from the App-V package, this issue can be suppressed and ignored.
ACE218	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE checks whether the App-V package contains a J2EE application server. If the J2EE application is not an important part of the application, or if it can be separately installed from the App-V package, this issue can be suppressed and ignored.
ACE219	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE checks whether the App-V package contains an ASP.NET or IIS application component. If the ASP.NET/IIS application is not an important part of the application, or if it can be separately installed from the App-V package, this issue can be suppressed and ignored.

Table 12-7 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Test	Test Group/Test Category	Description
ACE220	Virtualization and Windows Installer Best Practices/Microsoft App-V Best Practices	This ACE checks whether an App-V package contains files that indicate that the package includes unsupported applications such as antivirus software or server software such as Exchange Server or SQL Server. If these unsupported application components are not an important part of the application, or if they can be separately installed from the App-V package, this issue can be suppressed and ignored.

Virtual Package Editor Reference

Reference information for the Virtual Package Editor is organized into the following sections:

Table 12-8 • Reference Sections

Section	Description
Virtual Package Editor Start Page	Provides information about the Start Page in the Virtual Package Editor.
Virtual Package Editor Menu, Toolbar, and Window Reference	Describes various components of the Virtual Package Editor user interface, including menus, toolbars, and windows.
Virtual Package Editor Dialog Box Reference	Contains reference information on each of the dialog boxes that are displayed in the Virtual Package Editor.
Virtual Package Editor View Reference	Describes each of the views that are displayed in the Virtual Package Editor.

Virtual Package Editor Start Page

The Virtual Package Editor Start Page is a tab that provides quick access to product information, to recently opened projects, and to Virtual Package Editor resources. The Start Page includes the following sections:

Table 12-9 • Sections on the Start Page

Section	Description
Package Tasks	Click a package task to quickly open an existing virtual package.
Help Topics	Frequently accessed help topics are listed in this section. To access the entire Virtual Package Editor Help Library from anywhere within the Virtual Package Editor, you can press F1, click the Help button, or click one of the appropriate commands on the Help menu.

Table 12-9 • Sections on the Start Page (cont.)

Section	Description
Recent Packages	The section in the middle of the Start Page lists your most recently accessed virtual packages, their locations, and the dates on which they were last modified.
Resources	The Resources section contains links to connect you to helpful product information.

Virtual Package Editor Menu, Toolbar, and Window Reference

This section describes the various components of the Virtual Package Editor user interface, including menus, toolbars, and windows.

Menus in the Virtual Package Editor

The menus in the Virtual Package Editor are located on the menu bar, which is at the top of the Virtual Package Editor interface. Each menu contains a list of commands. Some of these commands have icons next to them so that you can quickly associate the command with the icon.

Each of the menus in the Virtual Package Editor is described in this section:

- File
- Edit
- View
- Window
- Help

File Menu in the Virtual Package Editor

The following table lists the File menu commands, as well as associated keyboard shortcuts and icons.

Table 12-10 • File Menu Commands

Command	Shortcut	Icon	Description
Open	CTRL+O		Opens an existing virtual package.
Close			Closes the currently selected tab.
Save	CTRL+S		Saves the currently selected virtual package as a new package. To learn about the various save options, see Saving a Virtual Package.

Table 12-10 • File Menu Commands (cont.)

Command	Shortcut	Icon	Description
Save As			Enables you to save the currently selected virtual package with a new name and location. Also lets you specify whether you want to save the virtual package as an upgrade package or as a new package. To learn about the various save options, see Saving a Virtual Package.
Start Page			Opens or closes the Start Page.
Save Options			Provides several commands: • Include App-V Launcher • Append Package Version • Build Wrapper MSI • Include SFT in Wrapper MSI • Compress Wrapper MSI To learn about the various save options, see Saving a Virtual Package.
1, 2, 3, 4, 5, 6			Opens one of the recently accessed virtual packages.
Exit			Closes the open virtual packages and closes the Virtual Package Editor.

Edit Menu in the Virtual Package Editor

The following table lists the Edit menu commands, as well as associated keyboard shortcuts and icons.

Table 12-11 • Edit Menu Commands

Command	Shortcut	Icon	Description
Undo	CTRL+Z		Undoes the last action performed.
Redo	CTRL+Y		Reverses the last action that was performed with the Undo command.
Cut	CTRL+X		Removes the currently selected text and places it on the Clipboard.
Copy	CTRL+C		Copies the currently selected text to the Clipboard.

Table 12-11 • Edit Menu Commands (cont.)

Command	Shortcut	Icon	Description
Paste	CTRL+V		Inserts the contents of the Clipboard at the insertion point, and replaces any selected text.
Remove	Delete		Deletes the currently selected text.
Refresh	F5		Refreshes the currently selected view.

View Menu in the Virtual Package Editor

The following table lists the View menu commands, as well as associated keyboard shortcuts and icons.

Table 12-12 • View Menu Commands

Command	Shortcut	Icon	Description
Toolbars			Provides the following commands: • **Standard**—Shows or hides the Standard toolbar. • **Customize**—Opens the Customize dialog box, which lets you add or remove toolbar buttons or implement other toolbar customization. If you created custom toolbars, they are also listed.
Settings			Opens or closes the Settings window.
Output Window			Opens or closes the Output window.
Status Bar			Opens or closes the status bar.
Show in Explorer	CTRL+E		Opens the folder that contains the currently selected virtual package in a Windows Explorer window. If an .appv or .sft tab is not currently selected, this command is disabled.

Window Menu in the Virtual Package Editor

The following table lists the Window menu commands.

Table 12-13 • Window Menu Commands

Command	Description
1, 2, 3	Opens the tab for the corresponding virtual package.

Help Menu in the Virtual Package Editor

The following table lists the Help menu commands.

Table 12-14 • Help Menu Commands

Command	Description
Contents	Displays the Contents tab of the help library.
Index	Displays the Index tab of the help library.
Search	Displays the Search tab of the help library.
About Virtual Package Editor	Displays the Virtual Package Editor dialog box, where you can find version information.

Standard Toolbar in the Virtual Package Editor

The Virtual Package Editor interface offers one built-in toolbar that gives you quick access to frequently used menu commands: the Standard toolbar.

The following table lists all of the buttons on the Standard toolbar.

Table 12-15 • Standard Toolbar Buttons

Button	Name	Shortcut	Description
	Open	CTRL+O	Opens an existing virtual package.
	Save	CTRL+S	Saves the currently selected virtual package. To learn about the various save options, see Saving a Virtual Package.
	Cut	CTRL+X	Removes the currently selected text and places it on the Clipboard.
	Copy	CTRL+C	Copies the currently selected text to the Clipboard.
	Paste	CTRL+V	Inserts the contents of the Clipboard at the insertion point, and replaces any selected text.
	Explore	CTRL+E	Opens the folder that contains the currently selected virtual package in a Windows Explorer window. If an .appv or .sft tab is not currently selected, this command is disabled.

Table 12-15 • Standard Toolbar Buttons (cont.)

Button	Name	Shortcut	Description
	Help	F1	Opens the help library.

Script Window

The Virtual Package Editor displays a Script window when you select a script in the Shortcuts view. Use the Script window to enter the script that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched.

To learn how to configure a script's settings and enter your script in the Script window, see Configuring a Script in a Virtual Package.

Settings Window

The Settings window in the Virtual Package Editor contains a grid that lists information about the item that is selected in an open view.

The following table describes the buttons that are displayed above the settings in the Settings window.

Table 12-16 • Controls in the Settings Window

Name of Control	Icon	Description
Categorized		Sorts the settings according to categories.
Alphabetical		Sorts the settings alphabetically.

Output Window

The Output window displays task-specific information such as details about the virtual package that you are opening. It also shows save information.

Note that closing the Output window clears its contents. The Virtual Package Editor automatically shows the Output window whenever a task—such as saving or opening a virtual package—generates output.

Virtual Package Editor Dialog Box Reference

This section of the documentation describes dialog boxes that are displayed in the Virtual Package Editor.

- Browse for Folder
- Edit Value

- Save As
- Select a File
- Select a Folder
- Select Files to Add to the Virtual Package

Browse for Folder Dialog Box

The Virtual Package Editor displays the Browse for Folder dialog box when you click the Browse Your Computer button in settings such as the Working Directory setting for the target that is selected in the Shortcuts view. The Browse for Folder dialog box lets you select an existing folder on your machine or create a new folder.

The Browse for Folder dialog box is also displayed when you right-click an item in the Files and Folders view and then click Add Folder. In this case, the Browse for Folder dialog box lets you select a local or network folder that you want to add to your virtual package.

Edit Value Dialog Box

The Virtual Package Editor displays the Edit Value dialog box in the following scenarios:

- You click the ellipsis button (...) in the Value Data setting for a registry value in the Registry view. The ellipsis button is displayed in this setting for a value type of REG_BINARY or REG_MULTI_SZ. Use the Edit Value dialog box to specify the value data.
- You click the ellipsis button (...) in the Group Dependencies setting or the Service Dependencies setting for a service in the Virtual Services view. Use the Edit Value dialog box to specify the groups or services that are required by the virtual service in your virtual package.

Table 12-17 • Edit Value Dialog Box Settings

Setting	Description
Value Name	This read-only setting shows the name of the value that you are editing.
Value Data	Enter the value data (the registry value data, the group dependencies, or the service dependencies). Specify each value on a separate line.

Save As Dialog Box

The Save As dialog box lets you specify the name and location where you want to save your virtual package. This dialog box lets you specify whether you want to save the current virtual package as an update package or as a new package.

Task ***To access the Save As dialog box:***

On the **File** menu, click **Save As**.

Table 12-18 • Save As Dialog Box Settings

Setting	Description
Virtual Package	Enter the path and file name that you want to use for the package file. As an alternative, you can click the ellipsis button (...) to browse to the file.
Save an update package	To save the virtual package as an update package that can upgrade earlier versions of the virtual application, click this option.
Save as a new package	To save the virtual package as a new package that you can deploy alongside earlier versions of the virtual package in the same virtual environment, click this option.

Select a File Dialog Box

The Virtual Package Editor displays the Select a File dialog box when you click the Browse This Package button in settings such as the Target setting and the Icon setting for a target in the Shortcuts view. The Select a File dialog box displays the directory tree for your virtual package, enabling you to select the appropriate file.

Select a Folder Dialog Box

The Virtual Package Editor displays the Select a Folder dialog box when you click the Browse This Package button in settings such as the Working Directory setting for a target in the Shortcuts view. The Select a Folder dialog box displays the directory tree for your virtual package, enabling you to select the appropriate folder that you want to use as the working directory for the selected target.

Select Files to Add to the Virtual Package Dialog Box

The Virtual Package Editor displays the Select Files to Add to the Virtual Package dialog box when you right-click a folder in the Files and Folders view and then click Add Files. The dialog box lets you select local or network files that you want to add to your virtual package.

Virtual Package Editor View Reference

The View List in the left pane is a navigational element that consists of folders and subnodes that you can click to open various areas within the Virtual Package Editor. Each folder and subnode in the View List represents a view within the Virtual Package Editor. The Virtual Package Editor View Reference section describes each of the subnode views.

- Application Paths View
- Asset Intelligence View
- Dependencies View

- Environment Variables View
- File Extensions View
- Files and Folders View
- General Information View
- Registry View
- Shortcuts View
- Virtual Services View

Application Paths View

Version • *The Application Paths view is available for App-V 5 packages.*

When you select an application path variable in the Application Paths view, the following settings are available.

Table 12-19 • Settings in the Application Paths View

Setting	Description
Name	Enter the name of the executable file for which you want to create an application path.
Executable Path	Enter the path and file name of the executable file for which you want to create an application path.
Path Prefix	Enter a semicolon-delimited list of fully qualified directories for the search path that the system should use to load DLLs and other files for the application. These directories are added to the beginning of the system PATH environment variable.

Asset Intelligence View

Asset intelligence is used to enhance the inventory capabilities of Microsoft System Center 2012 Configuration Manager by extending hardware inventory and adding license management functionality. The System Center 2012 Configuration Manager asset intelligence features can report application data such as digital PID, MSI product codes, and publisher names for each virtual application registered on a client computer.

In App-V 5 packages, asset intelligence information is incorporated into the package itself, with the information being captured during sequencing.

You can view and modify Asset Intelligence properties of an App-V 5.0 package by selecting **Asset Intelligence** under **Package Information** in the Virtual Package Editor tree, and then selecting an Asset Intelligence entry in the pane on the right.

The following settings are available on this view.

Table 12-20 • App-V Package Settings in the Asset Intelligence View

Setting	App-V Version	Description
Software Code	App-V 5.0	Name of the Add/Remove Programs Uninstall registry key for this package. For Windows Installer packages that were converted to App-V packages, this is the `ProductCode`.
Product Name	App-V 5.0	`DisplayName` value under the Add/Remove Programs Uninstall registry key for this package. This property contains the name of the application.
Product Version	App-V 5.0	`DisplayVersion` value under the Add/Remove Programs Uninstall registry key for this package. This property contains the version of the package in string format.
Publisher	App-V 5.0	`Publisher` value under the Add/Remove Programs Uninstall registry key for this package. This property contains the name of the manufacturer of the product.
Product ID	App-V 5.0	`ProductID` value under the Add/Remove Programs Uninstall registry key for this package. Often, this is a serial number or product SKU.
Language	App-V 5.0	Language value under the Add/Remove Programs Uninstall registry key for this package. This property specifies the language the installer should use for any strings in the user interface that are not authored into the database. This property must be a numeric language identifier (LANGID).
Channel Code	App-V 5.0	`ChannelCode` value under the Add/Remove Programs Uninstall registry key for this package.
Install Date	App-V 5.0	`InstallDate` value under the Add/Remove Programs Uninstall registry key for this package. It is the last time this product received service.
Registered User	App-V 5.0	`RegisteredUser` value under the Add/Remove Programs Uninstall registry key for this package.
Installed Location	App-V 5.0	`InstalledLocation` value under the Add/Remove Programs Uninstall registry key. for this package
CM DSLID	App-V 5.0	Definite Software ID (DSLID) for this package, if one exists.
Version Major	App-V 5.0	`VersionMajor` value under the Add/Remove Programs Uninstall registry key for this package. It is a numeric representation of a part of the product version.

Table 12-20 • App-V Package Settings in the Asset Intelligence View (cont.)

Setting	App-V Version	Description
Version Minor	App-V 5.0	`VersionMinor` value under the Add/Remove Programs Uninstall registry key for this package. It is a numeric representation of a part of the product version.
Service Pack	App-V 5.0	This may correspond to the service pack level of the product.
Upgrade Code	App-V 5.0	This may be the `UpgradeCode` of the Windows Installer package used to originally install this application.
OS Component	App-V 5.0	Numeric value that indicates whether the application is part of the operating system: • **0**—Application is not part of the operating system. • **1**—Application is part of the operating system.

Dependencies View

Version • *The Dependencies view is available for App-V 4.x packages.*

The Dependencies view is where you specify other App-V packages that the open App-V package requires.

Icons in the Dependencies View

The Dependencies explorer in the Dependencies view uses different icons to help you distinguish between different types of items. Following is a list of the possible icons in the Dependencies view.

Table 12-21 • Icons in the Dependencies View

Icon	Description
	This icon identifies the root node—the Dependencies explorer.
	This icon identifies an App-V package (.sft) that the open App-V package (the primary App-V package) requires.
	This icon identifies a target in the primary App-V package. It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.

Settings in the Dependencies View

The settings that are displayed in the Dependencies view differ, depending on whether you select an App-V package (.sft file) or a target in this view:

- .sft Settings
- Target Settings

.sft Settings

When you select an .sft file in the Dependencies view, the following settings are available.

Table 12-22 • .sft Settings in the Dependencies View

Setting	Description
Name	This read-only setting shows the name of the required .sft file.
GUID	This read-only setting shows the globally unique identifier (GUID) that is associated with the required App-V package.
SysGuard File	This read-only setting shows the folder and name of the required App-V package's SysGuard file (`osguard.cp`). The SysGuard file describes how the virtual environment needs to be set up.
HREF	Enter the URL for the published location of the required App-V package on the virtual application server. Typically, this location matches the App-V server URL for the App-V package that contains the dependency.

Target Settings

When you select a target in the Dependencies view, the following settings are available.

Table 12-23 • Target Settings in the Dependencies View

Setting	Description
Name	This read-only setting shows the name of the target that is associated with the required App-V package.
Mandatory	Specify whether the required App-V package is mandatory in order for the primary package (the App-V package that you are editing in the Virtual Package Editor) to run properly. Note that if the dependency is mandatory, the primary package cannot run without loading the required package.

Environment Variables View

Version • *The Environment Variables view is available for App-V 5 packages. The environment variable settings are also available when you select an environment variable in the Shortcuts view for App-V 4.x packages.*

When you select an environment variable, the following settings are available.

Table 12-24 • Environment Variable Settings

Setting	Description
Name	Enter the name of the environment variable that you want to configure for the virtual application.
Value	Enter the path or value for this environment variable. To enter multiple paths, separate the paths with a semicolon (;).

File Extensions View

Version *• The File Extensions view is available for App-V 5 packages. The file extension settings are also available when you select a file extension in the Shortcuts view for App-V 4.x packages.*

The File Extensions view (for an App-V 5 package) and the Shortcuts view (for an App-V 4.x package) enable you to associate a file extension with an application file.

Icons for File Extensions

The File Extensions explorer in the File Extensions view (for an App-V 5 package) and the Shortcuts view (for an App-V 4.x package) uses different icons to help you distinguish between different types of items. Following is a list of the possible icons.

Table 12-25 • Icons for File Extensions

Icon	Description
	This icon identifies a file extension.
	This icon identifies a verb for a file extension.

File Extension Settings

When you select a file extension in the File Extensions view (for an App-V 5 package) or the Shortcuts view (for an App-V 4.x package), the following settings are available.

Table 12-26 • File Extension Settings

Setting	Description
Extension	To associate a file extension with the App-V application, enter the extension. ***Version*** • *For App-V 5 packages, the file extension that you enter must include a dot—for example,* ***.txt****. However, for App-V 4.x packages, the file extension that you enter should not include a dot—for example,* ***txt****.*
Description	Enter the description text that you want to display for this file extension in the Application Virtualization Client.
MIME	Enter the MIME type that is associated with the file extension.
ProgId	Enter the program identifier—ProgId, also known as *application identifier* or *tag name*—that you want to associate with the file extension. A file type's ProgId is an arbitrary string, but it should be unique on the client system. One ProgId naming convention is to append the word **`file`** to your extension without a dot—the .ext extension might use the ProgId *extfile*. Another convention is to name a file-type ProgId after the application that is used to open the file type, as in **`SampleApp.Document`**. For example, an **.xyz** file extension could point to an **xyzfile** ProgId, and all of the **.xyz** file-type information would be registered under **xyzfile**.
Icon	Enter the path to the icon file (.ico, .exe, or .dll) that contains the icon resource for the file extension. The location can be in the App-V package, or on your computer or the network. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the icon. If the icon file that you specify contains more than one icon resource, after the icon path, add a comma and then the index number. For example, 0 refers to the first icon in the file, 1 refers to the second icon, and 2 refers to the third icon. To specify the third icon in the icon file, enter the following after the icon file path: **, 2**
Perceived Type	Select the appropriate type of file.

Table 12-26 • File Extension Settings (cont.)

Setting	Description
File Type Attributes	To set various file type attributes for the file association, select the appropriate option for each of the following settings: • **Open Is Safe**—Specify whether the open verb can be used safely for downloaded files that have this file extension. • **Always Unsafe**—Specify whether files that have this file extension should be considered to be a possible security risk. • **Always Show Extension**—Specify whether you want the file extension to be displayed with the file name, even if the client system is configured to hide extensions for known file types. • **No Recent Documents**—Specify whether you want to exclude files with this extension from the Recent Documents folder.
Shell New Enabled	Specify whether you want to include this file type in the submenu that is displayed when end users click New on the context menu in Windows Explorer.

Verb Settings for a File Extension

The settings that are displayed when you select a verb under a file extension in the File Extensions view (for an App-V 5 package) or the Shortcuts view (for an App-V 4.x package) are organized into the following main categories:

- General
- Dynamic Data Exchange

General Settings

Use the General area for a verb to specify details such as the name and description of the verb.

Table 12-27 • General Settings for a Verb Under a File Extension

Setting	Description
Target Exe	***Version*** *• This setting is available for App-V 5 packages.* Specify the path to the .exe file for which you are creating a verb for the file extension association.

Table 12-27 • General Settings for a Verb Under a File Extension (cont.)

Setting	Description
Name	Enter the name of the verb, such as **Open** or **Print**, that you want to be used when an end user right-clicks a file with the selected extension and then clicks the corresponding command. To include an underlined letter that indicates that end users can click the letter to select the command, precede that letter with an ampersand (&). For example, to display **Open** (with an underlined letter *O*) on the context menu for this file extension, enter the following: **&Open**
Display Name	Enter the text that you want to display for this verb on the context menu that Windows Explorer displays when an end user right-clicks a file with the associated file extension. To include an underlined letter that indicates that end users can click the letter to select the command, precede that letter with an ampersand (&). For example, to display **Open with SampleApp** (with an underlined letter *O*) on the context menu for this file extension, enter **`&Open with SampleApp`**. This setting is optional. If you do not specify a display name, the name of the verb as it appears in the Name setting is used on the context menu for a file with this file extension on the client system. Note that if you use one of the canonical verbs—such as open, print, or find—and you do not specify a display name, Windows automatically localizes the verb on each system.
Arguments	Enter the command-line arguments for the verb. ***Note*** • *Verify that the syntax is correct because the Virtual Package Editor does not do this.* ***Tip*** • *Use %1 in the argument in place of the file name. For example, if `-p %1` is the argument for the verb, and the end user right-clicks the file `C:\File.ext` and then clicks the command for this verb, the command-line argument becomes `-p C:\File.ext`. In some cases, it is necessary to enclose the %1 argument in quotation marks—as in **`"%1"`**—to correctly handle file names that contain spaces.*

Dynamic Data Exchange Settings

If your App-V application supports dynamic data exchange (DDE), use the Dynamic Data Exchange area for a verb to specify DDE settings for the verb.

Table 12-28 • Dynamic Data Exchange Settings for a Verb Under a File Extension

Setting	Description
DDE Command	Enter the DDE command for the verb. ***Note*** *• Verify that the syntax is correct because the Virtual Package Editor does not do this.* ***Tip*** *• Use %1 in the argument in place of the file name. In some cases, it is necessary to enclose the %1 argument in quotation marks—as in* ***"%1"****—to correctly handle file names that contain spaces.*
DDE Ifexec	Enter the DDE command that you want to use if the DDE conversation cannot be initiated.
DDE Application	Enter the application name that you want use to establish the DDE conversation. If you leave this setting blank, the DDE Command setting is used as the application name.
DDE Topic	Enter the name that you want to use as the topic name of the DDE conversation. If you leave this setting blank, *System* is used as the topic name.

Files and Folders View

The Files and Folders view is where you specify the files and folders that are in the App-V package. This includes folders and files that are in the App-V package's root folder, the virtual file system (VFS) folder, and the SoftGridUserSettings folder. This view also lets you extract folders and files from the App-V package file (.sft) to a location that you specify.

Icons in the Files and Folders View

The Files and Folders view uses different icons to help you distinguish between different types of files and folders. Following is a list of the possible icons in the Files and Folders view.

Table 12-29 • Icons in the Files and Folders View

Icon	Description
	This icon identifies a folder.

Table 12-29 • Icons in the Files and Folders View (cont.)

Icon	Description
	This icon identifies that the folder's **Isolation** setting is set to **Override**. This means that the App-V application sees only the file content of the folder that is inside the App-V package. For an App-V 5.x package, this setting is inherited by all subfolders. If this folder's **Isolation** property was set to **Merge** instead, the App-V application would see a merged view of the file content inside the App-V package and of the file content of the corresponding folder on the physical client system, and no overlay icon would appear on top of the folder icon.
	This icon identifies a file.
	This icon identifies a font file.

Settings in the Files and Folders View

When you select a file or folder in the Files and Folders view, the following settings are available.

Table 12-30 • File and Folder Settings

Setting	Description
Name	Enter the name of the file or folder in the App-V package.
Short File Name	Enter the name of the file or folder using the 8.3 format.
Path	This read-only setting shows the location of the file or folder in the App-V package.
Isolation	Specify whether you want the selected folder in the App-V package to override the corresponding folder on the client system. Available options are: • **Override**—The App-V application sees only the file content of the folder that is inside the App-V package. For an App-V 5.x package, this setting is inherited by all subfolders. Overriding the isolation setting is also referred to as "Fully Virtualized". • **Merge**—The App-V application sees a merged view of the file content inside the App-V package and of the file content of the corresponding folder on the physical client system. ***Note*** • *This setting applies only to folders and not to files.* ***Note*** • *For App-V 4.x packages, setting this option to **Override** automatically sets the read-only **VFS Path** setting, and setting this option to **Merge** automatically clears the **VFS Path** setting.*

Table 12-30 • File and Folder Settings (cont.)

Setting	Description
Size	This read-only setting shows the size of the file. This setting applies to files; it does not apply to folders.
Attributes	To set various attributes for the selected file or folder, use the following settings: • **Read-Only**—Specify whether the file is read-only—protected from being changed or accidentally deleted. • **Hidden**—Specify whether the file or folder is visible in directory listings when default folder viewing options are enabled. • **System**—Specify whether the file or folder is a system file or folder that the operating system uses. • **Archive**—Specify whether the file or folder should be archived. Some applications use this attribute to determine whether to back up a file or folder. • **Normal**—Specify whether the file should have its other attributes configured. Selecting True for this setting is valid only if False is selected for the other True-False attributes. • **Not Content-Indexed**—Specify whether you want to avoid indexing the contents of the file or folder for faster searching.
Created	This read-only setting shows the date and time when the file or folder was created.
Modified	This read-only setting shows the date and time when the file or folder was last modified.
Source Path	If the file has not yet been saved as part of the App-V package, this read-only setting shows the fully qualified path of the source file. This setting applies to files; it does not apply to folders.
Register Font	Specify whether you want the font to be registered in the virtual environment.
Feature Block 1	Specify whether the file is part of the primary feature block, the part of the App-V package that is required to start the application. This setting applies to files; it does not apply to folders.
GUID	This read-only setting shows the globally unique identifier (GUID) that is associated with the file or folder.
Package Version	For a file, this read-only setting shows the version number of the App-V package that corresponds with the last time that the file was modified. For any folder other than the root folder, this read-only setting shows the latest version number of the App-V package.

Table 12-30 • File and Folder Settings (cont.)

Setting	Description
Data Type	Specify the data type of the file or folder. Available options are: • **Application Data**—Changes to the file or folder are saved for all users of the App-V package on the client system. • **User Data**—Changes to the file or folder are saved for only the logged-on user. • **Unspecified**—The data type of the file or folder is not configured.
VFS Path	This read-only setting contains the path that is used by the App-V client to overlay the virtual file or folder onto the corresponding physical location. It is always set for files that are inside of the VFS folder. And it is set for folders found inside the VFS folder only when the folder's **Isolation** setting has been set to **Override**. This indicates that the folder should be fully virtualized.

General Information View

Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

The General Information view contains basic information about your virtual package. It contains a History pane, plus a number of settings that you can configure.

History Pane

Version • *The History pane is available for App-V 4.x packages.*

The History pane in the General Information view shows read-only information such as each date on which the package was saved, the GUID that corresponds with each saved version, and the version of App-V that was used when building the App-V package. Each time that you save your .sft file, the Virtual Package Editor adds a new history entry to the History pane.

General Information Settings

The General Information view settings are organized into the following main categories:

- App-V Settings
- Advanced Settings
- App-V Server URL Settings

App-V Settings

Use the App-V area of the General Information view settings to view or specify basic information such as the name of the virtual package and details such as the package GUID and version number. The following settings are available in this area.

Table 12-31 • App-V Settings in the General Information View

Setting	App-V Version	Description
Name	App-V 4.x, App-V 5	Enter a name for the App-V package. ***Tip*** • *If your virtual package contains multiple applications, you can specify the name that identifies the entire package. For example, Microsoft Office could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
Publisher	App-V 5	Displays the publisher (manufacturer) of the applications contained in this App-V package.
Version	App-V 5	Displays the main software version of the applications in this App-V package.
Comments	App-V 4.x, App-V 5	Enter a short description of the App-V package. This setting is optional.
OS	App-V 4.x, App-V 5	The OS setting and its subsettings let you specify one or more operating systems on which the application can be run. If the application is operating system independent, select False for all of the OS subsettings.
Package GUID	App-V 4.x, App-V 5	This read-only setting shows the globally unique identifier (GUID) that is associated with the App-V package.
Version GUID	App-V 5	This read-only setting shows the globally unique identifier (GUID) that is associated with the version (revision) of the App-V package.
Package Version	App-V 4.x, App-V 5	This read-only setting shows the version number of the App-V package.
Root Folder Mapping	App-V 5	Directory that the root folder of the package is mapped to during package creation.
Minimum Client Version	App-V 4.x, App-V 5	This read-only setting shows the minimum version number of the Application Virtualization Client that is required to use the App-V package.

Table 12-31 • App-V Settings in the General Information View (cont.)

Setting	App-V Version	Description
Root Folder Name	App-V 4.x	This setting specifies the root folder of the App-V package's file system. During run time, the Application Virtualization Client mounts the package's file system to the App-V virtual drive; the Q drive is the default. The long and short names of the root folder must be unique because two packages with the same root folder name cannot be deployed simultaneously.
Feature Block 1 Size	App-V 4.x, App-V 5	This read-only setting indicates the size of the primary feature block, the part of the App-V package that is required to start the application.
Total File Size	App-V 4.x, App-V 5	This read-only setting indicates the size of the entire package.
Compressed	App-V 4.x	Specify whether you want the App-V package's contents to be compressed.
Enforce Security Descriptors	App-V 4.x	This read-only setting indicates whether security descriptors of the application in the App-V package are enforced after it is deployed to the client system.
Allow Local Interaction	App-V 4.x	Specify whether you want named objects (events, mutexes, semaphores, file mappings, and mailslots) and COM objects to be created in the global namespace. Available options are: • **Yes**—Named objects and COM objects are created in the global namespace, allowing virtual applications to interact with the applications of the client operating system. • **No**—Named objects and COM objects are isolated inside the virtual environment.

Advanced Settings

Use the Advanced Settings area of the General Information view or specify advanced App-V settings such as named object interaction, COM object interaction, and browser helper object settings. The following settings are available in this area

Table 12-32 • Advanced Settings in the General Information View

Settings	App-V Version	Description
Named Objects Interaction	App-V 5	Enable this option to allow all named objects to interact with the local system. The default setting is **False** to keep these objects isolated from the local system.
COM Objects Interaction	App-V 5	Enable this option to allow COM to interact with the local system. The default setting is **Isolated** to keep the COM components isolated from the local system.

Table 12-32 • Advanced Settings in the General Information View (cont.)

Settings	App-V Version	Description
COM In Process Enabled	App-V 5	Enable this option to allow in-process COM to interact with the local system. The **COM Objects Interaction** option also has to be set to **Integrated** for this option to have effect.
COM Out of Process Enabled	App-V 5	Enable this option to allow out-of-process COM to interact with the local system. The **COM Objects Interaction** option also has to be set to **Integrated** for this option to have effect.
Full VFS Write Mode	App-V 5	Set to **True** to enable a virtual application to write to its VFS files and folders. The default value is **False**.
Enable Browser Helper Objects	App-V 5	This is a new setting in App-V 5.1 that allows enabling or disabling browser helper object extensions. Select **True** to enable this settings. The default behavior is to leave it at **True**. This setting will appear disabled for packages that do not contain any browser help object extensions.

App-V Server URL Settings

Use the App-V Server URL area of the General Information view settings to specify the location from which the App-V package is streamed. The following settings are available in this area.

Table 12-33 • App-V Server URL Settings in the General Information View

Setting	App-V Version	Description
Protocol	App-V 4.x	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: ● **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. ● **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. ● **FILE**—The App-V package are streamed from a file share. ● **HTTP**—The hypertext transport protocol streams the App-V package. ● **HTTPS**—The secure hypertext transport protocol streams the App-V package.

Table 12-33 • App-V Server URL Settings in the General Information View (cont.)

Setting	App-V Version	Description
Host	App-V 4.x	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter **%SFT_SOFTGRIDSERVER%** to indicate an environment variable. ***Note*** *• If you enter* ***%SFT_SOFTGRIDSERVER%****, you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Port	App-V 4.x	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.
Path	App-V 4.x	Specify the relative path on the virtual application server where the App-V package is stored. This is also the path from which the App-V package is streamed. If the App-V package is stored in a subdirectory of CONTENT, the path must be specified in this setting; otherwise, you can leave this setting blank.

Registry View

The Registry view enables you to define registry keys, values, and data for your App-V package. This view also lets you configure isolation options for selected registry keys. Isolation options indicate how the isolation environment provides access to system resources that the application needs: you can choose to override one or more keys on the client system, or you can choose to create a merged view of one or more keys for the virtual environment.

Note that the registry entries that are configured in the Registry view affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Icons in the Registry View

The Registry view uses different icons to help you distinguish between different types of registry entries. Following is a list of the possible icons in the Registry view.

Table 12-34 • Icons in the Registry View

Icon	Description
	This icon identifies for the root node—the Registry explorer. This icon also identifies the MACHINE and USER predefined keys.
	This icon identifies a registry key.
	This icon identifies a registry key that is configured to override the registry content on the physical client system.
	This icon identifies a REG_NONE, REG_SZ, REG_EXPAND_SZ, or REG_MULTI_SZ registry value.
	This icon identifies a REG_BINARY, REG_DWORD, or REG_QWORD registry value.

Settings in the Registry View

The Registry view contains the registry entries that are configured for your App-V package. When you select a registry key or value in the Registry view, the following settings are available.

Table 12-35 • Registry Key and Value Settings in the Registry View

Setting	Description
Name	Enter the name of the selected registry key or value.
Value Data	Enter the data for the selected registry value, or (for a registry key) enter the data for the selected key's default value.
Value Type	Select the appropriate type of registry data for the selected registry entry. If you select the REG_QWORD type, ensure that the operating system of the client system supports it. This setting applies to registry values; it does not apply to registry keys. The default value of a registry key is always the REG_SZ type of value.

Table 12-35 • Registry Key and Value Settings in the Registry View (cont.)

Setting	Description
Isolation	Specify whether you want the selected registry key in the App-V package to override the corresponding key on the client system. Available options are: • **Override**—The App-V application sees the registry content that is inside the App-V package for this key and all subkeys. Thus, the application does not see any registry content from the physical client system. • **Merge**—The App-V application sees a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If the registry key has subkeys on the physical client system but not in the App-V package, these keys are merged into the registry view that is available to the App-V application. However, registry values that are on the physical client system and that are in registry keys that also exist in the App-V package are not merged into the App-V application's registry view. ***Tip*** • *If your virtual package includes a registry key that has multiple subkeys whose Isolation setting should be configured with the same value, you can quickly change the value of the Isolation setting for all of that key's subkeys simultaneously. You can also quickly configure the Isolation setting for all of the subkeys that belong to multiple parent keys. To learn how, see Configuring the Isolation Setting for All of the Subkeys Under One or More Keys.* This setting applies to registry keys; it does not apply to registry values.

Shortcuts View

The Shortcuts view lets you define the targets for your virtual application. This view also lets you define entry points, such as shortcuts, for each target. Entry points enable end users to launch each target in an App-V package from within the virtual environment.

Icons in the Shortcuts View

The Targets explorer in the Shortcuts view uses different icons to help you distinguish between different types of items. Following is a list of the possible icons in the Shortcuts view.

Table 12-36 • Icons in the Shortcuts View

Icon	Description
	This icon identifies the root node—the Targets explorer.
	This icon identifies a target in the primary App-V package. It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.

Table 12-36 • Icons in the Shortcuts View (cont.)

Icon	Description
	This icon identifies a container that holds the shortcuts that are associated with a target.
	This icon identifies a shortcut.
	This icon identifies a container that holds the environment variables that are associated with a target.
	This icon identifies an environment variable.
	This icon identifies a container that holds the file extensions that are associated with a target.
	This icon identifies a file extension.
	This icon identifies a verb for a file extension.
	This icon identifies a container that holds the scripts that are associated with a target.
	This icon identifies a script.

Settings in the Shortcuts View

Use the Targets explorer in the Shortcuts view to define each target in your App-V package. Under each target, you can configure associated shortcuts, environment variables, file extensions, and scripting. The settings that are displayed in the Shortcuts view differ, depending on what type of item you select in this view. For descriptions of each of the settings in the Shortcuts view, see the following:

- Target Settings
- Shortcut Settings
- Environment Variables View
- File Extension Settings
- Verb Settings for a File Extension
- Scripting Settings

Target Settings

When you select a target in the Shortcuts view, the following settings are available.

Table 12-37 • Target Settings in the Shortcuts View

Setting	Description
Name	Enter the name of the target.
Target	Enter the path and file name of the file in the App-V package or on the client system that should be launched when end users launch the target's shortcut or other entry point. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the target file.
Icon	Enter the path to the icon file (.ico, .exe, or .dll) that contains the icon resource for the shortcut. The location can be in the App-V package, or on your computer or the network. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the icon. If the icon file that you specify contains more than one icon resource, after the icon path, add a comma and then the index number. For example, 0 refers to the first icon in the file, 1 refers to the second icon, and 2 refers to the third icon. To specify the third icon in **C:\MyIcons.dll**, enter the following: `C:\MyIcons.dll,2`
Target Version	Enter the version number of the target.
Arguments	Enter the command-line arguments for the shortcut. These arguments work in the same way as any other command-line arguments. For example, you can link a file to an executable file or cause an executable file to run silently by passing command-line arguments. ***Note*** • *Verify that the syntax is correct because the Virtual Package Editor does not do this.*
Working Directory	Enter the working directory for the target. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to select or create the directory. The working directory is the default directory that is displayed in standard file-opening and file-saving dialog boxes, as well as the current directory used by the App-V application.
Terminate Children	Specify whether you want all of the applications and processes that were launched by the App-V application to be closed when the end user exits the App-V application.

Note • *These Target Settings are displayed for App-V 4.xpackages only. For App-V 5.0 packages, these settings are under Shortcut Settings.*

Shortcut Settings

When you select a shortcut in the Shortcuts view, the following settings are available.

Table 12-38 • Shortcut Settings in the Shortcuts View

Setting	Description
Display Name	Enter the name of the shortcut as it should appear on the client system.
Location	Enter the path to the folder that contains the shortcut file. As an alternative to manually entering a value, you can click the Browse Your Computer button or select a predefined folder from the drop-down list.
Target	Enter the path and file name of the file in the App-V package or on the client system that should be launched when end users launch the target's shortcut or other entry point. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the target file. ***Note*** *• App-V 5.0 packages only. For App-V 4.x packages, this setting is under Target Settings.*
Target Version	Enter the version number of the target. ***Note*** *• App-V 5.0 packages only. For App-V 4.x packages, this setting is under Target Settings.*
Arguments	Enter the command-line arguments for the shortcut. These arguments work in the same way as any other command-line arguments. For example, you can link a file to an executable file or cause an executable file to run silently by passing command-line arguments. ***Note*** *• Verify that the syntax is correct because the Virtual Package Editor does not do this.* ***Note*** *• App-V 5.0 packages only. For App-V 4.x packages, this setting is under Target Settings.*
Working Directory	Enter the working directory for the target. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to select or create the directory. The working directory is the default directory that is displayed in standard file-opening and file-saving dialog boxes, as well as the current directory used by the App-V application. ***Note*** *• App-V 5.0 packages only. For App-V 4.x packages, this setting is under Target Settings.*

Table 12-38 • Shortcut Settings in the Shortcuts View (cont.)

Setting	Description
Icon	Enter the path to the icon file (.ico, .exe, or .dll) that contains the icon resource for the shortcut. The location can be in the App-V package, or on your computer or the network. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the icon. If the icon file that you specify contains more than one icon resource, after the icon path, add a comma and then the index number. For example, 0 refers to the first icon in the file, 1 refers to the second icon, and 2 refers to the third icon. To specify the third icon in **C:\MyIcons.dll**, enter the following: `C:\MyIcons.dll,2` ***Note*** • *App-V 5.0 packages only. For App-V 4.x packages, this setting is under Target Settings.*
Application User Model Id	Specify the `AppUserModelId` that is associated with the target of this shortcut. ***Note*** • *App-V 5.0 packages only.*

Scripting Settings

When you select a script in the Shortcuts view, the following settings are available.

Table 12-39 • Scripting Settings in the Shortcuts View

Setting	Description
Event	Select the timing for the script that you want to launch. Available options are: • **Pre-stream**—The script or executable file runs after the end user launches the App-V application, but before feature block 1 of the application is streamed to the client system and before the virtual environment is set up. This type of script or executable file is run outside the virtual environment. • **Post-stream**—The script or executable file runs after the end user launches the App-V application and after feature block 1 of the application is streamed to the client system, but before the virtual environment is set up. This type of script or executable file is run either inside or outside the virtual environment. • **Pre-launch**—The script or executable file runs after the end user launches the App-V application, after feature block 1 of the application is streamed to the client system, and after the virtual environment is set up. This type of script or executable file is run either inside or outside the virtual environment. • **Post-launch**—The script or executable file runs after the App-V application is launched, but before the end user has access to the application. This type of script or executable file is run either inside or outside the virtual environment. • **Post-shutdown**—The script or executable file runs after the App-V application has been closed. This type of script is run outside the virtual environment.
Type	Specify the type of script that you want to be run. Available options are: • **Single command (HREF)**—The App-V package references an external script or an executable file. The contents of the script are launched directly on the client system. The Command Prompt window is not displayed unless the process that is being called opens it. • **Command script (SCRIPTBODY)**—The contents of the script are stored in the App-V package and copied to a temporary .bat file in the root folder (typically under the Q drive) of the App-V package on the client system. The .bat file is launched from a visible Command Prompt window. You can use either type of scripting to call an executable file that exists in the folder on the virtual application server where the App-V package is stored.
Protect	Specify whether to run the script or executable file inside the virtual environment. Available options are: • **Yes**—The script or executable file is run inside the virtual environment. Protected scripts are useful for troubleshooting issues in the virtual environment. • **No**—The script or executable file is run outside the virtual environment. Unprotected scripts are useful for modifying the client system.

Table 12-39 • Scripting Settings in the Shortcuts View (cont.)

Setting	Description
Wait	Specify whether to wait for the script or executable file to complete before continuing to the next scheduled task—either another script or the appropriate subsequent event.
Timeout	Enter the maximum number of seconds to wait for the script or executable file to complete before continuing. To wait until the script or executable file completes, enter the number 0 or leave this setting blank.
Success Result	Enter the return code that indicates that the script or executable file finished successfully. This setting is optional. For information on triggering the appropriate behavior if the script fails or succeeds, see Causing the App-V Application to Close After a Script Failure. ***Note*** *• If you specify Post-shutdown for the Event setting, any value that you specify for the Success Result setting is ignored.*
Abort Result	Enter the return code that indicates that the script or executable file failed. This setting is optional. For information on triggering the appropriate behavior if the script fails or succeeds, see Causing the App-V Application to Close After a Script Failure. ***Note*** *• If you specify Post-shutdown for the Event setting, any value that you specify for the Abort Result setting is ignored.*

Virtual Services View

Windows services are executable files that Windows–based systems run in the background to manage various system tasks. A service is an executable file, but it must be designed as a service; you cannot automatically use an arbitrary executable file as a service. Windows services can be configured to run every time that the system starts or on demand when needed. The Virtual Services view enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.

The Virtual Services view shows the services that are configured for your App-V package. The Virtual Services view settings are organized into the following main categories:

- General Settings
- Error Handling Settings

General Settings

Use the General area of the Virtual Services view settings to specify information such as the name and location of the virtual service, as well as the type of service. This area is also where you specify when the service should be started. The following settings are available in this area:

Table 12-40 • General Settings in the Virtual Services View

Setting	Description
Name	Enter the name of the service. The name that you enter is used on the service's Properties dialog box. (To access an installed service's properties: In the Services administrative tool, right-click the service and then click Properties.) The maximum number of characters that is allowed is 256. The forward slash (/) and the backslash (\) are not valid characters for service names. The case of the name that you enter is preserved in the service control manager. Display name comparisons are always case-insensitive.
Display Name	Enter the name that you want to be displayed for this service in the service control manager and in other user interfaces. The maximum number of characters that is allowed is 256. The case of the display name that you enter is preserved in the service control manager. Display name comparisons are always case-insensitive.
Path to Executable File	Enter the fully qualified path to the executable file for the service. If the path contains one or more spaces, surround the path with quotation marks. You can include arguments that you want to be passed for a service that starts automatically. As an alternative to manually entering the path, you can click the Browse This Package button to browse to the executable file.
Service Type	Select the type of service that you are installing.
Service Is Interactive	Specify whether you want the service to be able to interact with users.
Startup Type	Specify when to start the service. Available options are: • **Automatic**—The service starts automatically when the system starts. • **On Demand**—The service starts when the service is requested through the service control manager. • **Never (Disabled)**—The service cannot be started.

Table 12-40 • General Settings in the Virtual Services View (cont.)

Setting	Description
Error Control	Select the appropriate severity of the error to indicate the action that the service control manager should perform if the service fails to start. Available options are: • **Ignore**—Ignore the error and continue with the service startup. • **Normal**—Log the error and continue with the service startup. • **Severe**—Log the error. If the last-known good configuration is being started, continue with the service startup. Otherwise, restart the system with the last-known good configuration. • **Critical**—Log the error. If the last-known good configuration is being started, the service startup fails. Otherwise, restart the system with the last-known good configuration.
Group	Enter the name of the load-ordering group, if any, of which this service is a member. Note that this setting can override the value of the Service Dependencies setting.
Group Dependencies	To specify one or more load-ordering groups that this service requires, click the ellipsis button (...) in this setting. When you do so, the Edit Value dialog box opens, enabling you to specify one or more groups. Enter each group on a separate line. The system attempts to start at least one member of the load-ordering group before starting this service.
Service Dependencies	To specify one or more services that this service requires, click the ellipsis button (...) in this setting. When you do so, the Edit Value dialog box opens, enabling you to specify one or more services. Enter each service on a separate line. The system attempts to start at least one member of the load-ordering group before starting this service.

Error Handling Settings

Use the Error Handling area of the Virtual Services view settings to specify what behavior should occur if the service fails.The following settings are available in this area:

Table 12-41 • Settings in the Virtual Services View

Setting	Description
Reset Period	Specify the amount of time (in seconds) between the reset intervals for the service's failure count. As an alternative, you can select one of the values from the list in this setting. The service control manager counts the number of times that the service has failed since the system was last restarted. When this interval has elapsed, the count is reset to the number 0 if the service has not failed during the reset period. When the service fails, the system performs an action that is specified for the First Error setting, the Second Error setting, or the Additional Errors setting, depending on how many errors have occurred since the last failure count reset or system restart. To indicate that the failure count should never be reset, select Never or enter a value of -1.
Reboot Message	Specify the message that should be displayed before the computer is restarted in response to an error. Note that Reboot the Computer must be listed as one of the action types for the First Error setting, the Second Error setting, or the Additional Errors setting; otherwise, the Reboot Message setting is ignored.
Command	Specify the command line that should be run if the Run a Command option is selected for the First Error, Second Error, or Additional Errors setting, and the first, second, or subsequent error occurs during service startup. Programs or scripts that you specify should not require input from end users. The command line that you specify is used to create a new process that runs under the same account as the service.
First Error	Select the action that you want the service control manager to perform the first time that the service fails.
First Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the First Error setting. As an alternative, you can select one of the values from the list in this setting.
Second Error	Select the action that you want the service control manager to perform the second time that the service fails.
Second Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the Second Error setting. As an alternative, you can select one of the values from the list in this setting.
Additional Errors	Select the action that you want the service control manager to perform the third and subsequent times that the service fails.

Table 12-41 • Settings in the Virtual Services View (cont.)

Setting	Description
Subsequent Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the Additional Errors setting. As an alternative, you can select one of the values from the list in this setting.

13

Creating Customized Virtual Applications

You can use AdminStudio to create customized virtual applications in the Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual application formats.

Information about creating virtual applications is organized into the following sections:

- About Virtualization
- About the AdminStudio Virtualization Interface
- Creating Microsoft App-V Packages
- Creating Citrix Profiles
- Creating ThinApp Applications

About Virtualization

Note • *This section provides a description of virtualization in general for those that are not familiar with it. It does not represent the architecture of any specific vendor.*

Virtualization enables you to isolate an application in its own environment so that it does not conflict with existing applications or modify the underlying operating system.

- Limitations of a Standard Installation Environment
- Benefits of Application Virtualization

Limitations of a Standard Installation Environment

A typical Windows application has dependencies on components that are shared by multiple applications. Applications access these shared system resources, such as the registry or Windows system files. When an installation author recognizes that their application references a shared system component, they include a merge module to install that component.

When one of these shared components is installed, it is possible that a previously installed version of the same component could be overwritten; this may cause the existing application to break. A similar problem could occur when one of these applications containing a shared component is uninstalled. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment.

The following diagram provides an example of two conflicting installed applications.

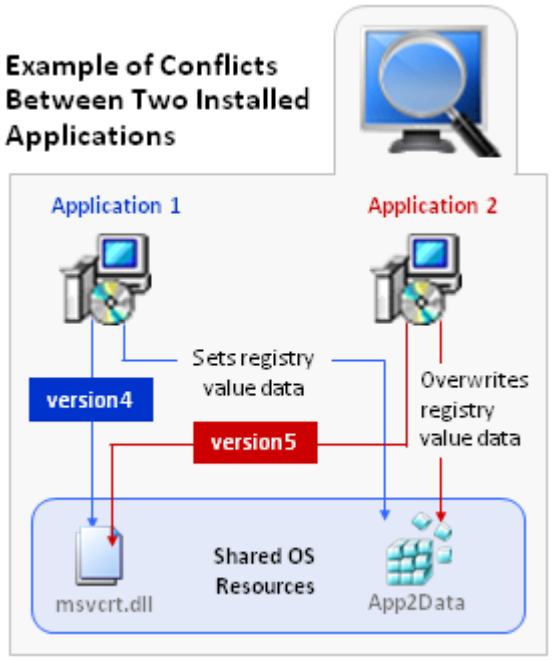


Figure 13-1: Example of Conflicts Between Two Installed Applications

Benefits of Application Virtualization

Virtual applications run in virtual environments that keep the application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

Even though virtual applications are not installed on the local machine, they exhibit the same functionality and access to local services as locally installed applications, and also nearly the same performance characteristics.

The following diagram provides an example of how application virtualization would solve the conflicts that are shown in the previous example.

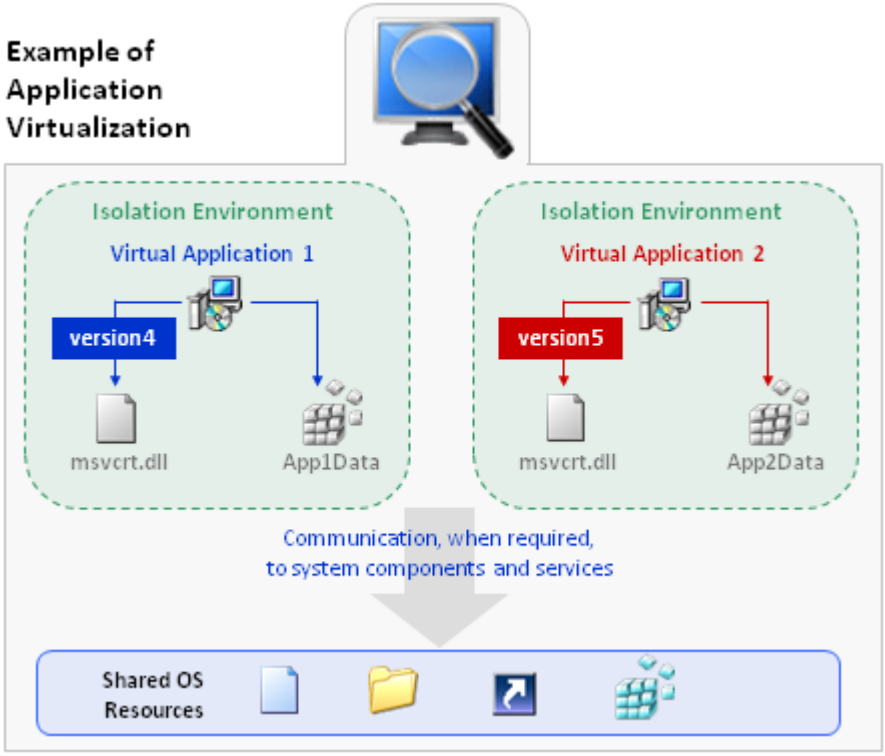


Figure 13-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files, and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies compatibility testing.

About the AdminStudio Virtualization Interface

Project • *The Microsoft App-V, ThinApp, and Citrix Assistants are available in the following project types:*

- *Basic MSI*
- *MSI Database (Direct Edit Mode)*
- *Transform (Direct MST Mode)*

AdminStudio provides the Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant to help you author a virtual application. You cannot configure a virtual application's options using the Installation Designer.

Information about the interface of these Assistants is organized in the following topics:

- About the Virtualization Assistant Tabs
- Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant

- Navigating in a Virtualization Assistant
- Opening the Installation Designer
- Showing or Hiding the Virtualization Assistants

About the Virtualization Assistant Tabs

When you create a new Basic MSI or MSI Database project, the **Microsoft App-V**, **VMware ThinApp**, and **Citrix XenApp** tabs are displayed in the AdminStudio interface. The home page of each Assistant has a diagram that illustrates the process of creating a virtual application using that technology.

You can work within these Assistants to create a project and configure its options and requirements. You can also use the Project Assistant or the Installation Designer to define the traditional Windows Installer version of your product installation.

How the Virtualization Assistants Work

When you create a new Basic MSI or MSI Database project, the **Microsoft App-V**, **VMware ThinApp**, and **Citrix XenApp** tabs are displayed in the AdminStudio interface.

The **Project Assistant** tab and the **Installation Designer** tab show the underlying framework for your product's Windows Installer–based installation. Some of these product elements are also displayed in the virtualization Assistants, where you can configure a virtual application's options and requirements.

Integration with the Project Assistant and the Installation Designer

Information that you enter in a virtualization Assistant is saved directly to the underlying project file. The Microsoft App-V Assistant, ThinApp Assistant, Citrix Assistant, Project Assistant, and Installation Designer run simultaneously. Any changes that you make in one are reflected instantly in the other. For example, if you remove a file in one of the virtualization Assistants, that file is no longer available in your project, and it does not appear in the Project Assistant or the Installation Designer.

Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant

The left column on each page of the virtualization Assistants contains one or more lists of links to help you in creating your installation and finding information:

- **More Options**—Provides additional configuration options relating to the specific virtualization Assistant page. These are less common options that complete the functionality of the Assistant.
- **Other Places**—The view in the Installation Designer that corresponds to the current virtualization Assistant page. Clicking the link launches the full Installation Designer and activates that view.
- **Help Links**—This list provides links to help topics pertinent to the current virtualization Assistant page.

Navigating in a Virtualization Assistant

Task ***To navigate from one page of a virtualization Assistant to another, do one of the following:***

- To navigate directly to a specific page, click the appropriate icon in the navigation bar at the bottom of the page.
- To follow the assistant steps sequentially, do one of the following:
  - Click the Next or Back arrow buttons to move forward or backward.
  - Press CTRL+TAB to move to the next page and CTRL+SHIFT+TAB to move to the previous page.
- To move back to the Home page and view the overview diagram, click the Home button on the navigation bar.

Opening the Installation Designer

The **Installation Designer** tab displays the views in the AdminStudio interface. You can use this tab to configure your Windows Installer package. To open a view in the Installation Designer, click the **Installation Designer** tab.

Note • *The Installation Designer and the virtualization Assistants run simultaneously. Any changes that you make in one are reflected instantly in the other.*

Showing or Hiding the Virtualization Assistants

If you want to hide the Assistant for a virtualization technology that you do not use, you can hide it so that its tab is not displayed in the InstallShield interface. Similarly, if one of the virtualization Assistants is hidden, you can choose to display it.

Task ***To show or hide a virtualization Assistant:***

On the **View** menu, click **Microsoft App-V Assistant**, **ThinApp Assistant**, or **Citrix Assistant**.

When an Assistant's command has a check mark next to it, the tab for that Assistant is shown in the InstallShield interface. When the check mark is not displayed, that Assistant is hidden.

Creating Microsoft App-V Packages

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

Information about Microsoft App-V and creating Microsoft App-V packages is presented in the following sections:

- Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant

- Using the Microsoft App-V Assistant to Create an App-V Package
- Microsoft App-V Assistant Reference

Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

The Microsoft App-V Assistant, which you can use to configure and build App-V packages, consists of the following pages:

Table 13-1 • Pages Comprising the Microsoft App-V Assistant

Page	Description
Microsoft App-V Assistant Home Page	Displays a diagram that illustrates the process of creating an App-V package.
Package Information Page	Enter the package name, enter a comment, specify any operating system requirements, and identify the deployment server.
Files Page	View existing files and folders, add and delete files, and set isolation options for selected files and folders. Isolation options specify how the virtual environment provides access to files and folders that the virtual application requests.
Applications Page	Create, delete, include, exclude, or rename App-V package shortcuts.
Registry Page	Add, delete, or modify the registry settings, and set the isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys that the virtual application requests.
Dynamic Suite Composition Page	Use to control virtual application interaction between multiple App-V packages. On this page, you can select one or more packages that need to be linked to this App-V package in order for it to execute correctly. ***Version*** • *This page is available for App-V 4.x packages.*
Build Options Page	[Basic MSI Project mode] Select the releases that you want to build. [Direct Edit or Direct MST mode] To enable the Build function for an App-V package, select the **Build App-V package** option.

For information on Microsoft Application Virtualization and the Microsoft App-V Assistant, see the following topics:

- About Microsoft Application Virtualization (App-V) and the Microsoft App-V Assistant

- Components of an App-V Package
- About the Microsoft App-V Assistant

About Microsoft Application Virtualization (App-V) and the Microsoft App-V Assistant

This section provides an overview of Microsoft Application Virtualization and its infrastructure, and explains the benefits of using the Microsoft App-V Assistant to create App-V packages:

- Overview
- Microsoft Application Virtualization Infrastructure
- Benefits of Using the Microsoft App-V Assistant

Overview

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

Because App-V packages are not installed on the client, there is minimal impact on the host operating system or other applications. As a result, application conflicts and the need for regression testing are dramatically reduced.

Using Microsoft Application Virtualization enables you to centralize the installation and management of deployed applications, and control access to applications. The App-V client presents to the end user a list of applications to which that end user has access.

Microsoft Application Virtualization Infrastructure

The Microsoft Application Virtualization (App-V) infrastructure includes:

- **App-V Sequencer**—The App-V Sequencer converts application data into a format that is compatible with the App-V server and client, producing an App-V package.
- **App-V Server**—An App-V package can be placed on one or more App-V servers so that it can be streamed down to the clients on demand and cached locally.
- **Application Virtualization Client**—The App-V client is the system component that enables the end user to interact with the App-V packages that are available on the App-V server.

Benefits of Using the Microsoft App-V Assistant

Instead of using the App-V Sequencer to create App-V packages, you can use the InstallShield Microsoft App-V Assistant, as shown in the following diagram:

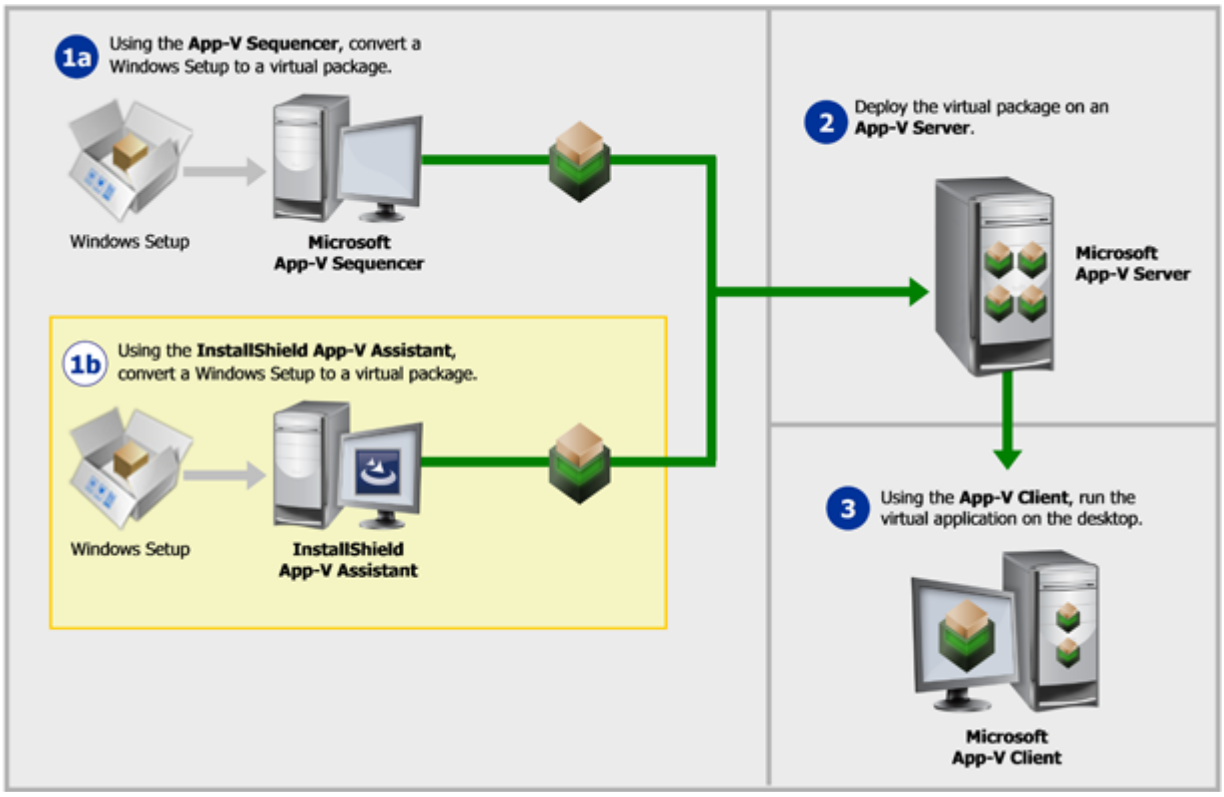


Figure 13-3: Using the Microsoft App-V Assistant to Create an App-V Package

Using the Microsoft App-V Assistant instead of the Microsoft App-V Sequencer to create an App-V package offers the following benefits.

Product Installation on a Clean Machine Is Not Required

The Microsoft App-V Sequencer obtains the information it needs to create an App-V package by installing a package on a clean machine and then comparing the file system snapshot that it took prior to installation with one it takes after installation. To perform this task properly, there are two requirements:

- **The product must be installed on a clean machine**—To ensure that all proper changes made by the installation are captured, sequencing needs to be performed on a clean machine (a computer with only the operating system, necessary service packs, and the App-V Sequencer installed on it). A new clean machine would need to be re-created for each application that is sequenced.
- **The installation directory must be known before sequencing can begin**—In order to sequence the application effectively, you must have detailed knowledge of the how the installation is supposed to work. Prior to beginning the sequencing process, you are required to specify the installation directory for the application being sequenced. This information is often not readily available, and may require you to open the installation in an editing tool, such as InstallShield, in order to find it, or run the installation one time prior to sequencing.

Instead of installing the package, the Microsoft App-V Assistant obtains the information it needs to create an App-V application directly from the installation. You are not required to have any knowledge of settings within the installation, such as the installation directory. Because there is no need to install the application to obtain this information, no permanent changes are made to the local machine and a clean machine is not required.

Ability to Test the App-V Package Immediately After Conversion

To run an App-V package, the App-V client must be installed on the machine. Because sequencing must be performed on a clean machine, which does not have the App-V client installed, you cannot immediately test a newly created App-V package on the same machine where you sequenced it.

The Microsoft App-V Assistant includes a launch utility that allows you to launch and test the App-V package locally immediately after conversion, before distributing it to the App-V server.

This feature requires that the App-V client is installed on the local machine.

Support for Including Diagnostic Tools in the App-V Package

When running a virtual application in its virtual environment, you may at some point want to examine its contents to evaluate or debug it. However, the standard diagnostic tools that you use to examine installed applications (such as the Registry Editor and the Windows Command Prompt window) are not normally available within the virtual environment for App-V 4.x. When a virtual application is running within its virtual environment, applications outside of that virtual environment cannot see into it.

When you use the InstallShield App-V Assistant to create an App-V 4.x package, you can choose to include shortcuts for diagnostic tools in the App-V package; these shortcuts enable you to use `Cmd.exe` and `Regedit.exe` on the local machine, with access to the virtual environment.

Beginning with App-V 5.x, it is no longer necessary to include diagnostic tool shortcuts directly in the App-V package, since the App-V Launcher is capable of launching a Command Prompt window within the virtual environment.

Components of an App-V Package

The files that comprise an App-V package depend on the version of the App-V package.

Components of an App-V 5.x Package (.appv)

The following table describes the main components of an App-V 5.x package (.appv):

Table 13-2 • Components of an App-V 5.x Package

File	Description
.appv	The .appv file is the compressed package file that contains all of the other parts of the package.
[Content_Types].xml	This file contains a list of file extensions that the package supports and the type of content to which each extension type maps.
AppxBlockMap.xml	This file contains a list of files with details such as header size and file size.
AppxManifest.xml	This file contains metadata about the package.
FilesystemMetadata.xml	This file contains information such as short file names, the directory-file hierarchy, and the mapping between the root folder and INSTALLDIR.
Registry.dat	This file contains registry data for the package.

Table 13-2 • Components of an App-V 5.x Package (cont.)

File	Description
StreamMap.xml	This file contains feature block 1 information.

Components of an App-V 4.x Package (.sft)

The following table describes the main components of an App-V 4.x package (.sft):

Table 13-3 • Components of an App-V 4.x Package

File	Description
.sft	The .sft file contains all of the files, registry information, and other configuration details of the package.
Manifest file	This file is an XML file that lists all of the .osd files in an App-V package.
.osd	The .osd files are XML-based files that describe the package's individual targets (or applications) that can be run.
.ico	The .ico files are icons files that are used for published shortcuts and file type associations.
.sprj	This file is the Microsoft App-V Sequencer project file. It contains references to the .sft and .osd files, and to a large number of settings related to the sequencing process.

About the Microsoft App-V Assistant

Information about the Microsoft App-V Assistant is organized into the following sections:

- Process for Authoring an App-V Package Using the Microsoft App-V Assistant
- Supported InstallShield Project Types
- How Transforms are Included in an App-V Package
- How Windows Services Are Integrated into an App-V Package

Process for Authoring an App-V Package Using the Microsoft App-V Assistant

You can use the Microsoft App-V Assistant to convert a Windows Installer package into an App-V package. During this process, you can perform the following tasks:

- **Specify Package Information and Deployment Options**—Specify the package name, root folder name, enter a comment, specify any operating system requirements, and identify the deployment server.
- **Specify Files, Folders, Shortcuts, Registry Settings**—Specify the files, folders, application shortcuts, and registry settings that will be included in the App-V package.
- **Configure Isolation Options**—Set the isolation options for selected files, folders, and registry keys.
- **Build**—Specify build options and build an App-V package.

The following diagram illustrates the creation process for an App-V package:

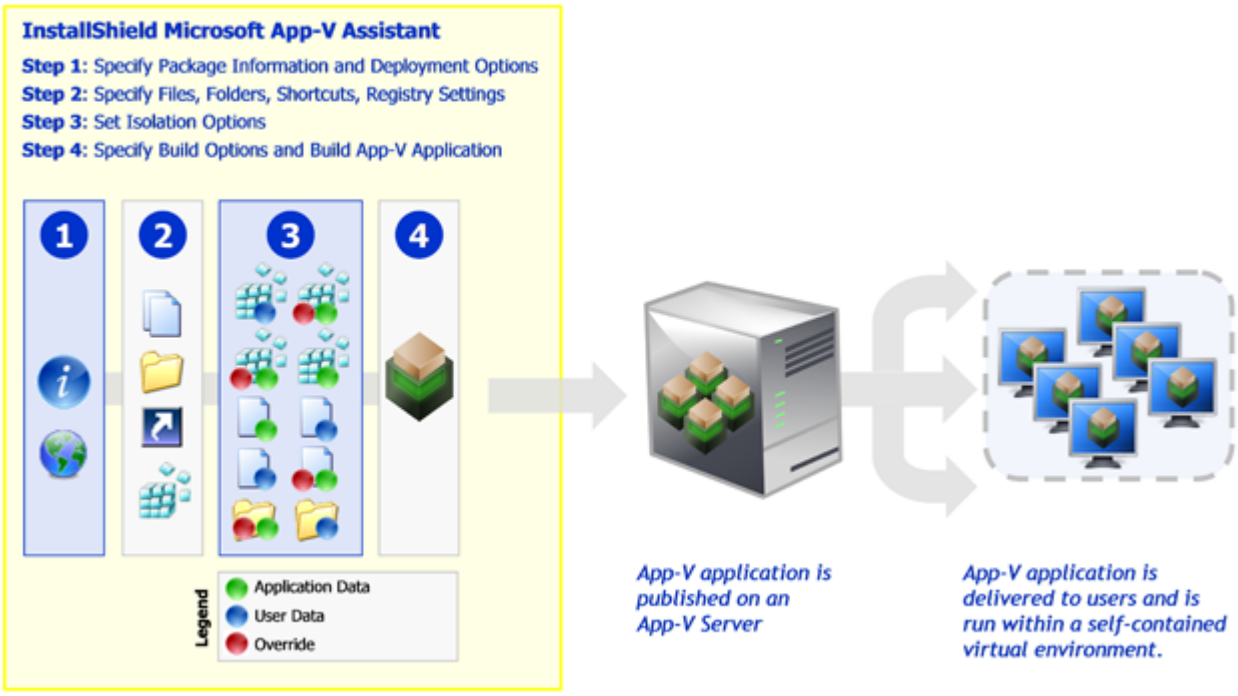


Figure 13-4: Creating an App-V Package

Supported InstallShield Project Types

The **Microsoft App-V** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in an App-V Package

The Microsoft App-V Assistant supports the inclusion of transform files with Windows Installer packages in an App-V package.

- **How transforms are applied when building an App-V package**—When you are building an App-V package, transforms that you have specified are automatically applied to the base Windows Installer (.msi) package to create a temporary package, and then the App-V package is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build an App-V package from that transform file. When you create a new transform file in InstallShield, you specify the root .msi file in the Open Transform wizard. The steps you take to generate an App-V package after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the App-V package, you need to open

one of the transforms in InstallShield (rather than the .msi file). The Open Transform wizard will open, and you will be prompted to specify the root .msi file and which of the existing .mst files you want to include. The steps you take to generate an App-V package after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to an App-V package must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the App-V package is built.*

How Windows Services Are Integrated into an App-V Package

When you use the Microsoft App-V Assistant to convert a Windows Installer package to an App-V package, references to Windows services that are encountered are integrated into the App-V package. In a Windows Installer package, a Windows service may be indicated by either an entry in its `ServiceInstall` table or by a Registry entry for Windows services.

- **ServiceInstall table**—If a Windows Installer package's use of a Windows service is indicated by an entry in the `ServiceInstall` table, the Microsoft App-V Assistant will convert that entry to a standard Registry entry for Windows services.
- **Registry entry**—If a Windows Installer package's use of a Windows service is indicated by a Registry entry for Windows services (perhaps as the result of being repackaged), the Microsoft App-V Assistant does not need to make any changes to support the application's use of the Windows service within the virtual environment.

Start Up and Shut Down Sequences

If an App-V package has an associated Windows service, App-V will start up the Windows service first, in the virtual environment, and then start up the virtual application. You will see the Windows service start up in the Task Manager as a separate process, but App-V will be running the service within the virtual environment.

Upon shut down, App-V will first shut down the virtual application and then shut down the Windows service.

Using the Microsoft App-V Assistant to Create an App-V Package

The steps you need to take to create an App-V package are the following:

Table 13-4 • Steps for Creating an App-V Package Using the Microsoft App-V Assistant

Step #	Description
Step 1	Specifying Package Information and Deployment Options
Step 2	Managing Files in an App-V Package
Step 3	Setting Isolation Options for Folders and Files
Step 4	Modifying Shortcuts to the App-V Package's Executable Files
Step 5	Modifying App-V Package Registry Settings

Table 13-4 • Steps for Creating an App-V Package Using the Microsoft App-V Assistant (cont.)

Step #	Description
Step 6	Setting App-V Package Registry Isolation Options
Step 7	Performing Dynamic Suite Composition ***Version*** • *This information applies to App-V 4.x packages.*
Step 8	Modifying Build Options
Step 9	Building an App-V Package
Step 10	Testing an App-V Package Using the App-V Launcher Tool

Specifying Package Information and Deployment Options

When you are creating an App-V package, the first step is to specify the package name, root folder name, and enter a comment on the Package Information page. On this page, you can also specify any operating system requirements, identify the deployment server, and specify whether to include diagnostic tools with the virtual package. This page also lets you specify upgrade information for your App-V package if appropriate.

Specifying Package Information

The first step in creating an App-V package is to enter information such as the package name.

Task ***To specify package information:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **Package Name** field, enter a name for the virtual package.
3. In the **Comments** field, enter a short description of the App-V package.

Specifying Operating System Requirements

To specify operating system requirements for an App-V package, perform the following steps:

Task ***To specify operating system requirements:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. Set the **Does your App-V package have any specific operating system requirements?** option to one of the following:

- **No**—Select this option if this application will run on all of the listed operating systems. When this option is selected, the operating system check boxes are locked and cannot be changed.
- **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems.

3. If you selected **Yes**, select the operating systems that this application supports, and clear those that this application does not support.

Specifying Upgrade Package Information

InstallShield enables you to specify whether you want to create an upgrade for your App-V package. If you specify that you do want to create an upgrade, you can specify additional information about the upgrade, such as whether to append the version number to the App-V package file name.

Task ***To specify upgrade information:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **More Options** area, click **Upgrade Settings**. The **App-V Package Upgrade Settings** dialog box opens.
3. Do one of the following:
   - To create an upgrade for your App-V package, select the **Enable Upgrade** check box. Then specify which package should be upgraded. If you want InstallShield to include the version number in the package name, select the **Append version number to package name** check box.
   - To avoid creating an upgrade package, clear the **Enable Upgrade** check box.

Specifying the Deployment Server

To specify the deployment server for an App-V package, perform the following steps:

Task ***To specify the deployment server:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. Under **Where will the App-V package be deployed?**, configure the appropriate options. For more information, see Package Information Page.

Including Diagnostic Tools in an App-V Package

Version • *The diagnostic tools are available for App-V 4.x packages. Starting with App-V 5.x, it is no longer necessary to inject diagnostic tool shortcuts directly into the package. The App-V Launcher tool is capable of launching a Command Prompt window within the virtual environment of an App-V 5.x package.*

The Microsoft App-V Assistant lets you specify whether you want to include the Registry Editor and Windows Command Prompt diagnostic tools with your App-V package.

If you include diagnostic tools with your App-V package, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running an App-V package and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Task ***To include diagnostic tools with an App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **More Options** area, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.
3. If you want to include the Registry Editor with your App-V package so that you can use `Regedit.exe` on the local machine and have access to the virtual environment, select the **Registry Diagnostics** option.

   If the **Registry Diagnostics** option is selected, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment.
4. If you want to include the Windows Command Prompt application with your App-V package so that you can use `Cmd.exe` on the local machine and have access to the virtual environment, select the **File System Diagnostics** option.

   If the **File System Diagnostics** option is selected, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment.
5. Click **OK**.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the Registry Diagnostics or File System Diagnostics options on the Diagnostic Tools dialog box, shortcuts to those tools are automatically added to the App-V package.

When an end user runs this App-V package, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When an end user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Managing Files in an App-V Package

The next step in creating an App-V package is to view existing files and folders, add and delete files and folders, and set isolation options for files and folders.

The following tasks are performed on the Files page.

- Adding, Deleting, and Moving Files and Folders in an App-V Package
- Controlling the Display of Predefined Folders
- Specifying the Primary Application Directory

Adding, Deleting, and Moving Files and Folders in an App-V Package

The directories in the destination tree on the Files page of the Microsoft App-V Assistant represent how your application will look when it is installed on to your end users' machines.

On the Files page, you can view all of the files and folders that are currently in your App-V package, add new files and folders to include in the App-V package, and delete files and folders from the App-V package.

Adding Files to an App-V Package

To add files to an App-V package, perform the following steps:

Task ***To add a files to an App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Files** page. The files and folders are listed in the **Microsoft App-V Application** tree, organized by installation directory.

   Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files from Your System

You can also add files or folders to your App-V package on the Files page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and Its Contents) to an App-V Package

To add an existing folder and all of the files and subfolders within it to an App-V package, perform the following steps:

Task ***To add an existing folder to an App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Files** page. The files and folders are listed in the **Microsoft App-V Application** tree, organized by installation directory.
2. Browse through the folder tree to find the folder that you would like to add a folder into.
3. Select the folder and click the **Add Folders** button. The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.
4. Select a folder and click **OK**.

   If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.

5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with App-V options, it is recommended that you select No to indicate that you do not want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

Task ***To create a new folder:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. In the **Microsoft App-V Application** tree, right-click the folder that you want to contain the new folder, and click **New Folder**. InstallShield creates a new folder as a subfolder of the selected folder.
3. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the App-V package folder tree structure, perform the following steps:

Task ***To move a file or folder:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. In in the **Microsoft App-V Application** tree, drag the file or folder that you want to move to the new location.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from an App-V package, perform the following steps:

Task ***To delete a file or folder:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. In in the **Microsoft App-V Application** tree, right-click the file or folder that you want to delete. You are prompted to confirm the deletion.
3. Click **Yes**.

InstallShield removes the selected file or folder.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire project, not just from the App-V package.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the Files page, the Microsoft App-V package tree initially displays the more commonly used predefined folders, such as [CommonFilesFolder] and [ProgramFilesFolder]. You can show and hide predefined folders on this page.

Task ***To display additional predefined folders:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. Right-click the **Microsoft App-V Application** node (or any of the files or folders that are listed), point to **Show Predefined Folder**, and then click predefined folder that you want to show.

   The folders that are currently displayed are preceded by a check mark; the folders that are not currently displayed do not have a check mark.

AdminStudio adds the predefined folder to root level of the Microsoft App-V package tree.

Tip • *To hide a predefined folder, click it in the list of predefined folders.*

Note • *It is not possible to hide* [ProgramFilesFolder].

Specifying the Primary Application Directory

When App-V packages are run on a machine that has Application Virtualization Client installed, they are run from the App-V cache drive.

For optimum performance, the bulk of the application's files should be mounted to this drive at run time. In order to achieve this, it is useful to determine an application's primary application directory so that this directory can be mounted to the App-V literal cache drive when the App-V package is loaded.

When an App-V package is built using InstallShield or any AdminStudio tool, the following series of steps are used to determine an App-V package's primary application directory:

Table 13-5 • Steps to Automatically Determine the Primary Application Directory

#	Step	Description
1	**Explicitly set primary application directory**	If a directory is specified on the Primary Application Directory dialog box. that directory is used. For more information, see File Mapping Dialog Box.
2	**Value of INSTALLDIR variable**	If the Windows Installer package has a value for `INSTALLDIR` (the Windows Installer property that specifies the root destination directory for an installation), that value is used as the primary application directory. ***Note*** • *All Windows Installer packages created by InstallShield or AdminStudio have a value for the* `INSTALLDIR` *property.*
3	**Location of shortcut in a subdirectory of the ProgramFilesFolder**	If one of the .exe targets for a shortcut is in a subdirectory of `ProgramFilesFolder`, the folder directly under `ProgramFilesFolder` is used as the primary application directory. For example: `C:\Program Files\YourApplication`
4	**Location of shortcut in a directory other than ProgramFilesFolder**	If no .exe targets are located in a subdirectory of `ProgramFilesFolder`, the target directory of a shortcut that contains an .exe is used.
5	**ProgramFilesFolder**	If none of the above can be found, the primary application directory is set to `ProgramFilesFolder`. Typically, this would be: C:\Program Files

Explicitly Specifying the Primary Application Directory

To specify the primary application directory for an App-V package, perform the following steps.

Task ***To specify the primary application directory:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. In the **More Options** area, click **Primary Application Directory**. The **Primary Application Directory** dialog box opens, displaying the current primary application directory setting (if one has already been specified).
3. Click the ellipsis button (...). The **Browse for Directory** dialog box opens, listing all of the currently available destination directories for this App-V package.
4. Select one of the listed directories and click **OK**.

Setting Isolation Options for Folders and Files

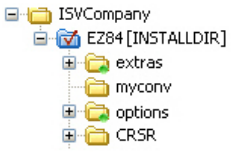
Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

The Microsoft App-V Assistant enables you to configure isolation options for folders (in App-V 4.x and 5.x packages) and for files (in App-V 4.x packages). Isolation options indicate how the isolation environment provides access to system resources that the virtual application needs: you can choose to ignore one or more folders on the client system, or you can choose to create a merged view of one or more folders.

Task ***To configure isolation options for a folder or file:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. Right-click the file or folder that you want to configure and then click **Isolation Options**. The **Options** dialog box opens.
3. Select the appropriate option.
4. Click **OK**.

Files and folders that have a custom isolation option are marked with a special icon:



For information on the various options that are available, see the following:

- Options Dialog Box (for Configuring Isolation Options for a Folder)
- Options Dialog Box (for Configuring Isolation Options for a File)

Inheritance of Isolation Options from Folders to Files

Isolation options for files and folders are always inherited. The App-V isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

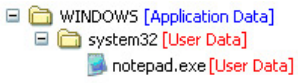


Figure 13-5: Example of Inheritance of Isolation Options from Folders to Files

Modifying Shortcuts to the App-V Package's Executable Files

You define application shortcuts to enable end users to launch an App-V package from within the virtual environment.

By default, the Microsoft App-V Assistant creates App-V packages for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the Applications page.

Caution • *You must define at least one shortcut to enable end users to launch the application from the isolation environment.*

On the Applications page, you can create, delete, include, exclude, or rename executable files, which are derived from the shortcuts in its Windows Installer package.

- App-V Packages and the Virtual Environment
- App-V Shortcut Requirements
- Creating a New App-V Package
- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Excluding vs. Deleting App-V Package Shortcuts
- Renaming a Shortcut

Caution • *If you delete a shortcut on the* ***Applications*** *page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

App-V Packages and the Virtual Environment

Shortcuts provide the most visible entry points for launching the applications in the App-V package. Most App-V packages should have at least one shortcut.

On the Applications page of the Microsoft App-V Assistant, you can define application shortcuts to enable end users to launch an application in the App-V package. The Microsoft App-V Assistant creates shortcuts for any executable files that are added through the Files page. All shortcuts are added to the App-V package and published to the system when the package is published.

To deploy an App-V package—on a local drive or a network share—systems administrators simply need to give end users access to the App-V package.

App-V Shortcut Requirements

For each App-V package, you can define one or more shortcuts. Shortcuts provide the most visible entry points for launching the applications in the App-V package. Most App-V packages should have at least one shortcut.

Creating a New App-V Package

On the Applications page of the Microsoft App-V Assistant, specify the files for which you want to create shortcuts.

Task ***To create a new App-V shortcut:***

1. In the **Microsoft App-V Assistant**, open the **Applications** page.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this App-V package.
3. Select the target for the shortcut that you are creating.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.
5. To include this shortcut in the App-V package, make sure that its check box is selected.

Including an Existing App-V Shortcut

If you want to include a previously excluded shortcut in an App-V package, perform the following steps.

Task ***To include an existing App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Applications** page.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding or Deleting an Existing App-V Package

To prevent a shortcut from being created in the App-V package, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the App-V package, but it will remain in the InstallShield project.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the App-V package and the InstallShield project.

Caution • *If you delete a shortcut on the Applications page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

If you have any unnecessary shortcuts in your project, you can simply exclude them from the App-V package by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding a Shortcut

If you want to exclude one of these shortcuts from being created in the App-V package, perform the following steps.

Task ***To exclude a shortcut:***

1. In the **Microsoft App-V Assistant**, open the **Applications** page.
2. Select the shortcut that you want to exclude and clear its check box.

Note • *When you exclude a shortcut, it is not created in the App-V package, but it remains in the InstallShield project.*

Deleting a Shortcut

To delete a shortcut, perform the following steps.

Task ***To delete a shortcut:***

1. In the **Microsoft App-V Assistant**, open the **Applications** page.
2. Select the shortcut and click **Delete**.

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Excluding vs. Deleting App-V Package Shortcuts

To prevent a shortcut from being created in the App-V package, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the App-V package, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the App-V package and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project.

Renaming a Shortcut

Task ***To rename a shortcut:***

1. In the **Microsoft App-V Assistant**, open the **Applications** page.
2. Select the shortcut that you want to rename and click **Rename**.
3. Enter a new name for the shortcut.

Modifying App-V Package Registry Settings

Using the Microsoft App-V Assistant, you can view existing registry keys, values, and data, and add or delete registry items in your App-V package.

You can also set isolation options for selected registry keys. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

Information about modifying registry settings on the Registry page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- `HKEY_LOCAL_MACHINE`
- `HKEY_USERS`
- `HKEY_CURRENT_USER`
- `HKEY_CLASSES_ROOT`

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the Registry page is performed much like it is performed in the InstallShield Registry view. To learn more, see Editing the Registry.

Setting App-V Package Registry Isolation Options

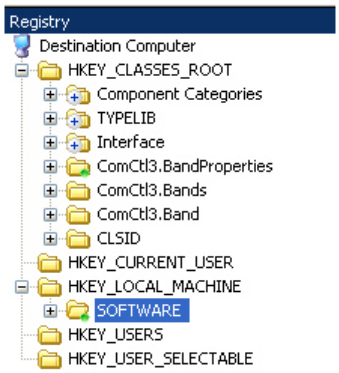
The Microsoft App-V Assistant enables you to configure isolation options for a registry key.

***Important** • Although you cannot explicitly set an isolation option for a registry value, registry values are subject to the isolation options of their keys.*

Task ***To configure an isolation option for a registry key:***

1. In the **Microsoft App-V Assistant**, open the **Registry** page.
2. Right-click the key that you would like to configure and then click **Isolation Options**. The **Options** dialog box opens.
3. Select one of the following options:
   - **Merge with local key**—The App-V package sees a merged view of the registry entries for the selected key from both the local registry and from the App-V package's registry.
   - **Override local key**—The App-V package sees only the registry entries for the selected key that are part of that App-V package.
4. Click **OK**.

Registry keys that have an override isolation option are marked with a special icon.



***Tip** • To launch the Registry Import Wizard and import an existing registry (.reg) file, click the Import a .reg file option in the More Options area on the Registry page.*

Inheritance of Isolation Options in the Registry

Isolation options for registry keys are always inherited. The isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

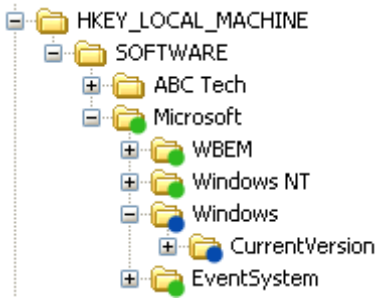


Figure 13-6: Example of Inheritance of Isolation Options from Folders to Files

Performing Dynamic Suite Composition

Version • *This information applies to App-V 4.x packages.*

A virtual package may rely on one or more other virtual packages in order to function properly. The Microsoft App-V Assistant lets you specify other App-V packages that the open App-V package (the primary package) requires. This capability, called *Dynamic Suite Composition*, enables your virtual package to interact with the other virtual applications in the virtual environment. Dynamic Suite Composition enables you to deploy common system components once on each client system, making them available for use by many App-V packages, rather than requiring you to include them with each of the App-V packages that are dependent on them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary virtual application.

To specify App-V packages that you want to include in a dynamic suite, use the Dynamic Suite Composition page of the Microsoft App-V Assistant.

Task ***To add one more dependencies to your App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Dynamic Suite Composition** page.
2. To add a dependency App-V package, click the **New** button. The **Open** dialog box opens.
3. Open the directory where the dependency App-V package that you want to add is located. That application's `.osd` and `.sft` files are listed.
4. Select one of the following:
   - **One of the .osd files**—If this dependency App-V package is or is going to be published on a server, select any one of the `.osd` files that are listed. If these `.osd` files were created properly, each of them should contain the information that will identify to the primary App-V package the published location of the dependency App-V package.

     Note • *It is not necessary to select more than one* `.osd` *file. All of them contain the same reference to the location of the dependency App-V package's* `.sft` *file, which is the only reference that is necessary in order for the primary App-V package to locate it.*

- **The .sft file**—If this dependency App-V package is or is going to be published locally on the client or at an accessible network location, you may select just the `.sft` file.

The selected App-V package is now listed in the **Dependency Applications** list and, by default, the **Mandatory** option is selected.

5. Set the status of the selected App-V package by doing one of the following:
   - **If the dependency App-V package is mandatory**—If the primary App-V package will not run unless it can locate this dependency App-V package, leave the **Mandatory** option selected. If a dependency App-V package that is configured as mandatory is not available, an error will be generated when someone attempts to run the primary App-V package.
   - **If the dependency App-V package is not mandatory**—If the primary App-V package will run even if it cannot locate this dependency App-V package, clear the **Mandatory** option.
6. Build the primary App-V package.

Deleting a Dependency Application from the List

To remove an App-V package from the Dependency Applications list, select the application and click the Delete button.

Modifying Build Options

On the Build Options page, specify for which releases of this InstallShield project you want to build an App-V package, and specify whether you want to include additional Windows Installer packages in the virtual package.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the Build App-V 5.x package check box or the Build App-V 4.x package check box on the Build Options page before you will be able to build an App-V package for that Windows Installer package.

Important • *You must create at least one release (in the Releases view of the Installation Designer) before you can select a release on the Build Options page.*

Selecting the Releases for Which You Want to Build App-V Packages

You select the releases that you want to build an App-V package for on the Releases tree of the Build Options page.

Important • *You cannot create or edit a release in the Microsoft App-V Assistant. If no releases exist, you can simply click the Build toolbar button to create a new release or open the Releases view of the InstallShield Installation Designer. You must create at least one release before you will be able to build an App-V package. For more information, see Creating and Building Releases.*

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the Releases tree on the Build Options page is not displayed.

Task ***To select releases to build:***

1. In the **Microsoft App-V Assistant**, open the **Build Options** page.
2. Select the releases in the **Releases** tree that you want to build an App-V package for.

Important • *When you select a release on the Build Options page, you are specifying that whenever you build that particular release, you want to also build an App-V package for that release. However, the releases that are selected on the Build Options page have no bearing upon which release is built when you click the Build button on the toolbar. When you initiate a build by clicking the Build button, a build is initiated for the active release—the release that was most recently selected on the Installation Designer Releases view. The output of that build would depend upon what releases were selected on the Build Options page:*

- **Active release selected**—A Windows Installer package and an App-V package would be built.
- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Enabling App-V Package Building When in Direct Edit Mode

When you are editing a Windows Installer (.msi) package or a transform (.mst) file in the Microsoft App-V Assistant, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting Save on the File menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's Build function is disabled.

However, you do need to run the build process to build an App-V package for this Windows Installer package. To do this, perform the following steps:

Task ***To enable App-V package building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. In the **Microsoft App-V Assistant**, open the **Build Options** page.
3. Select the **Build App-V 5.x package** check box or the **Build App-V 4.x package** check box.

The Build toolbar button is enabled.

Specifying Whether to Compress the Data Files in an App-V Package

Version • *This information applies to App-V 4.x packages.*

InstallShield lets you specify whether you want to use zlib compression for the data files in the App-V package.

Task ***To specify whether to compress the data files in the App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Build Options** page.
2. Specify the appropriate option for the **Would you like to compress the data in the virtual package?** option.

Including Additional Windows Installer Packages in an App-V Package

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the Microsoft App-V Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in an App-V package, set the **Would you like to include additional MSI files in the virtual package?** option on the Build Options page to Yes, and then select the packages that you want to add.

Task ***To include additional Windows installer packages in an App-V package:***

1. In the **Microsoft App-V Assistant**, open the **Build Options** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the **Open** button and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
   - Use the Delete button ( ) to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of an App-V Package

You can choose to build a Windows Installer package to assist in the distribution of an App-V package. This simplifies the deployment of an App-V package by enabling you to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management.

To build a Windows Installer package with your App-V package, select the **Generate an installation package as part of the build output** check box on the Advanced Settings dialog box. You can access this dialog box by clicking the Advanced Settings link in the More Options area on the Build Options page of the Microsoft App-V Assistant. By default, this check box is not selected.

When you run this Windows Installer file, the minimally required App-V package files will be “installed” in the local App-V client system cache. (The App-V package files remain on the App-V server until the client requests that they be downloaded when the application is launched for the first time.)

Note • *The App-V client must be installed on the local machine before you can install an App-V package. The installation will detect and warn if the App-V client is not available, and the installation will fail.*

To remove an installed App-V package, you need to use the Application Virtualization Client tool, which is available in the Administrative Tools of the Windows Control Panel.

Specifying Package Feature Block Optimizations

You can use the package optimization to control the performance and network traffic that is associated with running an App-V package. The package optimization support you select determines how quickly the App-V package launches, and how often additional functionality needs to be streamed to the client while the App-V package is being used.

The files in an App-V package can be grouped into two feature blocks:

- **Feature block 1**—Feature block 1 must contain the core functionality of the App-V package that is necessary to launch the application. At application launch, all of the files in feature block 1 are streamed to the client in one unit.
- **Feature block 2**—Feature block 2 can contain additional functionality of the App-V package that is not necessary to launch the application. While the App-V package is being used, the files in feature block 2 can be streamed in small packets on an as-needed basis.

You can either choose to include all App-V package files in feature block 1 or, to improve launch speed, you can choose to group the files into two feature blocks: feature block 1 and feature block 2.

You indicate your package optimization preference on the Package Optimizations dialog box, which is opened by clicking the Package Optimizations link in the More Options area on the Build Options page.

The Package Optimizations dialog box includes the following options:

Table 13-6 • Package Optimizations Options

Option	Description
Optimize for Streaming	If you choose this option, the Microsoft App-V Assistant will perform a static analysis of the shortcuts in the application and decide which files should be in feature block 1 and which should be in feature block 2. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent use of the application.*

Table 13-6 • Package Optimizations Options (cont.)

Option	Description
Optimize for Offline Use	If you choose this option, all files in the App-V package will be included in feature block 1 and will be streamed to the client at startup in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable end users to use the App-V package when not connected to the App-V server and if you want to eliminate network traffic when the App-V package is being used. ***Note*** *• When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*

Building an App-V Package

The method for App-V package depends upon what file you have open in InstallShield—an InstallShield project or a Windows Installer package.

Building an App-V Package from Within an InstallShield Project

To build an App-V package from within an InstallShield project, perform the following steps:

Task ***To build an App-V package for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

   Important *• You cannot create or edit a release in the Microsoft App-V Assistant. If no releases exist, or if you want to create a new release, open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build an App-V package. For more information, see Creating and Building Releases.*
3. In the **Microsoft App-V Assistant**, open the **Build Options** page.
4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build an App-V package for.
5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

Important *• When you select a release on the Build Options page, you are specifying that whenever you build that particular release, you want to also build an App-V package for that release. However, the releases that are selected on the Build Options page have no bearing upon which release is built when you click the Build button on the toolbar. When you initiate a build by*

clicking the Build button, a build is initiated for the active release—the release that was most recently selected on the Installation Designer Releases view. The output of that build would depend upon what was selected on the Build Options page:

- **Active release selected**—*A Windows Installer package and an App-V package would be built.*
- **Active release not selected**—*Only a Windows Installer package would be built.*

To learn how to build more than one release at a time, see Performing Batch Builds.

Building an App-V Package from Within a Windows Installer Package in InstallShield

To build an App-V package for a Windows Installer package, perform the following steps:

Task ***To build an App-V package for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).
   - On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
   - On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the Microsoft App-V Assistant to set App-V package options.
3. On the **File** menu, click **Save**.
4. In the **Microsoft App-V Assistant**, open the **Build Options** page.
5. Select the **Build App-V 5.x package** check box or the **Build App-V 4.x package** check box. The **Build Virtual Package** button is enabled.
6. Click the **Build Virtual Package** button.

Build Output for App-V Packages

Version • *Some functionality is specific to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

Location of the Build Output

If you build an App-V 5.x package, InstallShield saves the App-V output in the following directory path:

App-VPackage*ProductName*

If you build an App-V 4.x package, InstallShield saves the App-V output in the following directory path:

App-VPackage*ProductName*_v*N*

The version number (*N*) of the App-V 4.x package is appended to the end of that folder path. Each time that you build an upgrade, InstallShield creates a new subfolder and increments the version number in the name of the subfolder.

The default location of the App-VPackage folder depends on whether you are building an App-V 5.x or 4.x package from within an MSI Database project or a Basic MSI project:

- If you are building an App-V package from within an MSI Database project (.msi file), InstallShield creates the App-VPackage folder in the same folder as the .msi file.
- If you are building an App-V package from an .ism file, InstallShield creates the App-VPackage folder in the following location:

  *InstallShield Project Folder**ProjectName**Product Configuration**Release Name*\DiskImages\Disk1

Contents of the Build Output

If you are building an App-V package from within InstallShield, the output of the build consists of the following:

- An App-V package—For information on the files that are included in an App-V package, see Components of an App-V Package.
- A Windows Installer–based installation—Note that this is built only if you are building an App-V package from within a Basic MSI project. If you are building an App-V package from within an MSI Database project in InstallShield, InstallShield does not build this file.
- A new Basic MSI project that assists in the distribution of the App-V package—This is built if you choose to generate an installation package as part of the build process.

AdminStudio creates the new Basic MSI project in the release folder. Note the following details about this project:

- The project contains dynamic file links to the App-V package files.
- The properties and directories are updated in the project.
- The Releases view contains four or more releases that enable you to build different combinations of releases (for example, compressed or uncompressed, with or without an InstallShield prerequisite for the App-V client).
- A built release is also included with the new Basic MSI project. Any warnings or errors from this build are included as warning -9150 or error -9151 in the App-V project.
- You can use this project if you want to add more functionality to your Windows Installer package. For example, you can create major upgrades, digitally sign the package, and more.
- You can update the end-user license agreement (EULA) with your own custom EULA. The EULA that is included by default contains instructions that explain how to replace the default EULA text with your own EULA text.
- This project is overwritten each time that you rebuild your App-V project.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Building App-V Packages Through the Command Line

When you configure an App-V package in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the App-V package are built. When you use the standard InstallShield command-line build, you do not need to add any additional command-line parameters. All of the App-V package settings are saved within the InstallShield project.

Testing an App-V Package Using the App-V Launcher Tool

You can use the App-V Launcher tool to locally test a newly built App-V package before moving it to a deployment server.

To open the App-V Launcher tool, click **Test launch the App-V package** in the More Options area on the Build Options page. The App-V Launcher will attempt to launch that application. If there are multiple shortcuts in this App-V package, the Launch App-V Package dialog box opens, where you are prompted to select the shortcut that you want to launch from a list of all of the shortcuts.

Requirements for Using the App-V Launcher Tool

The machine on which you use the App-V Launcher to test an App-V package must meet the following requirements:

- The Microsoft Application Virtualization Client must be installed.
- The version of the Microsoft Application Virtualization Client that is present should be equal to or newer than the minimum client version of the App-V package.
- File streaming must be enabled because the App-V Launcher publishes the App-V package from a local file path. If file streaming is not enabled, the App-V Launcher displays an informative message asking if it can enable this functionality.

App-V Launcher Tool Location

When an App-V package is built, the App-V Launcher tool (`AppVLauncher.exe`) is placed in the same folder as the App-V package.

How the App-V Launcher Tool Works

In order to make it possible to test the App-V package without having to publish it on the server, a copy of the App-V Launcher tool is automatically copied to the output directory of each App-V package. The App-V Launcher looks for the .appv or .sft file that is located in the same directory.

Note *• The first time that you use the App-V Launcher to run an application in an App-V package, the entire package is published to that machine; this includes all of the package's shortcuts and file extension associations in the package. If you then use the App-V Launcher to run any application in the App-V package again, the App-V Launcher unpublishes the package (and its shortcuts and file extension associations) before republishing the package.*

Also note that the `AppVLauncher.exe` *file requires elevation. If you want to be able to test your App-V package in a locked-down environment where end users will not have elevated privileges, you may want to use the App-V Launcher once to launch and publish your App-V package with elevated privileges. Once you have done that, you can use the published shortcuts and file extension associations to start your application.*

The App-V Launcher is a convenient testing tool that makes it possible for you to reliably and accurately test your App-V packages on your local machine or any other system that has the App-V client installed before moving it to the App-V server.

Troubleshooting the Builds of App-V Packages

For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.

Application Features that Require Pre- or Post-Conversion Actions

Some application features are ignored when an App-V package is created. Therefore, some additional pre- or post-conversion actions must be taken in order for the App-V package to be created properly.

One action you could take to try to include ignored features in an App-V package is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to an App-V package.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

Microsoft App-V Assistant Reference

Reference information about the Microsoft App-V Assistant is organized into the following sections:

- Microsoft App-V Assistant Pages
- Microsoft App-V Assistant Dialog Boxes

Microsoft App-V Assistant Pages

The Microsoft App-V Assistant is comprised of the following pages:

- Microsoft App-V Assistant Home Page
- Package Information Page
- Files Page
- Applications Page
- Registry Page
- Dynamic Suite Composition Page
- Build Options Page

Microsoft App-V Assistant Home Page

The Microsoft App-V Assistant Home page displays a diagram that illustrates the process of creating an App-V package.

Click the following icons in the navigation bar at the bottom of the page to navigate through the Microsoft App-V Assistant interface:

Table 13-7 • Navigation Bar Icons

Icon	Destination
	Package Information Page
	Files Page
	Applications Page
	Registry Page
	Build Options Page
	Go to next page.
	Jump back to previous page.
	Microsoft App-V Assistant Home Page

Package Information Page

Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

On the Package Information page, specify the package name and enter a comment to document this virtual package. On this page, you can also choose to include diagnostic tools with the virtual package.

The Package Information page includes the following settings:

Table 13-8 • Package Information Page

Option	App-V Version	Description
Package Name	**App-V 4.x, App-V 5.x**	Enter a name for the virtual package. ***Tip*** • *If your virtual package contains multiple applications, you can specify the name that identifies the entire package. For example, Microsoft Office could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
Comments	**App-V 4.x, App-V 5.x**	Enter a short description of the App-V package. This setting is optional.
Does your App-V package have any specific operating system requirements?	**App-V 4.x, App-V 5.x**	Select one of the following: • **Yes**—The application does not support one of the listed operating systems. When you select this option, the check boxes become enabled, and you can clear the selection of the unsupported operating systems. • **No**—This application run on all of the listed operating systems. When this option is selected, the operating system check boxes are disabled and cannot be changed. Note that the list of operating systems that are displayed varies, depending on which version of App-V you are targeting.
Root Folder Name	**App-V 4.x**	Specify the root folder of the App-V package's file system. During run time, the Application Virtualization Client mounts the package's file system to the App-V virtual drive; the Q drive is the default. The long and short names of the root folder must be unique because two packages with the same root folder name cannot be deployed simultaneously. The default value for the Root Folder Name setting is based on the `[ProductName]` and `[ProductVersion]` properties of the App-V package's associated Windows Installer package using the 8.3 file naming convention. For example: • If `[ProductName]` is `MyApplication` and `[ProductVersion]` is `1.12.3.1`, the default folder name is `MyApplic.112`. • If `[ProductName]` is `MyApp` and `[ProductVersion]` is `1`, the default folder name is `MyApp.1`. • If `[ProductName]` is `MyBlueApp` and `[ProductVersion]` is `1.2.3.4`, the default folder name is `MyBlueAp.123`.

Table 13-8 • Package Information Page (cont.)

Option	App-V Version	Description
Protocol	**App-V 4.x**	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: • **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. • **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. • **FILE**—The App-V package are streamed from a file share. • **HTTP**—The hypertext transport protocol streams the App-V package. • **HTTPS**—The secure hypertext transport protocol streams the App-V package.
Host	**App-V 4.x**	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter **%SFT_SOFTGRIDSERVER%** to indicate an environment variable. ***Note*** *• If you enter* **%SFT_SOFTGRIDSERVER%***, you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Port	**App-V 4.x**	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.
Path	**App-V 4.x**	Specify the relative path on the virtual application server where the App-V package is stored. This is also the path from which the App-V package is streamed. If the App-V package is stored in a subdirectory of CONTENT, the path must be specified in this setting; otherwise, you can leave this setting blank.

Table 13-8 • Package Information Page (cont.)

Option	App-V Version	Description
Diagnostic Tools	**App-V 4.x**	For testing purposes, you can choose to include diagnostic tools in your App-V package by clicking the Diagnostic Tools link in the More Options area. For more information, see App-V Diagnostic Tools Dialog Box.
Upgrade Settings	**App-V 4.x, App-V 5.x**	To specify upgrade information for your App-V package, click the Upgrade Settings link in the More Options area. For more information, see App-V Package Upgrade Settings Dialog Box.

Files Page

Use the Files page of the Microsoft App-V Assistant to perform tasks such as the following:

- View the files and folders in the App-V package
- Add or remove files and folders in the App-V package
- Set isolation options

***Tip** • You can specify which of the Windows Installer predefined folders are listed in the Microsoft App-V package tree. To learn how, see Controlling the Display of Predefined Folders.*

Applications Page

Shortcuts provide the most visible entry points for launching the applications in the App-V package. Most App-V packages should have at least one shortcut.

On the Applications page of the Microsoft App-V Assistant, you can define application shortcuts to enable end users to launch an application in the App-V package. The Microsoft App-V Assistant creates shortcuts for any executable files that are added through the Files page. All shortcuts are added to the App-V package and published to the system when the package is published.

For more information, see the following:

- Creating a New App-V Package
- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Renaming a Shortcut

Registry Page

On the Registry page, you can view existing registry keys, values, and data, and add or delete registry items. You can also override the default isolation options for a registry key. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the Microsoft App-V Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected registry keys to exert control over application interactions with client operating system resources. For an overview of the available isolation options, and for instructions on how to set them, see Setting App-V Package Registry Isolation Options.

Registry items that are listed on this page will be included in the App-V package, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To launch the Registry Import Wizard and import an existing registry (.reg) file, click the Import a .reg file option in the More Options area on the Registry page.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the Registry page is performed much like it is performed in the InstallShield Registry view. To learn more, see Editing the Registry.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Dynamic Suite Composition Page

Version • *This information applies to App-V 4.x packages.*

A virtual package may rely on one or more other virtual packages in order to function properly. The Microsoft App-V Assistant lets you specify other App-V packages that the open App-V package (the primary package) requires. This capability, called *Dynamic Suite Composition*, enables your virtual package to interact with the other virtual applications in the virtual environment. Dynamic Suite Composition enables you to deploy common system components once on each client system, making them available for use by many App-V packages, rather than requiring you to include them with each of the App-V packages that are dependent on them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary virtual application.

To specify App-V packages that you want to include in a dynamic suite, use the Dynamic Suite Composition page of the Microsoft App-V Assistant.

The following settings are available on the Dynamic Suite Composition page.

Table 13-9 • Dynamic Suite Composition Page

Option	Description
Dependency App-V Packages	List of all of the dependency App-V packages that have been selected for the primary App-V package.

Table 13-9 • Dynamic Suite Composition Page (cont.)

Option	Description
Mandatory	Indicates whether to selected dependency App-V package is required in order for the primary App-V package to run. • **If the dependency App-V package is mandatory**—If the primary App-V package will not run unless it can locate this dependency App-V package, leave the Mandatory option selected. If a dependency App-V package that is configured as Mandatory is not available, an error will be generated when someone attempts to run the primary App-V package. • **If the dependency App-V package is not mandatory**—If the primary App-V package will run even if it cannot locate this dependency App-V package, clear the Mandatory option.
New Button	To add an App-V package to the Dependency App-V Packages list, click this button and select the App-V package (.osd, .sft) that you want to add. Select one of the following: • **One of the .osd files**—If this dependency App-V package is or is going to be published on a server, select any one of the .osd files that are listed. If these .osd files were created properly, each of them should contain the information that will identify to the primary App-V package the published location of the dependency App-V package. It is not necessary to select more than one .osd file. All of them contain the same reference to the location of the dependency App-V package's .sft file, which is the only reference that is necessary in order for the primary App-V package to locate it. • **The .sft file**—If this dependency App-V package is or is going to be published locally on the client or at an accessible network location, you may select just the `.sft` file. The selected reference App-V package is now listed in the Dependency Applications list and, by default, the Mandatory option is selected.
Delete Button	Click to delete the selected App-V package from the list.

Build Options Page

The Build Options page is where you configure various settings for the build of your virtual package. Some of the settings are available directly on the Build Options page. Some are available through dialog boxes that you can launch from links in the More Options area of the Build Options page.

Settings on the Build Options Page

The following settings are available on the Build Options page.

Table 13-10 • Settings on the Build Options Page

Setting	Description
Build App-V 4.x package or **Build App-V 5.x package**	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, it is not necessary to build the package, because it is already built. Therefore, InstallShield's Build function is disabled. Select the Build App-V 4.x package check box or the Build App-V 5.x package check box to enable the Build function. When this option is selected, the Build Virtual Package button is enabled. For more information, see Enabling App-V Package Building When in Direct Edit Mode.
Build Virtual Package	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, if you select the Build App-V 4.x package check box or the Build App-V 5.x package check box, this button is enabled. Click it to build the App-V package. ***Note*** • *This button will also be enabled if the **Build Citrix profile** option is selected on the **Build Settings** page of the Citrix Assistant. In this scenario, if you click this button without also selecting the **Build App-V package** option on this page, the App-V package will not be built.*
Would you like to compress the data in the virtual package?	***Version*** • *This setting applies to App-V 4.x packages.* To use zlib compression to compress the data in the App-V package, select Yes.
Would you like to include additional MSI files in the virtual package?	Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. To include additional Windows Installer packages in an App-V package, set this option to **Yes**, and then select the packages that you want to add. For more information, see Including Additional Windows Installer Packages in an App-V Package.
Select releases for which you want to build an App-V package	Specify for which releases you want to build App-V packages. For more information, see Selecting the Releases for Which You Want to Build App-V Packages. ***Note*** • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the Releases tree on the Build Options page is not displayed.*

Links in the More Options Area on the Build Options Page

The following links are available in the More Options area of the Build Options page.

Table 13-11 • More Options Links on the Build Options Page

Setting	Description
Open App-V package folder	To open the folder that contains the App-V package, click this link in the More Options box.
Package Optimizations	To configure package optimization, click this link in the More Options box. To learn more, see Specifying Package Feature Block Optimizations.
Advanced Settings	To generate a Windows Installer package that can assist in the distribution of the App-V Client and to optionally include the App-V Launcher tool for testing purposes, click this link in the More Options box.
Test launch the App-V package	To launch your App-V package for testing on your build machine, click this link in the More Options box. For more information, see Testing an App-V Package Using the App-V Launcher Tool.

Microsoft App-V Assistant Dialog Boxes

The Microsoft App-V Assistant includes the following dialog boxes:

- App-V Diagnostic Tools Dialog Box
- File Mapping Dialog Box
- Isolation Options Dialog Box (for a Package)
- Isolation Options Dialog Box (for Registry Keys)
- Launch App-V Package Dialog Box
- Options Dialog Box (for Configuring Isolation Options for a File)
- Options Dialog Box (for Configuring Isolation Options for a Folder)
- Package Optimizations Dialog Box

Advanced Settings Dialog Box

Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

The Advanced Settings dialog box opens when you click the Advanced Settings link in the More Options section on the Build Options tab. This dialog box is where you specify build and run-time options.

Table 13-12 • Advanced Settings Dialog Box Options

Option	App-V Version	Description
Generate an installation package as part of the build output	**App-V 4.x, App-V 5.x**	To build an installation package with your App-V package, select this check box. This check box is not selected by default. If you do select this check box, you can specify whether you want to load the installation package from the media location or from the shared location. Building an installation package enables you or your end users to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management to distribute your App-V package. When you run this Windows Installer file, the minimally required App-V package files will be "installed" in the local App-V client system cache. (The .sft file remains on the App-V server until the client requests that it be downloaded when the application is launched for the first time.) ***Note*** **•** *The App-V client must be installed on the local machine before you can install an App-V package. The installation will detect and warn if the App-V client is not available, and the installation will fail.* To remove an installed App-V package, you need to use the Application Virtualization Client tool, which is available in the Administrative Tools of the Windows Control Panel.
Load from Media Location	**App-V 4.x**	To load the installation package from the media location, select this option.
Compress	**App-V 4.x, App-V 5.x**	To build a compressed installation package, select this check box. If this check box is cleared, an uncompressed installation package is built.
App-V Client Prerequisite (Generates Setup.exe)	**App-V 4.x, App-V 5.x**	If you want to include the AdminStudio prerequisite that installs the App-V client on the target system, select this check box. Note that a Setup.exe setup launcher is required if the AdminStudio prerequisite needs to be included in the release.
Load from Shared Location	**App-V 4.x**	To load the installation package from the shared location, select this check box.
Include App-V Launcher Tool	**App-V 4.x, App-V 5.x**	To use the Include App-V Launcher tool to locally test a newly built App-V package before moving it to a deployment server, select this check box. For more information, see Testing an App-V Package Using the App-V Launcher Tool.

App-V Diagnostic Tools Dialog Box

Version • *The diagnostic tools are available for App-V 4.x packages. Starting with App-V 5.x, it is no longer necessary to inject diagnostic tool shortcuts directly into the package. The App-V Launcher tool is capable of launching a Command Prompt window within the virtual environment of an App-V 5.x package.*

On the Diagnostic Tools dialog box, which is opened by selecting Diagnostic Tools in the More Options area on the Package Information page, you can choose to include the Registry Editor and Windows Command Prompt diagnostic tools with your App-V package.

If you include diagnostic tools with your App-V package, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running an App-V package and you encountered an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

The Registry Editor diagnostic tool lets you use `Regedit.exe` on the local machine and have access to the virtual environment. The Command Prompt diagnostic tool lets you use `Cmd.exe` on the local machine and have access to the virtual environment.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the Registry Diagnostics or File System Diagnostics options on the Diagnostic Tools dialog box, shortcuts to those tools are automatically added to the App-V package.

When an end user runs this App-V package, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When an end user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

App-V Package Upgrade Settings Dialog Box

Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

The App-V Package Upgrade Settings dialog box is where you specify whether you want to create an upgrade for your App-V package. If you specify that you do want to create an upgrade, you can specify additional information about the upgrade.

Table 13-13 • App-V Package Upgrade Settings Dialog Box Settings

Setting	App-V Version	Description
Enable Upgrade	**App-V 4.x, App-V 5.x**	To create an upgrade for an earlier App-V package, select this check box. If you do not want to create an upgrade, clear this check box. When this check box is cleared, the other settings on the App-V Package Upgrade Settings dialog box are disabled. This check box is cleared by default.
Always upgrade latest built package	**App-V 4.x**	If you selected the Enable Upgrade check box and you want InstallShield to build an upgrade that updates the latest built App-V package, select this option.
Choose package to upgrade	**App-V 4.x, App-V 5.x**	If you selected the Enable Upgrade check box and you want InstallShield to build an upgrade that updates a particular App-V package, select this option, and then specify the path of the earlier package that you want to be updated.
Previous package to upgrade	**App-V 5.x**	If you selected the Enable Upgrade check box and you want InstallShield to build an upgrade that updates a particular App-V package, select this option, and then specify the path of the earlier package that you want to be updated.
Append version number to package name	**App-V 4.x, App-V 5.x**	If you selected the Enable Upgrade check box and you want InstallShield to append the version number to the App-V package name, select this check box.

File Mapping Dialog Box

The File Mapping dialog box lets you specify how you want to store the files in your App-V package. It also lets you indicate whether you want to allow write access to the virtual file system.

To launch the File Mapping dialog box, click the File Mapping link in the More Options area on the Files page. This dialog box displays the current primary application directory setting (if one has already been specified).

The following settings are available on the File Mapping dialog box.

Table 13-14 • File Mapping Dialog Box Settings

Setting	Description
Specify how files are stored in this package	Select the appropriate option: • **Map all files into the virtual file system (VFS)**—This option matches the behavior that was introduced in the Sequencer for App-V 5 SP3. This support is available for App-V 4.x and 5.x packages. • **Specify a primary application directory. Files and folders that are descendants of this directory will be mapped to the root folder**—To use a primary application directory, select this option and then optionally specify the application directory that contains most of the files in the App-V packages. To specify the application directory, click the ellipsis button (...). When you do so, the Browse for Directory dialog box opens, listing all of the currently available destination directories for this App-V package. If you leave the application directory field blank, the directory is determined automatically at run time. To learn how, see Specifying the Primary Application Directory. At run time when the App-V package is loaded, the directory and its contents are mounted to the App-V virtual drive. This is the default option.
Allow full write permission to the VFS	If you want an App-V 5.x package that you are creating to have full write permissions to the virtual file system (VFS), select this check box. Selecting this check box may be useful for sequencing third-party applications.

Isolation Options Dialog Box (for a Package)

Use the Isolation Options dialog box to configure advanced settings for COM isolation and named object isolation. This support is available for App-V 5.x packages.

To access the Isolation Options dialog box, click the Isolation Settings link in the More Options area on the Package Information page.

The following settings are available on the Isolation Options dialog box.

Table 13-15 • Isolation Options Dialog Box

Setting	Description
COM isolation	Select the appropriate option: • Isolate COM objects from the local system • Allow COM objects to interact with the local system

Table 13-15 • Isolation Options Dialog Box (cont.)

Setting	Description
Named object isolation	Select the appropriate option: • Isolate named objects from the local system • Allow all named objects to interact with the local system

Isolation Options Dialog Box (for Registry Keys)

Use the Isolation Options dialog box to specify whether the App-V package should see only the registry entries for the selected key that are part of that App-V package, or see a merged view of the registry entries for the selected key from both the local registry and from the App-V package's registry.

To open the Isolation Options dialog box, right-click a registry key on the Registry page and then click Isolation Options.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

Use the Isolation Options dialog box to select one of the following options:

Table 13-16 • Options on the Isolation Option Dialog Box

Option	Description
Merge with local key	The App-V package sees a merged view of the registry entries for the selected key from both the local registry and from the App-V package's registry.
Override local key	The App-V package sees only the registry entries for the selected key that are part of that App-V package.

Launch App-V Package Dialog Box

If you use the InstallShield App-V Launcher to locally test a newly built App-V package before moving it to a deployment server, the Launch App-V Package dialog box opens when an App-V package has more than one shortcut. You are prompted to select the shortcut that you want to launch from a list of all of the shortcuts.

To open the App-V Launcher, click the **Test launch the App-V package** link in the More Options area on the Build Options page.

Options Dialog Box (for Configuring Isolation Options for a File)

Version • *This information applies to App-V 4.x packages.*

Use the Options dialog box to configure isolation options for the selected file.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, App-V, and registry settings.*

The following setting is available on the Options dialog box for a file that is selected on the Files page:

Table 13-17 • Setting on the Options Dialog Box for a File

Setting	Description
File Type	Specify the data type of the file. Available options are: • **Application Data**—Changes to the file are saved for all users of the App-V package on the client system. • **User Data**—Changes to the file are saved only for the logged-on user.

Options Dialog Box (for Configuring Isolation Options for a Folder)

Version • *Some settings apply to particular versions of App-V packages. Version-specific differences are noted where appropriate.*

Use the Options dialog box to configure isolation options for the selected folder.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, App-V, and registry settings.*

The following settings are available on the Options dialog box for a folder that is selected on the Files page:

Table 13-18 • Settings on the Options Dialog Box for a Folder

Setting	Description
Isolation	Specify whether you want to use isolation for the selected folder and all of its contents. Available options are: • **Merge with local directory**—The virtual application sees a combined view for the selected directory and its contents; it consists of the folder and its contents in the virtual application's directory and the folder and its contents that are on the local system. • **Override local directory**—The virtual application sees only the folders and their contents that are in the App-V package. The folders in the App-V package are isolated from the local directory.

Table 13-18 • Settings on the Options Dialog Box for a Folder (cont.)

Setting	Description
File Type	 ***Version*** • *This setting is available for App-V 4.x packages.* Specify the data type of the folder. Available options are: • **Application Data**—Changes to the folder are saved for all users of the App-V package on the client system. • **User Data**—Changes to the folder are saved only for the logged-on user.

Package Optimizations Dialog Box

The Package Optimizations dialog box enables you to specify your preference for control of performance and network traffic that is associated with running an App-V package. The package optimization option that you select determines how quickly the App-V package launches, and how often additional functionality needs to be streamed to the client while the App-V package is being used.

The Package Optimizations dialog box opens when you click the Package Optimizations link in the More Options area on the Build Options page.

Use the Package Optimizations dialog box to select one of the following options:

Table 13-19 • Package Optimizations Options

Option	Description
Optimize for Streaming	the Microsoft App-V Assistant performs a static analysis of the shortcuts in the application and decides which files should be in feature block 1 and which should be in feature block 2. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent use of the application.*

Table 13-19 • Package Optimizations Options (cont.)

Option	Description
Optimize for Offline Use	All files in the App-V package are included in feature block 1. All of the files are streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. If you want to enable end users to use the App-V package when the target system is not connected to the App-V server, and if you want to eliminate network traffic when the App-V package is being used, select this option. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*

For more information, see Specifying Package Feature Block Optimizations.

Advanced Table Settings for Conversion to Microsoft App-V

You can customize your virtual conversion process in the following ways:

- **Per package**—You can use InstallShield to directly edit the `ISVirtualPackage` table to modify the settings referenced below. You can also use the App-V Assistant user interface to modify the settings.
- **Globally for any conversion**—You can edit the `Settings.xml` file to specify default values for many of the settings that can be specified in the `ISVirtualPackage` table.

In addition to editing the `ISVirtualPackage` table, you can also edit other tables (directory, file, registry, shortcut) that store App-V conversion settings related to a particular item in the package, such as a particular shortcut, file, registry entry, or directory.

Information about customizing table settings for your App-V conversion process are organized into the following sections:

- ISVirtualPackage Table
- ISVirtualRelease Table
- ISVirtualDirectory Table
- ISVirtualFile Table
- ISVirtualRegistry Table
- ISVirtualShortcut Table
- Miscellaneous Virtual Conversion Settings
- Editing the Settings.xml File

Note • *If you want to modify the setting in the* `ISVirtualPackage` *table globally, you can edit the* `Settings.xml` *file, as described in Editing the Settings.xml File. However, the settings in the* `ISVirtualDirectory`, `ISVirtualFile`, `ISVirtualRegistry`, *and* `ISVirtualShortcut` *tables cannot be specified in the* `Settings.xml` *file.*

ISVirtualPackage Table

The `ISVirtualPackage` table is the main table that stores package-wide conversion settings. To edit this table, open the package in InstallShield and open the Direct Editor view. Also, if you make selections in the InstallShield Assistants, it will modify the settings in this table.

The following are App-V settings in the ISVirtualPackage table.

Table 13-20 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
`AppVComments`			SFT file comments
`AppVDiagFileSystem`	1	0	Include File System Diagnostic tool - a shortcut is included to run cmd.exe from the physical System32 folder. This cmd.exe and any programs launched from it will have access to the virtual environment of the package
`AppVDiagRegistry`	1	0	Include Registry System Diagnostic tool - a shortcut is included to run regedit.exe from the physical Windows folder. It will have access to the virtual environment.
`AppVDSC0, AppVDSC1, etc.`	Absolute path to OSD or SFT file [: MANDATORY]		Dynamic Suite Composition settings
`AppVFullVFSWriteMode`	1	0	Set to enable full VFS Write mode. ***Note*** • *This setting only applies for App-V 5.x packages.*
`APPVLOADING`	1	0	Set this option to not include the SFT file in the wrapper MSI. The SFT file will be streamed from the server location specified in the OSD and manifest files.
`AppVMsiWrapperCompress`	1	0	Compression setting for wrapper MSI

Table 13-20 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
AppVName		Same as name of MSI	Specify package name
AppVNoCompression	1	0	Compression setting - default is compressed
AppVNoSpacesInFileNames	1		Will replace spaces in the SFT, OSD, and Icon file names with '_'.
AppVOS	Bitwise or of flags representing OS	0	0 indicates OS independent. Otherwise, here is the OS list starting with bit 1: WinXP, WinXP64, Win2003Svr, Win2003TS, Win2003TS64, Win2008Svr, Win2008TS, Win2008TS64, WinVista, WinVista64, Win7, Win764, Win2008R2TS64
AppVPackageOptimization	Offline or Stream	Stream	Only the shortcut targets are put in feature block 1 (FB1) if Stream is selected. Otherwise the entire package is put in FB1.
AppVPrereq	1	0	Set this option to include App-V client setup as a setup prerequisite for the wrapper MSI. It will be necessary to obtain a redistributable copy of the App-V client setup to use this feature.
AppVRootFolderName		8.3 name based on product name and version	Specify root folder name
AppVRuntimeDrive	Drive letter such as M:	Q:	App-V client drive to use
AppVServerURLHost			Server location of SFT file
AppVServerURLPort			Server location of SFT file
AppVServerURLProtocol	RTSP, RTSPS, FILE, HTTP, or HTTPS		Protocol to use to access SFT file location

Table 13-20 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
AppVSpaceReplacementString	Some string		Use together with setting AppVNoSpacesInFileNames property to 1. Any spaces in SFT, OSD, and Icon file names will be replaced by the string specified in the value of this property. If the string 'EMPTYSTRING' is used, then spaces will just be removed.
AppVTestLauncher	1	1	AppVLauncher.exe is copied next to the newly built App-V package. This tool can be used to easily test deploy App-V packages.
AppVUpgrade	1	0	Enables creation of an upgrade package
AppVUpgradeAppendPackageVersion	1	1	Package version will be appended to the end of the SFT file name
AppVUpgradeLatest	1	0	Will locate the most recently built App-V package based on modified timestamp on SFT files found in appropriately named sub-folders next to the MSI file.
AppVUpgradePreviousPackage			Absolute path to SFT from previous package that will be upgraded.
AppVv5ComInprocess	1	0	Set to 1 to enable in-process COM interaction. Com Interaction has to also be enabled for this option to have effect. ***Note*** • *This setting only applies for App-V 5.x packages.*
AppVv5ComInteraction	1	0	Set to 1 to enable COM interaction. ***Note*** • *This setting only applies for App-V 5.x packages.*

Table 13-20 • App-V Settings in ISVirtualPackage Table

Name	Value	Default	Meaning
AppVv5ComOutofprocess	1	1	Set to 1 to enable out-of-process COM interaction. Com Interaction has to also be enabled for this option to have effect. ***Note*** *• This setting only applies for App-V 5.x packages.*
AppVv5EnableBrowserHelperObjects	0	1	Set to 0 to disable browser helper objects. ***Note*** *• This setting only applies for App-V 5.x packages.*
AppVv5NamedObjectsInteraction	1	0	Set to 1 to enable named objects interaction. ***Note*** *• This setting only applies for App-V 5.x packages.*
BuildMSI	1	0	Create a wrapper MSI file that can be used to deploy the App-V package
MSIFile0, MSIFile1, etc	Absolute path to MSI		Indicates other MSI packages to suite together with the current one into one package.
VirtualPackageBuildOutputFolder	Absolute path to a directory		Instead of creating the converted virtual applications in a folder next to the source MSI, put them in a new folder under this specified location - this overrides the global redirect option in settings.xml.

Note *• If you want to modify these settings globally, you need to edit the* `Settings.xml` *file, as described in* *Editing the Settings.xml File**.*

ISVirtualRelease Table

The `ISVirtualRelease` table stores the relationship between InstallShield project releases and the virtual package type you want to build. This table is only relevant when you are editing an InstallShield Basic MSI project (not when you are editing an MSI package in the DirectEdit mode). If you make the relevant selections in the Assistants, it will modify the settings in this table.

Note • *The settings in this table cannot be specified in the Settings.xml file.*

Table 13-21 • General Settings in ISVirtualRelease Table

ISRelease_	ISProductConfiguration_	Name	Value	Meaning
`Key to ISRelease`	Key to `ISProductConfiguration`	`BuildVirtualPackage`	1	Build virtual package when associated release is built
`Key to ISRelease`	Key to `ISProductConfiguration`	`Provider`	Semicolon separated list of Thinstall, AppV, and Citrix	Indicates virtual technologies to which to convert MSI packages

ISVirtualDirectory Table

The following are App-V settings in the ISVirtualDirectory table.

Table 13-22 • App-V Settings in ISVirtualDirectory Table

Directory_	Name	Value	Meaning
Key into Directory table	AppVUserData	1	If set, then treat this directory as user data. If unspecified, then default algorithm is used to determine whether to mark directory as user data or application data.
Key into Directory table	AppVOverride	1	Override directory contents during upgrade

ISVirtualFile Table

The following are App-V settings in the ISVirtualFile table.

Table 13-23 • App-V Settings ISVirtualFile Table

File_	Name	Value	Meaning
Key into File table	AppVUserData	1	If set, then treat this file as user data. If unspecified, then default algorithm is used to determine whether to mark file as user data or application data.
Key into File table	AppVOverride	1	Override file during upgrade

ISVirtualRegistry Table

The following are App-V settings in the ISVirtualRegistry table.

Table 13-24 • App-V Settings in ISVirtualRegistry Table

Registry_	Name	Value	Meaning
Key into Registry table	AppVOverride	1	If set, virtual application will only see the registry key contents in the virtual package and no child keys that may be present on the physical machine. Otherwise, virtual application will see only values in the virtual package, but will see child keys present on the physical machine, if they are not also present in the virtual package.

ISVirtualShortcut Table

The following are App-V settings in the ISVirtualShortcut table.

Table 13-25 • App-V Settings in ISVirtualShortcut Table

Shortcut_	Name	Value	Meaning
Key into Shortcut table	AppVApplication	0	A value of zero indicates that this shortcut will not be included in the converted App-V package.

Manually Adding an Entry to the ISVirtualShortcut Table

Typically the target version in an OSD file is automatically determined during conversion to App-V 4.x package format. The version of the shortcut target file is used, or if the target file does not have a version, then a default value of '1.0' is used. To set a custom version, you can manually add an entry to the ISVirtualShortcut table.

Task ***To manually populate the ISVirtualShortcut table using the InstallShield Editor:***

1. Open the **Direct Editor** view.
2. Select the **ISVirtualShortcut** table and click **New** to add a new record.
3. Enter the following values:
   - For **Shortcut**, enter the key of the shortcut.
   - For **Name**, enter the property name which is **AppVTargetVersion**.
   - For **Value**, enter the desired version number.
4. Click **OK**.

Note • *This setting only has effect for App-V 4.x conversion.*

Miscellaneous Virtual Conversion Settings

You can edit the following XML file to modify global settings that also govern the creation of virtual packages.

Table 13-26 • Miscellaneous Settings

Location	Name	Value	Meaning
`System\Msi.xml`	`IgnoreTables`	MSI table names	Control whether an error or warning is flagged for certain tables during conversion
`System\Msi.xml`	`IgnoreCustomActions`	MSI custom action names	List of custom actions that can safely be ignored during virtual conversion
`System\Msi.xml`	`PropertyDefaults`	MSI property names with given values	Default values to use for certain MSI properties rather than flagging them as warnings
`Support\0409\settings.xml`	`GlobalBuildRedirectFolder`	Absolute directory path	Instead of creating the converted virtual applications in a folder next to the source MSI, put them in a new folder under this specified location

Editing the Settings.xml File

To edit the `Settings.xml` file, add a property element for each setting in the `Virtualization/Properties` section of the file. You can find the `Settings.xml` file in the following directory:

```
[InstallShield Installation Directory]\Support\0409
```

Edit the following section of the file:

```
<Virtualization>
...
    <Properties>
        <!--Use this section to provide a global default for any setting
        that is found in the ISVirtualPackage table-->
        <!--<Property Name="AppVRuntimeDrive" Value="G:"/>-->
        <!--<Property Name="AppVServerURLPath" Value="%PackageName%_v%PackageVersion%" />-->
    </Properties>
</Virtualization>
```

Creating Citrix Profiles

You can use the Citrix Assistant to help you author a Citrix profile for an application. The Citrix profile can then be deployed on a Citrix XenApp. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine. Using the Citrix Assistant, you can configure a Citrix profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

Information about creating Citrix profiles using the InstallShield Citrix Assistant is organized into the following sections:

- Overview of the Citrix Assistant
- Using the Microsoft App-V Assistant to Create an App-V Package
- Microsoft App-V Assistant Reference

Overview of the Citrix Assistant

You can use the Citrix Assistant to help you author a Citrix profile for an application. The Citrix profile can then be deployed on a Citrix XenApp. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine. Using the Citrix Assistant, you can configure an application's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

The process for authoring a Citrix profile using the Citrix Assistant is as follows:

Table 13-27 • Steps to Convert a Windows Installer Package to a Citrix Profile

Step	Go To:	Actions
Getting Started	**InstallShield Start Page**	Create or open one of the following project types: • Basic MSI • MSI Database (Direct Edit Mode) • Transform (Direct MST Mode)
	Citrix Assistant Home Page	Click on the **Citrix XenApp** tab to open the **Citrix Assistant Home** page
Specifying Package Information and Deployment Options	**Profile Information Page**	Specify the name and version of the Citrix profile, whether this package can run executables that are not included with the Citrix profile, and whether to include diagnostic tools with the Citrix profile.
Specifying Operating System and Language Requirements	**Profile Requirements Page**	Specify the operating systems and language requirements that client workstations must meet in order for this application to operate properly. You can also specify pre-launch and post-exit scripts to execute.
Managing Files in an App-V Package	**Profile Files Page**	View existing files and folders, add and delete files.
Setting Isolation Options	**Profile Files Page**	Override the Citrix default isolation options for selected folders and files. Isolation options specify how the virtual environment will provide access to files and folders requested by the Citrix profile.
Modifying Shortcuts to the App-V Package's Executable Files	**Profile Shortcuts Page**	Create, delete, include, exclude, or rename a Citrix profile's executables, which are derived from the shortcuts in its Windows Installer package.
Modifying App-V Package Registry Settings	**Profile Registry Page**	Add, delete, or modify the registry settings in your Citrix profile, and override the Citrix default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the Citrix profile.

Table 13-27 • Steps to Convert a Windows Installer Package to a Citrix Profile

Step	Go To:	Actions
Modifying Build Options	**Build Settings Page**	Choose whether to digitally sign the Citrix profile and select the releases that you want to build. Also, when you have a Windows Installer package open in Direct Edit mode, you can enable the **Build Release** option on the **Build** menu by making a selection on this page.
Building an App-V Package	**Build on the Toolbar** OR **Build Citrix Profile (F7) on the Build Menu**	Click **Build** to build the active Release and create a Citrix profile. Also, when you have a Windows Installer package open in Direct Edit mode, you can enable the **Build Release** option on the **Build** menu by selecting the **Build Citrix Profile** option on this page.

Information about the Citrix XenApp and Citrix profiles is presented in the following topics:

- About Citrix XenApp
- Components of an App-V Package
- Benefits of Deploying Citrix Profiles
- Supported InstallShield Project Types
- How Transforms are Included in an App-V Package

About Citrix XenApp

Citrix XenApp is an application delivery system for Windows applications that offers both application virtualization and application streaming. Applications are centralized on the Citrix XenApp, and then those applications are deployed to users throughout the enterprise. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine.

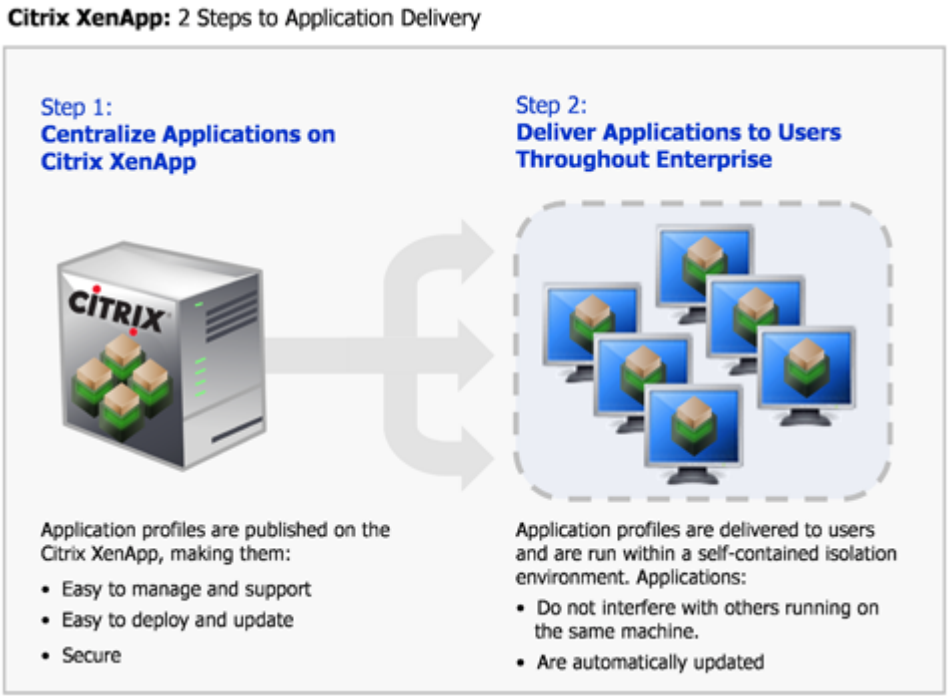


Figure 13-7: Citrix XenApp: Two Steps to Application Delivery

When applications are deployed on a Citrix XenApp, users can run those applications in an isolation environment, without installing, while connected or offline. Applications behave just like they were installed locally, but without any of the problems of installation, such as interfering with other applications on the same device. Files are saved locally and individual settings are preserved. Every time the application is run, it checks for errors or updates and they are delivered automatically.

Note • *For more information, see Benefits of Deploying Citrix Profiles.*

About the Citrix Assistant

You can use the Citrix Assistant to prepare a Windows Installer package for deployment on Citrix XenApp by converting it to a Citrix profile. During this process, you:

- **Profile Information**—Specify profile information.
- **OS and Language Requirements**—Specify the operating system and language requirements for the application.
- **Files, Folders, Shortcuts, Registry Settings**—Specify files, folders, shortcuts, and registry settings included in application.
- **Isolation Options**—Define a set of options for running the application in isolation on the user desktop.
- **Build**—Specify build settings and build a Citrix profile.

The following diagram illustrates the Citrix profile creation process:

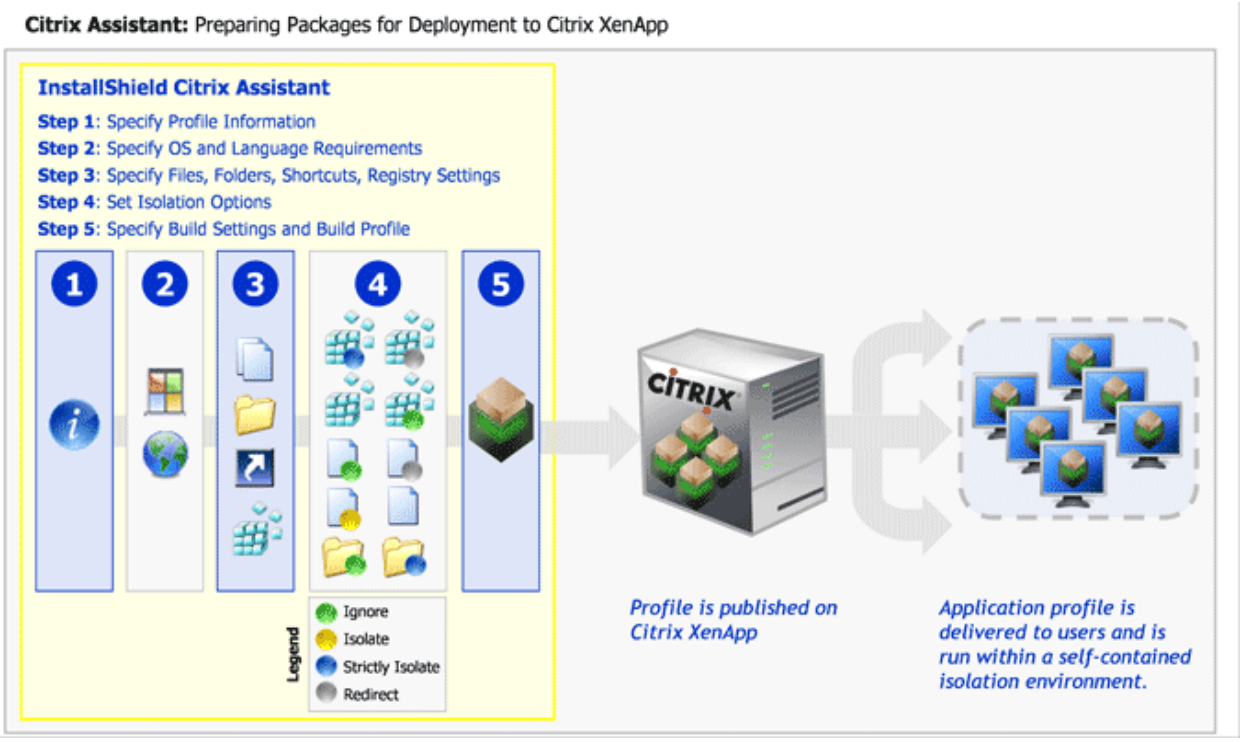


Figure 13-8: Preparing Packages for Deployment on Citrix XenApp

About Citrix Profiles

When you use the Citrix Assistant to prepare a Windows Installer package for deployment on the Citrix XenApp, the resources you generate are called *profiles*. A profile consists of the following files and directories:

Table 13-28 • Components of an Citrix Profile

Component	Name	Description
Profile Manifest File	`myapp.profile`	An XML file that defines the profile.
CAB File	`[alphanumeric_string].cab`	Compressed cabinet file that provides the isolation environment contents for the application.
Hashes File	`Hashes.txt`	Hash key file for digital signatures and signing profiles.
Icons File	`Icons.bin`	Icons repository.
Scripts Folder	`Scripts`	Folder containing any pre- launch or post-exit scripts that you have chosen to include.

Caution • *Modifying these files directly is* **not recommended**. *To make any modifications, use the Citrix Assistant.*

These files are saved in a directory named `CitrixProfile`. The location of the `CitrixProfile` directory depends upon the type of file you are editing in InstallShield:

- **InstallShield project**—The `CitrixProfile` directory will be located in a subdirectory of the directory that contains this InstallShield project file, such as:

  ```
  C:\InstallShield 2008 Projects\ProductName\ConfigurationName\ReleaseName\CitrixProfile
  ```

- **Windows Installer package**—The `CitrixProfile` directory will be located in the same directory as the Windows Installer file, such as:

  ```
  C:\FolderContainingMSI\CitrixProfile\ProductName
  ```

The contents of the application profile are published on the Citrix XenApp.

A profile can contain a single application or suite of applications. For example, you can profile Microsoft Word by itself, or you can profile the entire Microsoft Office suite in a single profile.

Benefits of Deploying Citrix Profiles

Converting a Windows Installer package to a Citrix profile and deploying it on a Citrix XenApp offers the following benefits:

- Reduces Application Conflicts
- Enables Rapid, Low-Cost Application Deployment
- Enables Automatic Software Updates
- Centralized Application Management Provides Controlled Access and Security
- Enables User-Based Application Access Rather Than Machine-Based Access

Reduces Application Conflicts

Traditionally to deploy an application throughout an enterprise, the application was installed on each user's desktop. Therefore, prior to installation, each application had to be tested for conflicts against each target desktop image (operating system with existing applications). After resolving conflicts that were found during testing, each application then had to be installed on each desktop. This process was very time consuming not only during initial installation, but also when applying patches or upgrading.

Citrix profiles run within isolation environments, which separate the interaction between an application and the underlying operating system's resources in order to prevent the applications from interfering with others running on the same machine. Because applications do not interact, the need to perform any conflict analysis and regression testing prior to deployment is eliminated. This not only results in rapid application deployment, but it also reduces the total cost of application delivery, due to decreased labor by IT.

Also, because users running applications in an isolation environment encounter no conflicts with other applications, user calls to the help desk are decreased.

Enables Rapid, Low-Cost Application Deployment

Deploying Citrix profiles on Citrix XenApp simplifies the deployment of new applications, updates, and patch deployment, regardless of the diversity of the access devices, software languages, computing architectures, and networks that are involved.

- **Only a single instance of the application is installed**—Instead of deploying, managing, updating, and securing a vast array of heterogeneous client software on each individual user's access device, a single instance of the application is installed on the Citrix XenApp. The IT department needs to test for only one environment, and deploy and update in one place. This reduces the cost of application installation and support. Also, you can deploy a Citrix profile once on a Citrix XenApp and replicate it to other Citrix XenApps within the existing enterprise infrastructure.
- **Prevents application-specific server silos**—Deploying applications on Citrix XenApp prevents the build-up of application-specific server silos because you can safely install and reliably run multiple application versions and incompatible applications on the same server.
- **Enables you to quickly install and update software throughout your enterprise**—Because your IT staff can manage the delivery of all of your Windows-based applications from one centralized location, your IT staff does not need to go from desktop to desktop, traveling to each office, in order to install or update software. With Citrix XenApp, you can deliver applications and updates instantly anywhere, any time.

Enables Automatic Software Updates

When an upgrade or patch needs to be deployed, you would only need to update the Citrix profile on the Citrix XenApp, which will then automatically update all of the instances of that Citrix profile throughout the enterprise. This means that users always have the latest application updates and patches, automatically.

Centralized Application Management Provides Controlled Access and Security

With Citrix XenApp, you can centralize applications and data in secure data centers, which increases data security and ensures fast, reliable performance. Centralized application management using Citrix XenApp provides the following benefits:

- **Enhances security**—Enables you to control, protect, and retain intellectual property centrally to reduce the chance for data loss and theft. Citrix XenApp helps you prevent data from leaving the data center without your explicit permission, which supports regulatory compliance and security objectives. You can provide authorized access to appropriate users—such as employees, customers, and partners—while verifying the ongoing security of the environment.
- **Can provide managed access to applications to users outside of your organization**—You can standardize the use of applications, without having to standardize the machines that the applications use. This enables you to provide managed access to applications from computers that are not your own corporate assets, such as from contractor or consultant computers.
- **Monitors application usage and performance**—Citrix XenApp gives you end-to-end visibility into application usage and performance. It gives IT administrators the power to understand who is using what, how often, and to what extent. They can observe, monitor, measure, audit, report and archive all the dimensions of information flow throughout the computing environment. This enables informed decisions regarding application consolidation and retirement, capacity planning, service level agreements and departmental charge-back
- **Enables identity-driven access**—Citrix XenApp enables you to provide identity-driven access tailored to any user environment. It automatically analyzes the user's permissions and then delivers the appropriate level of access to applications without compromising security. Depending on who and where users are and what device and network they're using, they may be granted different levels of access. You can also easily "decommission" applications by simply turning off a user's permission to it.

Enables User-Based Application Access Rather Than Machine-Based Access

Users can access their applications anywhere on the network, regardless of where they are or what device they are using.

Supported InstallShield Project Types

The **Citrix XenApp** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in a Citrix Profile

The Citrix Assistant supports the inclusion of transform files with Windows Installer packages in a Citrix profile.

- **How transforms are applied during profile generation**—When building a Citrix profile, transforms that you have specified are automatically applied to the base Windows Installer (`.msi`) package to create a temporary package, and then the Citrix profile is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build a profile from that transform file. When you create a new transform file in InstallShield, you specify the root `.msi` file in the **Open Transform** wizard. The steps you take to generate a profile after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the Citrix profile, you need to open one of the *transforms* in InstallShield (rather than the `.msi` file). The **Open Transform** wizard will open, and you will be prompted to specify the root `.msi` file and which of the existing `.mst` files you want to include. The steps you take to generate a profile after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to a Citrix profile must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the profile is built.*

Using the Citrix Assistant to Create a Citrix Profile

The steps you need to take to create a Citrix profile are the following:

Table 13-29 • Steps to Take to Create a Citrix Profile Using the Citrix Assistant

Step #	Description
Step 1	Specifying Package Information and Deployment Options
Step 2	Specifying Operating System and Language Requirements
Step 3	Managing Files in an App-V Package
Step 4	Setting Isolation Options
Step 5	Modifying Shortcuts to the App-V Package's Executable Files

Table 13-29 • Steps to Take to Create a Citrix Profile Using the Citrix Assistant

Step #	Description
Step 6	Modifying App-V Package Registry Settings
Step 7	Modifying Build Options
Step 8	Building an App-V Package

Specifying Citrix Profile Information

When creating a Citrix profile, you need to specify the **Name**, **Description**, and **Version** of the Citrix profile. You also need to specify whether this package can run executables that are not included with the Citrix profile, and whether to include diagnostic tools with the Citrix profile. The following tasks are performed on the **Profile Information** page of the **Citrix Assistant**:

- Specifying the Profile Name, Description, and Version
- Specifying Whether Users Should Be Able to Update Applications
- Including Diagnostic Tools in an App-V Package

Specifying the Profile Name, Description, and Version

On the **Profile Information** page of the Citrix Assistant, you name the Citrix profile, and provide a description and version number.

Task ***To specify the Citrix profile name, description, and version:***

1. In the **Citrix Assistant**, open the **Profile Information** page.
2. In the **Name** field, enter a name for this Citrix profile. The name you enter here determines the file name of the generated Citrix profile.

   Tip • *Do not include the version number in the profile name.*
3. In the **Description** field, enter a brief explanation of the purpose of this package. This information is stored as package metadata.
4. In the **Version** field, enter the version number of this Citrix profile. This information is stored as package metadata.
5. On the **File** menu, click **Save** to save your changes.

Specifying Whether Users Should Be Able to Update Applications

The **Profile Information** page is where you specify whether users can update applications.

Task ***To specify whether users should be able to update applications:***

1. In the **Citrix Assistant**, open the **Profile Information** page.
2. Select or clear the **Enable User Updates (Allow profiled application to update itself—Not recommended)** check box:
   - To allow the profiled application to download and install vendor-supplied updates over the Internet, select this check box. The updates are stored within the user profile root location for the specific user.
   - To ensure that all executable files from the profile are launched from the installation root location, and not from the user profile location, clear this check box. When this check box is cleared, the system prevents code from being run if it is not streamed from the server. Clearing this check box enables you to control updates through the profiler.

   This check box is cleared by default.
3. On the **File** menu, click **Save** to save your changes.

Including Diagnostic Tools With a Citrix Profile

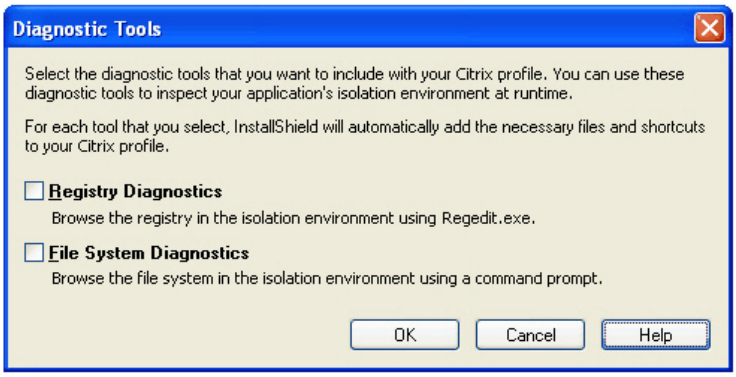
On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Profile Information** page, you can choose to include the Registry Editor and the Windows Command Prompt diagnostic tools with your Citrix profile.

If you include diagnostic tools with your Citrix profile, you will be able to look at the registry or file system for the application while it is running in its isolation environment. For example, if you were running a Citrix profile and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the Citrix profile. However, these tools may not be compatible with other operating systems.*

Task ***To include diagnostic tools with a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Information** page.
2. In the **More Options** list, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.



3. If you want to include the Registry Editor (regedit.exe) with your Citrix profile so that you can browse the profile registry at runtime from within the isolation environment, select the **Registry Diagnostics** option.
4. If you want to include the Windows Command Prompt application with your Citrix profile so that you can browse the virtual file system at runtime from within the isolation environment, select the **File System Diagnostics** option.

Launching the Diagnostic Tools Within the Isolation Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the profile.

When the user runs this Citrix profile application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's Citrix isolation environment.

Specifying Operating System and Language Requirements

The next step in creating a Citrix profile is to open the **Profile Requirements** page and specify the operating system and language requirements that client workstations need in order to run the application locally.

Some applications can run on multiple operating systems and languages, while others, such as custom applications, might be able to run only on a particular operating system or language. When creating a profile, you need to customize it for the supported operating systems and languages .

Information about specifying operating system and language requirements includes the following topics:

- Setting Operating System Requirements and Service Pack Levels
- Setting Language Requirements
- How Requirements are Applied at Runtime
- Adding Pre-Launch and Post-Exit Scripts

Setting Operating System Requirements and Service Pack Levels

To specify the operating system and service pack level requirements for your application, perform the following steps.

Task ***To specify operating system requirement and service pack levels:***

1. In the **Citrix Assistant**, open the **Profile Requirements** page.
2. For the **Does your Citrix profile have any specific operating system requirements?** option, select one of the following:
   - **No**—Select this option if this application will run on all of the listed operating systems (which are the operating systems that the Citrix client supports). When this option is selected, the operating system check boxes are locked and cannot be changed.
   - **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems.
3. If you set the previous option to **Yes**, do the following:
   a. Select the operating systems that this application supports, and clear those that this application does not support.
   b. For each of the selected operating systems, double-click on it and select **Service Packs Requirement** from the context menu to open the **Service Packs Requirements** dialog box, and choose one of the following options:
      - **No Service Pack Requirement**—This application supports all versions of this operating system, regardless of the number of Service Packs installed.
      - **No Service Pack Allowed**—This application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly.
      - **Exact Service Pack Level**—This application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box.
      - **At Least Service Pack Level**—To run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box.
      - **At Most Service Pack Level**—To run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box.
      - **Range of Service Pack Levels**—To run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the **Minimum Level** and **Maximum Level** in the boxes.

Setting Language Requirements

To specify language requirements for your application, perform the following steps.

Task ***To specify operating system requirement and service pack levels:***

1. In the **Citrix Assistant**, open the **Profile Requirements** page.
2. For the **Does your Citrix profile have any specific language requirements?** option, select one of the following:

- **No**—Select this option if this application will run on all of the listed languages (which are the languages that the Citrix client supports). When this option is selected, the language check boxes are locked and cannot be changed.
- **Yes**—Select this option if the application does not support one of the listed languages. When you select this option, the check boxes are unlocked and only **English** is selected by default.

3. If you selected **Yes** in the previous step, select only those languages that this application supports.

How Requirements are Applied at Runtime

The requirements you specify on the **Profile Requirements** page determine how, or if, a user has access to the application.

When a user attempts to run an application, the Citrix XenApp checks to see whether that user's workstation meets the profile's specified requirements. Then, depending upon the **Application Type** assigned to that profile when it was published on the server, the user is:

- granted access to run the application locally, *or*
- granted access to run the application from the server, *or*
- denied access to the application.

The user access scenarios are presented in the following table:

Table 13-30 • Citrix XenApp User Access Scenarios

Application Type	User Access to Application
Accessed from a server	User runs the application on the Citrix XenApp, using shared server resources.
Streamed if possible, otherwise from a server	User access depends upon whether their workstation meets the profile's specified requirements: • **Meets requirements**—The profile is streamed (copied) to the user's workstation, and the user runs the application locally (from within its isolation environment). • **Does not meet requirements**—User runs the application on the Citrix XenApp, using shared server resources.
Streamed to client	User access depends upon whether their workstation meets the profile's specified requirements: • **Meets requirements**—The profile is streamed (copied) to the user's workstation, and the user runs the application locally (from within its isolation environment). • **Does not meet requirements**—User cannot access the application.

Caution • *If an application has specific operating system or language requirements and you fail to specify them correctly when creating the profile, users who do not meet those requirements will be given access to run applications locally and they will probably encounter application errors.*

Adding Pre-Launch and Post-Exit Scripts

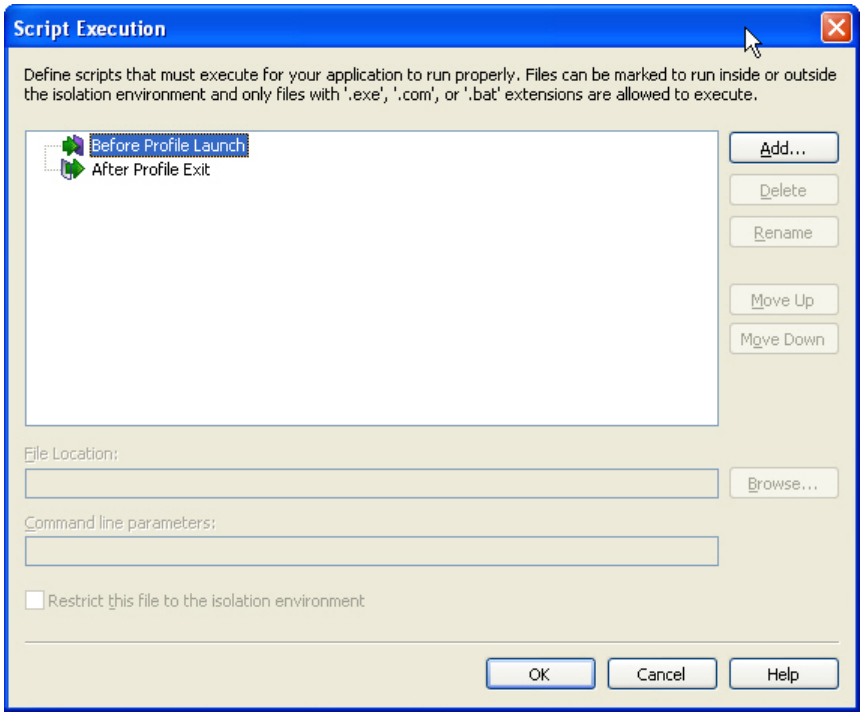
You can choose to include scripts with your profile that must execute either before profile launch or after profile exit in order for your application to run properly. On the **Script Execution** dialog box, which is opened by clicking **Script Execution** in the **More Options** list on the **Profile Requirements** page, you can view and manage all of the **Before Profile Launch** and **After Profile Exit** script files you are including with your Citrix profile.

- Files can be marked to run inside or outside of the isolation environment.
- Only files with `.exe`, `.cmd`, `.com`, or `.bat` extensions are allowed to execute.

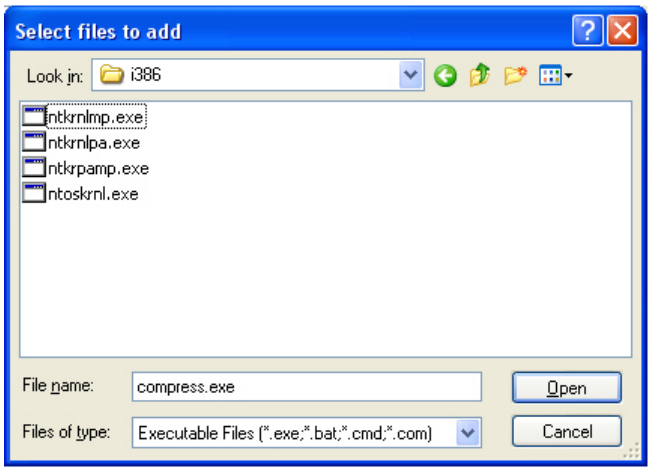
To add a script to your Citrix profile, perform the following steps.

Task ***To add a before launch or after exit script to your Citrix profile:***

1. Open the **Profile Requirements** page of the Citrix Assistant.
2. In the **More Options** list, click **Script Execution**. The **Script Execution** dialog box opens.



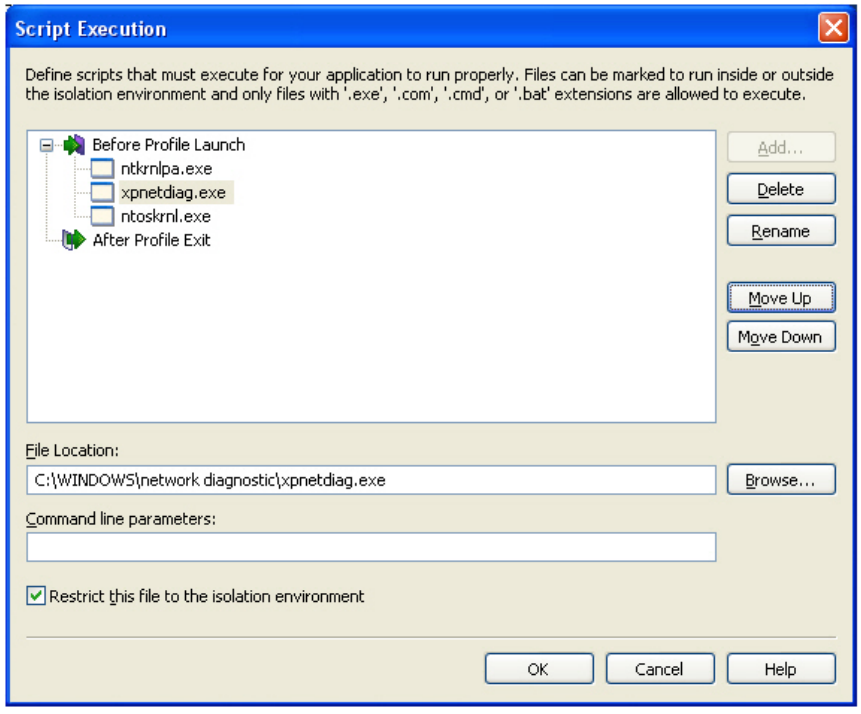
3. Select the **Before Profile Launch** or **After Profile Exit** node in the tree.
4. Click **Add...** The **Select Files to Add** dialog box opens.



5. Select the script file(s) (`.exe`, `.bat`, `.cmd`, or `.com`) that you want to add, and click **Open**. The file is added to the Script Execution tree on under the appropriate node.

Tip • *Use the Shift key to select multiple contiguous files, and use the Ctrl key to select multiple non-contiguous files.*

6. Select a script in the tree. Several new fields and options are enabled.



You can now perform any of the following tasks:

- **Rename the file's display name**—To rename the script file's display name, click the **Rename** button and enter a new name. The name that is displayed on this dialog box to identify the script is changed, but the original name of the script file is not changed.

- **Select a different script**—To select a different script, click the Browse button and select a different script file (`.exe`, `.bat`, `.cmd`, or `.com`).
- **Reorder scripts**—If multiple scripts are listed under a node, you can use the **Move Up** and **Move Down** buttons to change the order that the scripts will be run. You can also reorder the scripts using the Ctrl+Shift+Up Arrow and Ctrl+Shift+Down Arrow keys.
- **Restrict script to isolation environment**—If you want this script to only be able to run within the Citrix profile's isolation environment, select the **Restrict this file to the isolation environment** option.
- **Add command line parameters**—To add command line parameters to run along with the script, enter them in the **Command line parameters** box.
- **Delete a script**—To delete a script from the profile, select it and click the **Delete** button.

7. When you have set all desired options for the script, click **OK**.

Managing Files and Folders in a Citrix Profile

The next step in creating a Citrix profile is to view existing files and folders, add and delete files and folders, and override the default isolation options for folders and files.

The following tasks are performed on the **Profile Files** page.

- Adding, Deleting, and Moving Files and Folders in an App-V Package
- Controlling the Display of Predefined Folders
- Setting Isolation Options

Managing Files and Folders in a Citrix Profile

The directories in the destination tree on the **Profile Files** page of the Citrix Assistant represent how your application will be organized within its isolation environment.

On the **Profile Files** page, you can view all of the files and folders that are currently in your Citrix profile, add new files and folders to include in the Citrix profile, and delete files and folders from the Citrix profile.

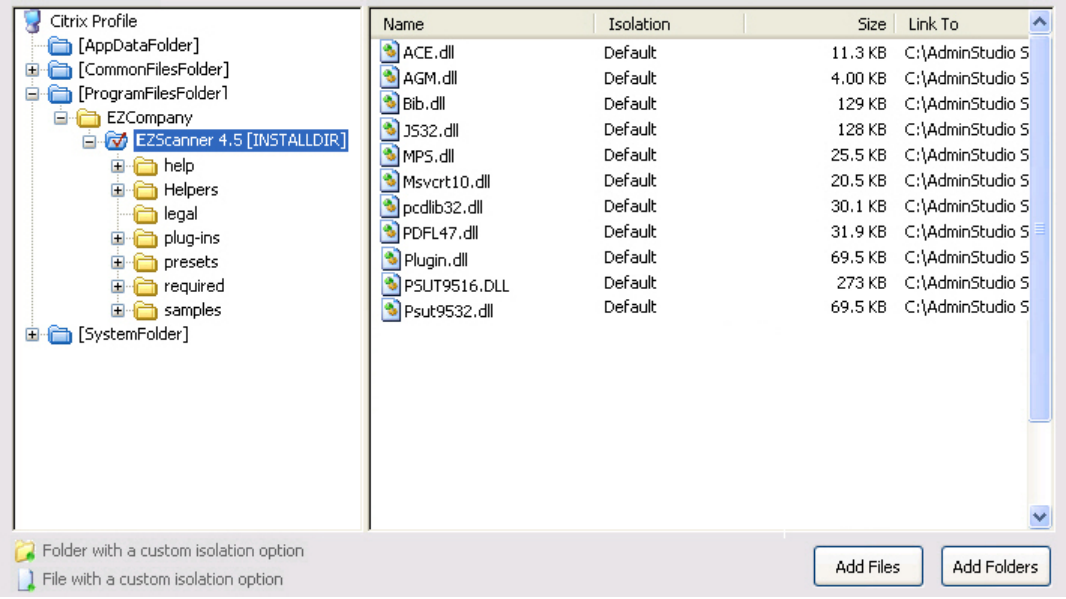
- Adding Files to a Citrix Profile
- Adding an Existing Folder (and its Contents) to a Citrix Profile
- Creating a New Folder
- Moving Files and Folders
- Deleting Files and Folders

Adding Files to a Citrix Profile

To add files to a Citrix profile, perform the following steps:

Task ***To add a files to a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Files** page. The files and folders are listed in the **Citrix Profile** tree, organized by installation directory.



   Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files From Your System

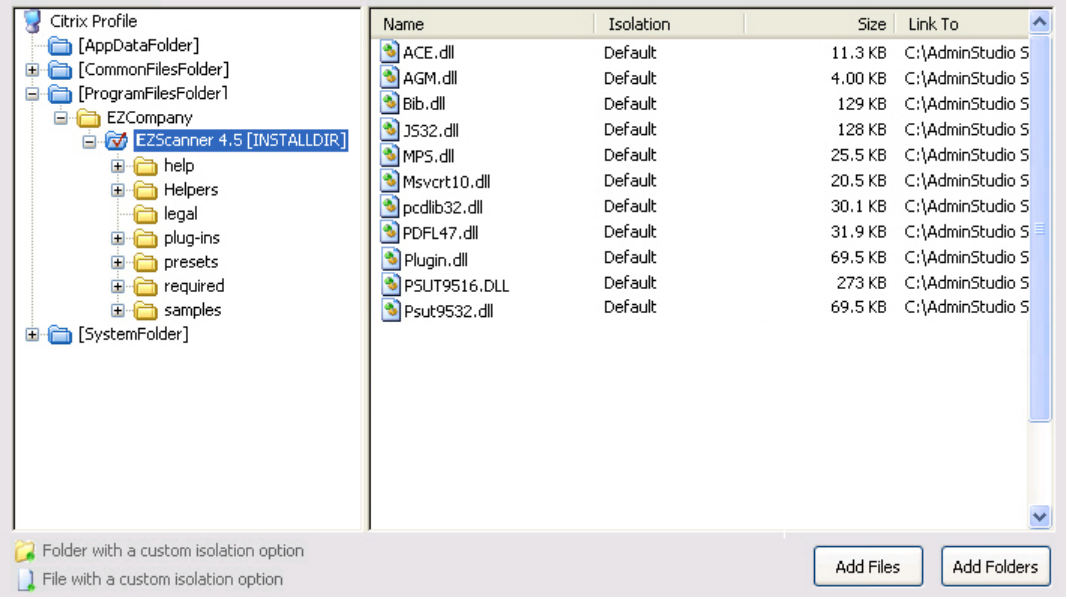
You can also add files or folders to your Citrix profile on the **Profile Files** page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and its Contents) to a Citrix Profile

To add an existing folder and all of the files and subfolders within it to a Citrix profile, perform the following steps:

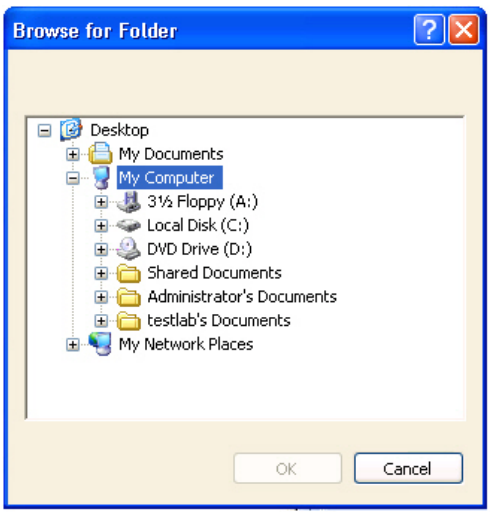
Task ***To add an existing folder to a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Files** page. The files and folders are listed in the **Citrix Profile** tree, organized by installation directory.



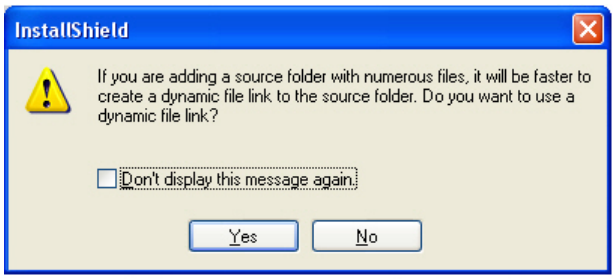
2. Browse through the folder tree to find the folder that you would like to add a folder into.

3. Select the folder and click the **Add Folders** button.The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.



4. Select a folder and click **OK**.

   If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.



5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with Citrix options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

You can create a new, empty folder by selecting an existing folder in the tree and selecting **New Folder** from the context menu.

Task ***To create a new folder:***

1. Right-click on a folder in the **Citrix Profile** tree and select **New Folder**. A new folder is created as a subfolder of the selected folder:

2. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the Citrix Profile folder tree structure, perform the following steps:

Task ***To move a file or folder:***

1. Select the file or folder that you want to move.
2. With the mouse button down, drag the file or folder to the new location.
3. Release the mouse button.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from a Citrix profile, perform the following steps:

Task ***To delete a file or folder:***

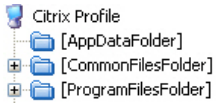
1. Select the file or folder in the **Citrix Profile** tree that you want to delete.
2. Select **Delete** from the context menu. You are prompted to confirm the deletion.
3. Click **Yes**. The selected file or folder is deleted.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire Project, not just from the Citrix profile.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the **Profile Files** page, the **Citrix Profile** tree initially displays the more commonly used predefined folders, such as `[ProgramFilesFolder]` and `[CommonFilesFolder]`.

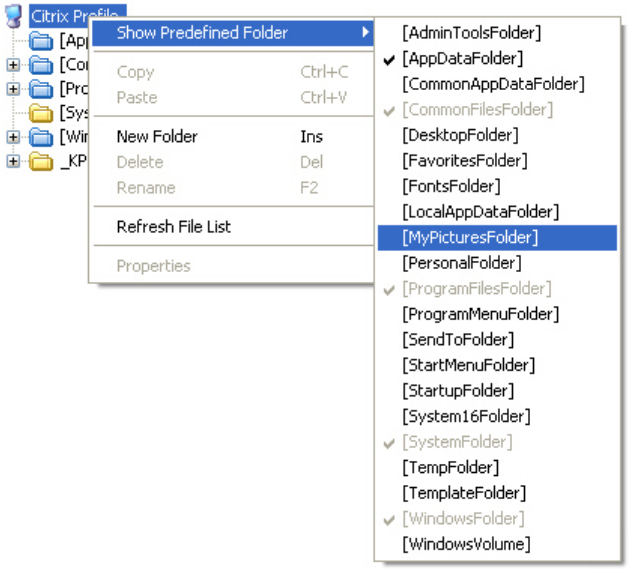


These predefined folders are dynamic, meaning that they do not use hard-coded paths. The value for each destination folder is obtained from the operating system of the target machine.

You can control which predefined folders are listed in this tree.

Task ***To change which predefined folders are listed:***

1. In the **Citrix Profile** tree, select the **Citrix Profile** node (or any of the files or folders that are listed, point to **Show Predefined Folder**. A list of predefined folders opens.



   Those folders that are already displayed are preceded by a check mark, and those that are not displayed do not have a check mark.

2. To add a folder to the tree listing, select a folder that is not currently listed in the tree.

   Note • *These predefined folders are always added to the root of the Citrix Profile tree, no matter what file or folder you had selected when you selected it from the Predefined Folders list.*

3. To remove a folder from the tree listing, select that folder name in this list (which is preceded by a check mark.

   Note • *You cannot turn off the display of the* `[ProgramFilesFolder]`.

Setting Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation options are set on the Citrix XenApp, and those defaults are adequate for most environments. However, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is open by selecting a file or folder and then selecting **Isolation Options** from the context menu.

Information about setting isolation options is presented in the following topics:

- Overview of Citrix Isolation Options
- Setting Isolation Options for Folders and Files
- Inheritance of Isolation Options from Folders to Files

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

Overview of Citrix Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation environments are set on the Citrix XenApp, and those defaults are adequate for most environments. However, in the Citrix Assistant, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a file or folder selected on the **Profile Files** page or a registry key selected on the **Profile Registry** page.

On the **Isolation Options** dialog box, you can choose one of the following isolation options:

Table 13-31 • Isolation Options

Option	Description
Default	Select this option if you want the default isolation option for this file/folder/registry key as defined on the Citrix XenApp to be applied to this selection. This is the default selection for all files, folders, and registry keys. ***Caution*** • *You should select this option unless you require specific custom handling.*

Table 13-31 • Isolation Options (cont.)

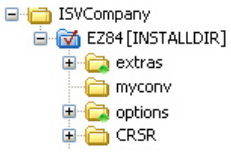
Option	Description
Ignore	Choose the **Ignore** option to direct the isolation environment to *always* use the copy of this selected file/folder/registry key that is on the system, not the one inside the isolation environment. Choosing this option gives the isolation environment direct access to the same location on the system that a non-isolated version of this application would have. By assigning the **Ignore** isolation option, you are creating a "hole" in the isolation environment to allow an application to write to the underlying system. For example, you would select **Ignore** in the following situations: ● If an application creates a directory for per-user data that is stored in a non-standard location. ● If the workstation has extra drive volumes and an installer writes to those drives while installing into a target. ● If your file share volume is on your packaging workstation. ● When the Citrix profile needs to share data with an application outside the isolation environment, such as when users print to a network printer.
Isolate	Choose the **Isolate** option to direct the isolation environment to first try to find the copy of this file/folder/registry key that is inside the isolation environment. If the item is not found there, then the isolation environment will use the copy of this file/folder/registry key that is on the system. Selecting **Isolate** ensures that the isolation environment is not given direct write access to the specified system resource.
Strictly Isolate	Choose the **Strictly Isolate** option to direct the isolation environment to always use the copy of this file/folder/registry key that is in the isolation environment, not on the system. This is useful when running two versions of an application on the same machine.
Redirect	Choose the **Redirect** option to redirect a request by the isolation environment for a file/folder/registry key to a specified location on the system (without first searching the user profile root and installation root locations). When selecting this option, you also need to select the location that the isolation environment should redirect to: ● **Source**—Lists the name of the selected item (filename, folder name, registry key). ● **Destination**—*[Files and folders only]* Click the Browse [...] button and select the file or folder on the system that you want to redirect to. ● **Destination Root**—*[Registry keys only]* Select the registry root of the registry key on the system that you want to redirect to. ● **Destination Key**—*[Registry keys only]* Select the registry key on the system that you want to redirect to.

Setting Isolation Options for Folders and Files

To override a file or folder's default isolation options set on the Citrix XenApp, perform the following steps:

Task ***To set an isolation option on a folder or file.***

1. Open the **Profile Files** page.
2. Browse through the folder tree to find the file or folder that you would like to modify.
3. Select the file or folder and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.
4. Select one of the following options, as described in Table 13-31, Isolation Options.
   - Default
   - Ignore
   - Isolate
   - Strictly Isolate
   - Redirect
5. Click **OK**. Files and folders that have an isolation setting other than default are marked with a special icon:



Inheritance of Isolation Options from Folders to Files

Isolation options for files, folders and registry keys are always inherited. The Citrix isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

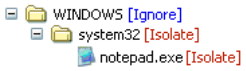


Figure 13-9: Example of Inheritance of Isolation Options from Folders to Files

Modifying Profile Shortcut Settings

You define profile shortcuts to enable users to launch a Citrix profile from within the isolation environment.

By default, the **Citrix Assistant** creates shortcuts to all of the executable (`.exe`) files that were added to the profile on the **Profile Files** page. These shortcuts are listed in a checklist on the **Profile Shortcuts** page.

Tip • *Citrix currently only supports 16 color icons for shortcuts. Therefore, if you specify an* ***Icon File*** *on the* ***Shortcuts*** *view of the Installation Designer, be sure to select an icon that includes only 16 colors.*

When you select each shortcut, details about it are displayed:

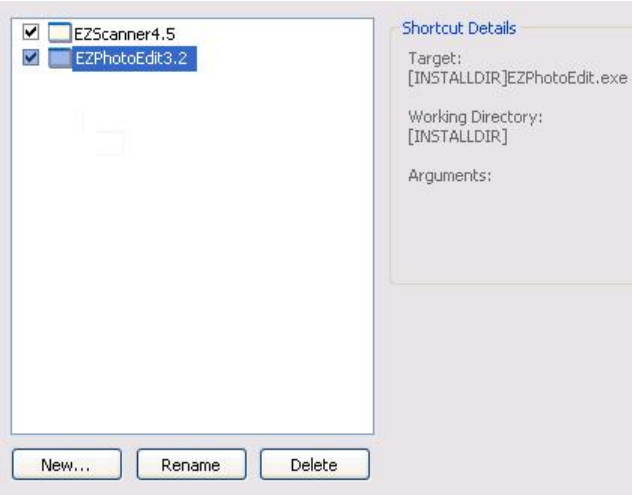


Figure 13-10: Initial List of Shortcuts for an Application

Caution • *You must define at least one shortcut to enable users to launch the application from the isolation environment.*

On the **Profile Shortcuts** page, you can create, delete, include, exclude, or rename a profile's shortcuts.

- App-V Packages and the Virtual Environment
- App-V Shortcut Requirements
- Creating a New App-V Package
- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Renaming a Shortcut

Shortcuts and the Isolation Environment

When a profile is published on the Citrix XenApp, the administrator has the option of placing available shortcuts on the client's desktop, client's Start menu, or only in the Citrix Program Neighborhood Agent applications list.

Shortcut presentation is specified in the **Application shortcut placement** area of the **Shortcut presentation** view of the Citrix Access Management Console **Application Properties** dialog box.

In the **Application shortcut placement** area, you have the following options:

Table 13-32 • Shortcut Presentation Options

Option	Description

Add to the client's Start menu	Select this option to create a shortcut to this application in the user's local **Start** menu. A Client Application folder appears in the first pane of the **Start** menu:
	`Start`
	`  MyApplicationFolder`
	`  ApplicationName`
	When you select this option, the **Place under Programs folder** and **Start menu folder** fields are enabled.

• **Place under Programs folder** (Program Neighborhood Agent Only)
• **Start menu folder** (Program Neighborhood Agent Only)
	Select the **Place under Programs folder** option to create a shortcut to this application under the **Programs** folder of the user's local **Start** menu.
	• If you leave the **Start menu folder** field blank, the shortcut is created in root folder of the **Programs** menu.
	`Start`
	`  Programs`
	`  MyApplicationFolder`
	`  ApplicationName`
	• If you specify a folder structure in the **Start menu folder** field, the shortcut is created in that folder structure within the local **Programs** folder, with each folder name separated with a backslash. For example, if you entered the following in the **Start menu folder** field:
	`MyApplicationFolder/ApplicationTools`
	Then, the shortcut would be created in the following folder structure:
	`Start`
	`  Programs`
	`  MyApplicationFolder`
	`  ApplicationTools`
	`  ApplicationName`

Add shortcut to the client's desktop	Select this option to create a shortcut to this application on the user's local desktop.

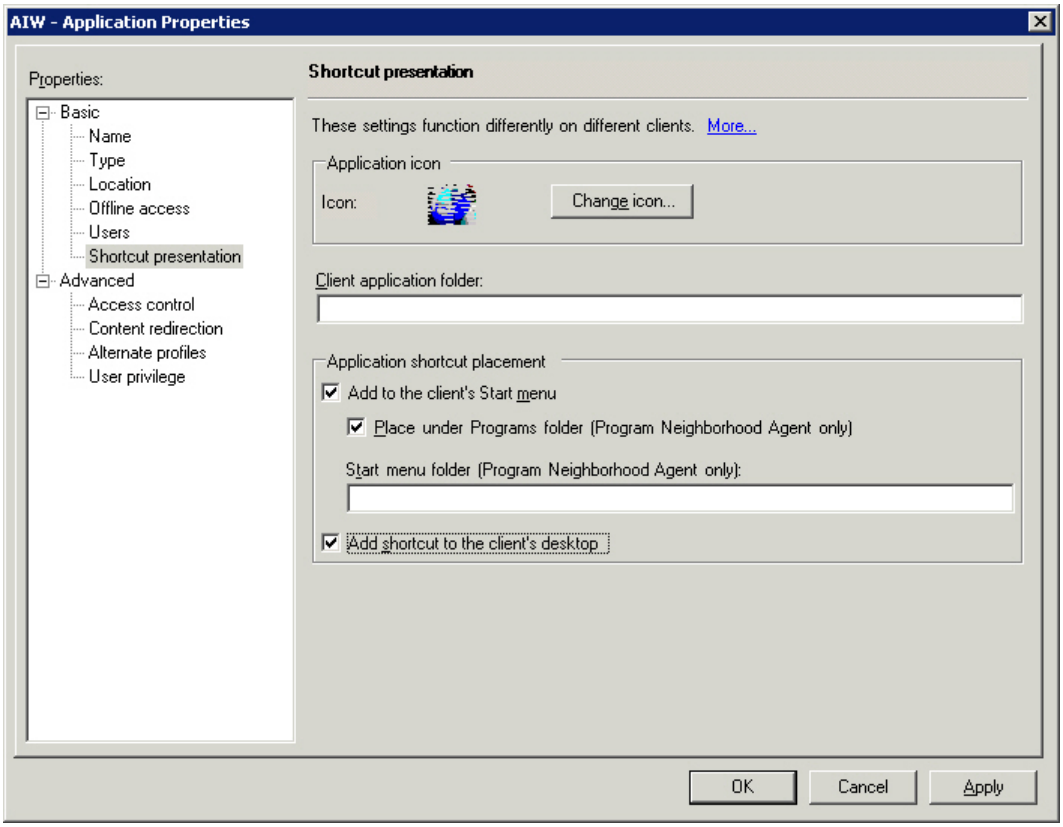


Figure 13-11: Shortcut Presentation View of the Citrix Presentation Server Application Properties Dialog Box

Shortcut Requirements

For each Citrix profile, you are required to define **at least one** shortcut. Profile shortcuts enable users to access the isolation environment and launch the application. If you build a Citrix profile that does not contain any shortcuts, users will not be able to launch the application.

Creating a New Profile Shortcut

On the **Profile Shortcuts** page, you can add a new shortcut to a file within the Citrix profile.

Task ***To create a new shortcut:***

1. Open the **Profile Shortcuts** page. All of the shortcuts are listed:
   - Those that are currently included in the profile are selected.
   - Those that are currently excluded from the profile are not selected.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this profile.
3. Select the file that you want to create a shortcut to.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.

5. To include this shortcut in the Citrix profile, make sure that its check box is selected.

Including an Existing Profile Shortcut

If you want to include a previously excluded shortcut in a Citrix profile, perform the following steps:

Task ***To include an existing profile shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (.exe) files that were added on the **Profile Files** page are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding vs. Deleting a Profile Shortcut

By default, the **Citrix Assistant** creates shortcuts to all of the executable (.exe) files that were added on the **Profile Files** page, and lists them in a checklist on the **Profile Shortcuts** page.

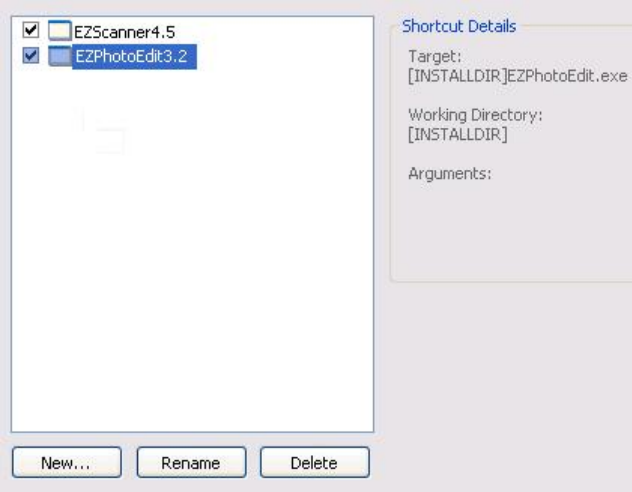


Figure 13-12: Initial List of Shortcuts for an Application

To prevent the shortcut from being created in the Citrix profile, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. See Deleting a Shortcut.

If you have any unnecessary shortcuts in your project, you can simply exclude them from the Profile by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding a Profile Shortcut

If you want to exclude one of these shortcuts from being created in the Citrix profile, perform the following steps:

Task ***To exclude a shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (.exe) files that were added on the **Profile Files** page are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To exclude a shortcut, select the shortcut and clear the check box.

Note • *When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project.*

Deleting a Shortcut

To delete a shortcut, perform the following steps.

Task ***To delete a shortcut:***

1. Open the **Profile Shortcuts** page. All of the shortcuts are listed.
2. Select the shortcut and click **Delete**.

Note • *If you delete a shortcut on the **Profile Shortcuts** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Conditions When a Shortcut Should be Excluded or Deleted

To prevent a shortcut from being created in the Citrix profile, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Renaming a Shortcut

To rename a shortcut, perform the following steps:

Task ***To add or delete a shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (**.**exe) files that were added on the **Profile Files** page are listed.
2. Select the shortcut that you want to rename and click **Rename**. A box appears around the shortcut name, and the shortcut name becomes an editable field.
3. Enter a new name for the shortcut.

Modifying Profile Registry Settings

Using the **Citrix Assistant**, you can add, delete, or modify the registry settings in your Citrix profile.

You can also override the Citrix default isolation options for selected registry keys. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

Information about modifying profile registry settings on the **Profile Registry** page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values
- Setting App-V Package Registry Isolation Options

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_USER
- HKEY_CLASSES_ROOT

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the **Profile Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

Setting Registry Isolation Options

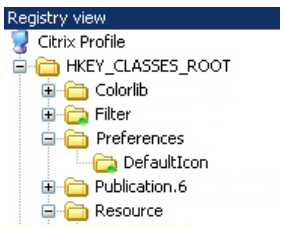
To override a registry key's default isolation options set on the Citrix XenApp, perform the following steps:

Task ***To set an isolation option on a registry key:***

1. Open the **Profile Files** page.
2. Browse through the registry tree to find the key that you would like to modify.
3. Select the folder or key and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.

   Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

4. Select one of the following options, as described in Table 13-31, Isolation Options.
   - Default
   - Ignore
   - Isolate
   - Strictly Isolate
   - Redirect
5. Click **OK**. Registry keys that have an isolation setting other than default are marked with a special icon:



Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Inheritance of Isolation Options in the Registry

Isolation options for registry keys are always inherited. The Citrix isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

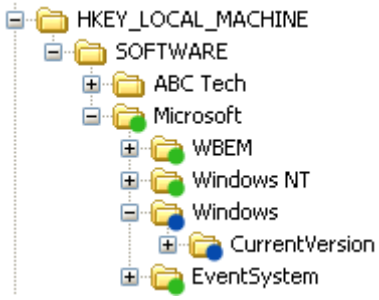


Figure 13-13: Example of Inheritance of Isolation Options from Folders to Files

Modifying Build Settings

On the **Build Settings** page, you choose which releases of this InstallShield project you want to build a Citrix profile for when the project is built, specify whether you want to digitally sign the Citrix profile, and specify whether you want to include additional Windows Installer packages in the Citrix profile.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the **Build Citrix Profile** option on the **Build Settings** page before you will be able to build a Citrix profile for that Windows Installer package.

- Selecting the Releases for Which You Want to Build App-V Packages
- Digitally Signing a Citrix Profile
- Including Additional Windows Installer Packages in a Citrix Profile
- Enabling App-V Package Building When in Direct Edit Mode

Important • *You must create at least one Release (on the* ***Releases*** *view of the Installation Designer) before you will be able to select a Release on the* ***Build Settings*** *page.*

Selecting Releases to Build

You select the releases that you want to build a Citrix profile for on the **Releases** tree of the **Build Settings** page.

Important • *You cannot create or edit a release in the Citrix Assistant. If no releases exist, you can simply click the* ***Build*** *toolbar button to create a new release or open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see Creating and Building Releases.*

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the ***Releases*** *tree on the* ***Build Settings*** *page is not displayed.*

Task ***To select releases to build:***

1. Open the **Build Settings** page.
2. Select the releases in the **Releases** tree that you want to build a Citrix profile for.

   Important • *When you select a release on the* ***Build Settings*** *page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the* ***Build Settings*** *page have no bearing upon which release is built when you click the* ***Build*** *button on the toolbar. When you initiate a build by clicking the* ***Build*** *button, a build is initiated for the* **active** *release—the release that was most recently selected on the Installation Designer* ***Releases*** *view. The output of that build would depend upon what releases were selected on the* ***Build Settings*** *page:*

   - **Active release selected**—*A Windows Installer package and a Citrix profile would be built.*
   - **Active release not selected**—*Only a Windows Installer package would be built.*

   Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Digitally Signing a Citrix Profile

You can digitally sign your Citrix profile to assure end users that neither your installation nor the code within your application has been tampered with or altered since publication. When you digitally sign your application, end users are presented with a digital certificate when they run your installation.

Task ***To digitally sign a Citrix profile:***

1. Open the **Build Settings** page.
2. Select the **Digitally sign Citrix profile** option. The **Personal Information Exchange file (.pfx)** field is enabled. A `.pfx` file is a standard file format for digital certificates.
3. Click **Browse** and select the `.pfx` file that you want to use to digitally sign this Citrix profile.

Including Additional Windows Installer Packages in a Citrix Profile

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual application, you can also use the Citrix Assistant to convert an application suite of multiple Windows Installer packages into one virtual application.

To include additional Windows Installer packages in a Citrix profile, set the **Would you like to include additional MSI files in the virtual package?** option on the **Build Settings** page to **Yes**, and then select the packages that you want to add.

Task ***To include additional Windows installer packages in a Citrix profile:***

1. Open the **Build Settings** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down () buttons.
   - Use the Delete button ( ) to delete a package from the list.

Enabling Citrix Profile Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Citrix Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a Citrix profile for this Windows Installer package. To do this, perform the following steps:

Task ***To enable Citrix profile building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. Open the **Build Settings** page of the Citrix Assistant.
3. Select the **Build Citrix Profile** option. After you select this option, the **Build Citrix Profile** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Building a Citrix Profile

The method for building a Citrix profile depends upon what file you have open—an InstallShield project or a Windows Installer package.

- Building an App-V Package from Within an InstallShield Project

- Building an App-V Package from Within a Windows Installer Package in InstallShield

Building a Citrix Profile for an InstallShield Project

To build a Citrix profile for an InstallShield project, perform the following steps:

Task ***To build a Citrix profile for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

   Important • *You cannot create or edit a release in the Citrix Assistant. If no releases exist, or if you want to create a new release, open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see Creating and Building Releases.*
3. Open the **Build Settings** page of the Citrix Assistant.
4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build a Citrix profile for.

   Important • *When you select a release on the* ***Build Settings*** *page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the* ***Build Settings*** *page have no bearing upon which release is built when you click the* ***Build*** *button on the toolbar. When you initiate a build by clicking the* ***Build*** *button, a build is initiated for the* **active** *release—the release that was most recently selected on the Installation Designer* ***Releases*** *view. The output of that build would depend upon what was selected on the* ***Build Settings*** *page:*

   - **Active release selected**—*A Windows Installer package and a Citrix profile would be built.*
   - **Active release not selected**—*Only a Windows Installer package would be built.*

   To build more than one release at a time, perform a batch build. See Performing Batch Builds.
5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

   The output of the build will be a Windows Installer package and a Citrix profile. For a description of the files that comprise a Citrix profile, see Components of an App-V Package.

Building a Citrix Profile for a Windows Installer Package

To build a Citrix profile for a Windows Installer package, perform the following steps:

Task ***To build a Citrix profile for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).

- On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
- On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.

2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the Citrix Assistant to set Citrix profile options.
3. On the **Build Settings** page of the Citrix Assistant, select the **Build Citrix profile** option.
4. Save the edits to the Windows Installer package or transform file by selecting **Save** on the **File** menu.
5. Click the **Build** toolbar button (or select **Build Citrix Profile** on the **Build** menu) to start building the Citrix profile.

   The output of the build will be a Windows Installer package and a Citrix profile. For a description of the files that comprise a Citrix profile, see Components of an App-V Package.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Citrix Assistant Reference

Reference information about the Citrix Assistant is organized into the following sections:

- Microsoft App-V Assistant Pages
- Microsoft App-V Assistant Dialog Boxes
- Building App-V Packages Through the Command Line
- Troubleshooting the Builds of App-V Packages
- Application Features that Require Pre- or Post-Conversion Actions

Pages

The Citrix Assistant is comprised of the following pages:

- Home Page
- Profile Information Page
- Profile Requirements Page
- Profile Files Page
- Profile Shortcuts Page
- Profile Registry Page
- Build Settings Page

Home Page

The Citrix Assistant Home page displays a diagram that illustrates the process of creating a Citrix profile for deployment on Citrix XenApp.

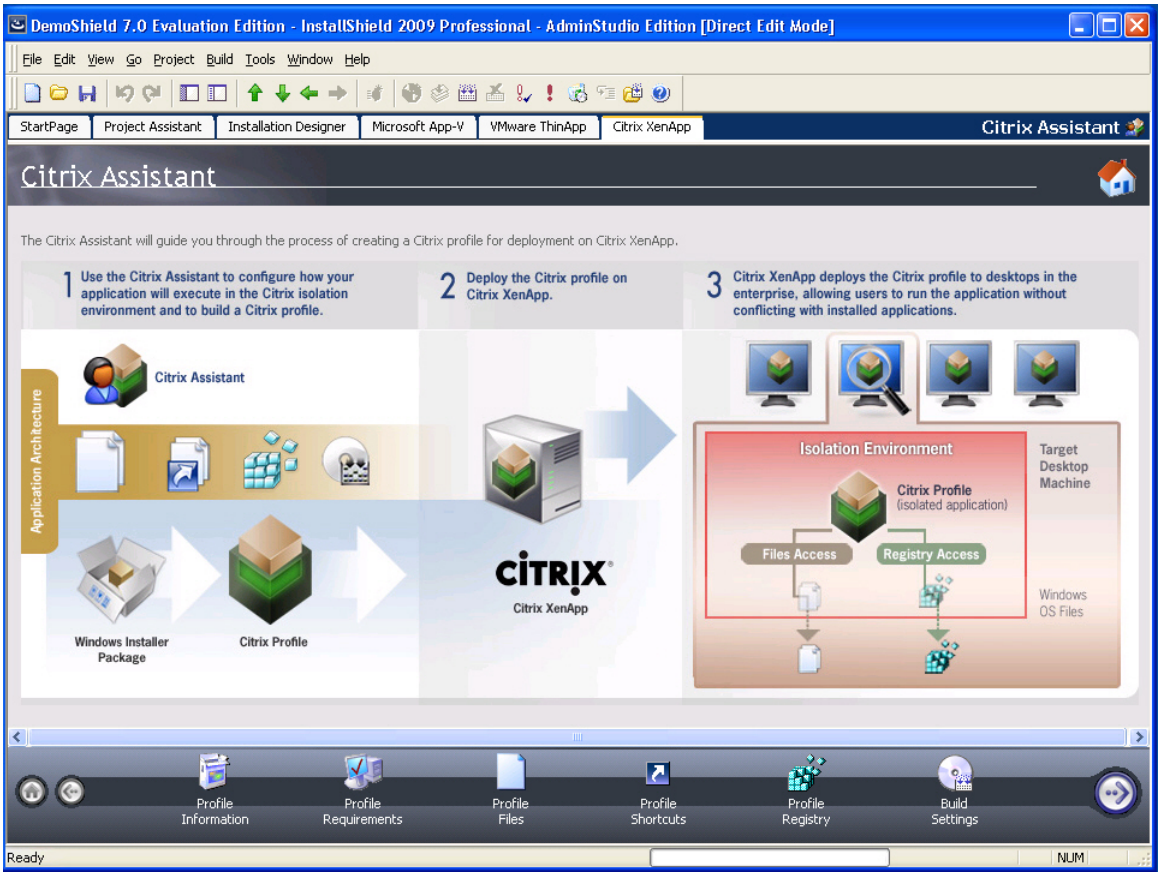


Figure 13-14: Citrix Assistant Home Page

Click the following icons in the navigation bar at the bottom of the page to navigate through the Citrix Assistant interface:

Table 13-33 • Navigation Bar Icons

Icon	Destination
	Profile Information Page
	Profile Requirements Page
	Profile Files Page

Table 13-33 • Navigation Bar Icons

Icon	Destination
	Profile Shortcuts Page
	Profile Registry Page
	Build Settings Page
	Go to next page.
	Jump back to previous page.
	Home Page

Profile Information Page

On the **Profile Information** page in the **Citrix Assistant**, you specify the **Name**, **Description**, and **Version** of the Citrix profile you are creating. This page is also where you specify whether users can update applications.

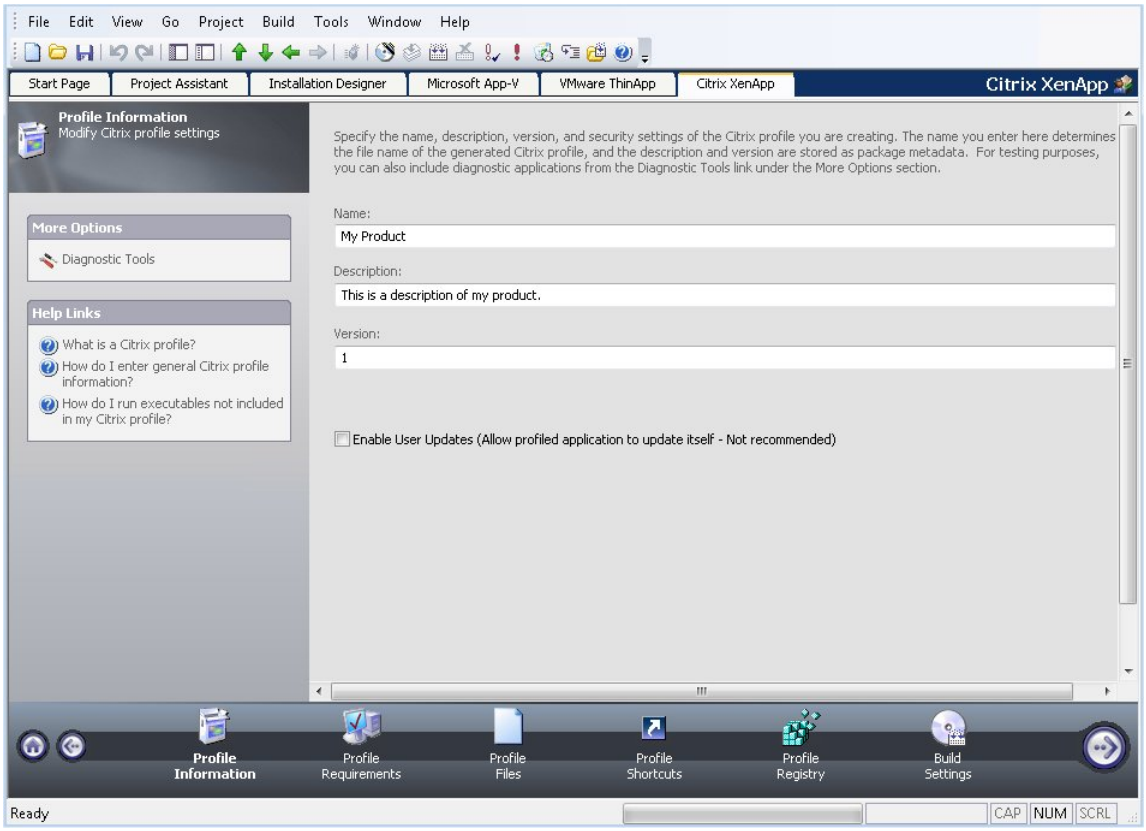


Figure 13-15: Citrix Assistant Profile Information Page

The Profile Information page includes the following options:

Table 13-34 • Profile Information Page

Option	Description
Name	Enter a name for this Citrix profile. The name you enter here determines the file name of the generated Citrix profile. ***Note*** • *Do not include the version number in the package name.*
Description	Briefly describe this package. This information is stored with the package as metadata.
Version	Enter the version number of this Citrix profile. This information is stored as package metadata.

Table 13-34 • Profile Information Page (cont.)

Option	Description
Enable User Updates (Allow profiled application to update itself—Not recommended)	To allow the profiled application to download and install vendor-supplied updates over the Internet, select this check box. The updates are stored within the user profile root location for the specific user. To ensure that all executable files from the profile are launched from the installation root location, and not from the user profile location, clear this check box. When this check box is cleared, the system prevents code from being run if it is not streamed from the server. Clearing this check box enables you to control updates through the profiler. This check box is cleared by default.

For testing purposes, you can also choose to include diagnostic tools in your Citrix profile by selecting the **Diagnostic Tools** link in the **More Options** list. For more information, see Diagnostic Tools Dialog Box.

Profile Requirements Page

On the **Profile Requirements** page of the Citrix Assistant, you can select the operating systems that client workstations must be running in order for your application to operate properly. By default, all operating systems that are supported by the Citrix client are selected.

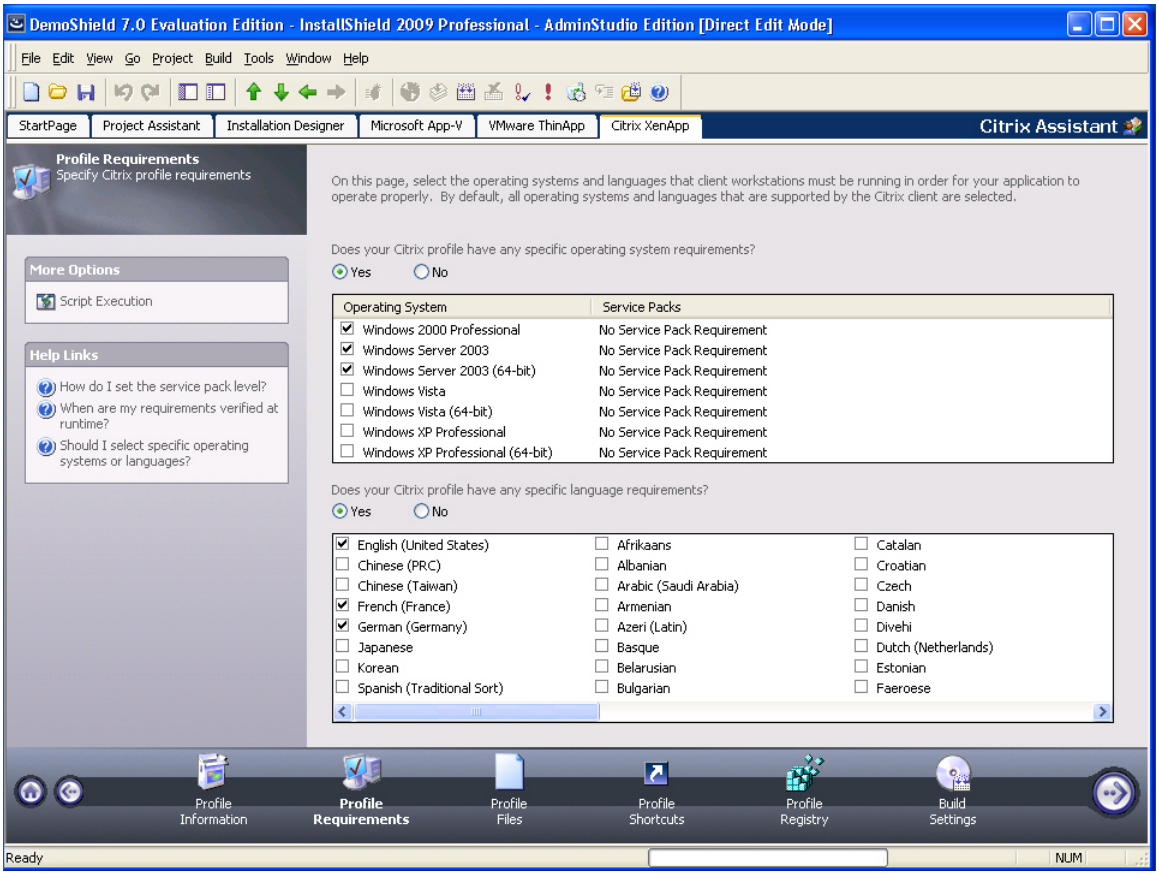


Figure 13-16: Citrix Assistant Profile Requirements Page

The **Profile Requirements** page has the following options:

Table 13-35 • Profile Requirements Page Options

Option	Description
Does your Citrix profile have any specific operating system requirements?	Select one of the following: • **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems. • **No**—Select this option if this application will run on all of the listed operating systems (which are the operating systems that the Citrix client supports). When this option is selected, the operating system check boxes are locked and cannot be changed.

Table 13-35 • Profile Requirements Page Options

Option	Description
Operating System / Service Packs List	If you set the **Does your Citrix profile have any specific operating system requirements?** option to **Yes**, this list becomes enabled. To specify operating system requirements, first select the operating systems that this application supports, and clear those that this application does not support. Then, for each of the selected operating systems, right-click on it and select **Service Packs Requirement** from the context menu to open the **Service Packs Requirements** dialog box, and choose one of the following options: • **Not Required**—This application supports all versions of this operating system, regardless of the number of Service Packs installed. • **No Service Packs Allowed**—This application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly. • **Exact Service Pack Level**—This application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box. • **At least Service Pack Level**—To run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box. • **At most Service Pack Level**—To run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box. • **Range of Service Pack Levels**—To run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the Minimum Level and Maximum Level in the boxes.
Does your Citrix profile have any specific language requirements?	Select one of the following: • **No**—Select this option if this application will run on all of the listed languages (which are the languages that the Citrix client supports). When this option is selected, the language check boxes are locked and cannot be changed. • **Yes**—Select this option if the application does not support one of the listed languages. When you select this option, the check boxes are unlocked and only **English** is selected by default.
Language List	If you set the **Does your Citrix profile have any specific language requirements?** option to **Yes**, this list becomes enabled. Select only those languages that this application supports.

Profile Files Page

On the **Profile Files** page of the Citrix Assistant, you can perform the following tasks:

- View Files and Folders
- Add Files and Folders
- Delete Files and Folders
- Set Isolation Options
- Modifying the Display of Predefined Folders

View Files and Folders

On the **Profile Files** page, you can view all of the files and folders that are currently in your Citrix profile.

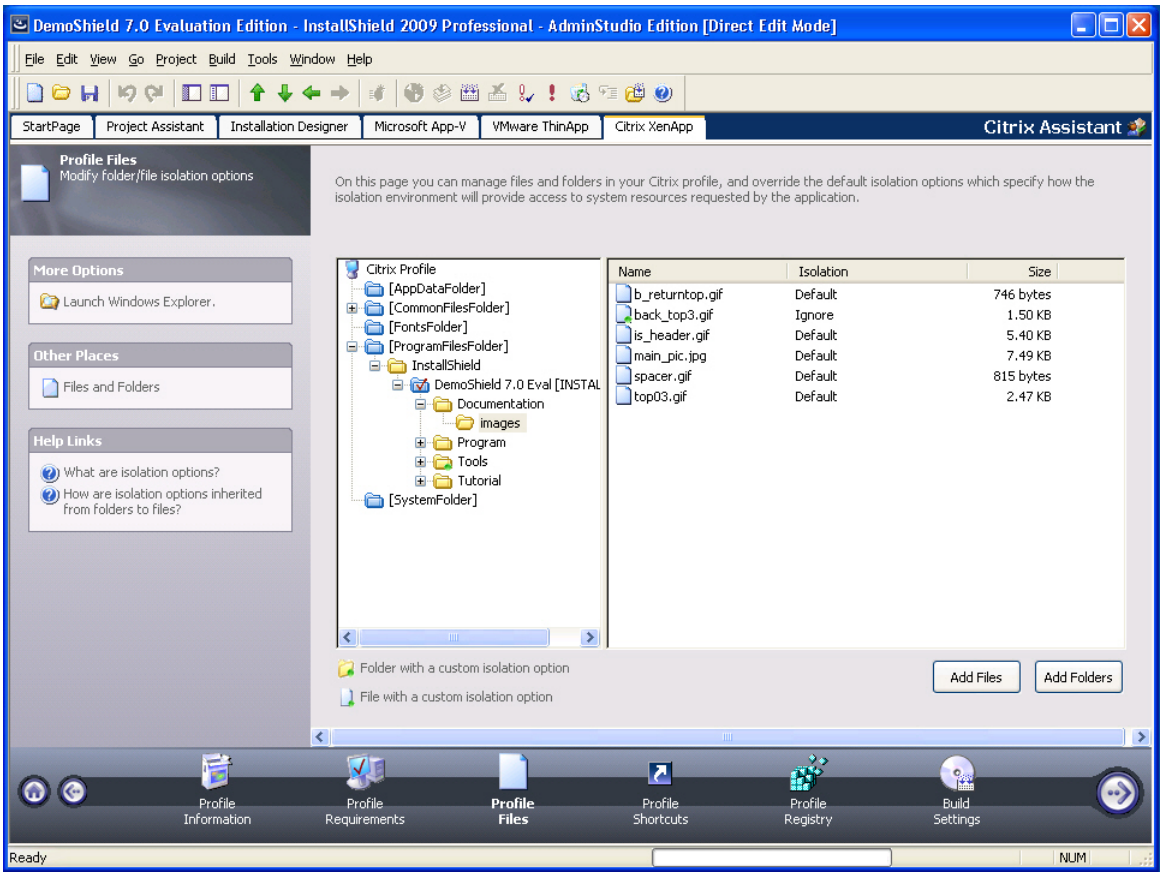


Figure 13-17: Citrix Assistant Profile Files Page

Folders are listed in the Citrix Profile tree on the left, and all of the files in the selected folder are listed on the right.

- The directories in the tree represent how your application will look when it is installed on to your customer's machine.
- Blue folders are the supported MSI standard folders.
- The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

Add Files and Folders

On the **Profile Files** page, you can use the **Add Files** and **Add Folders** buttons to add new files and folders to include in the Citrix profile. See Adding, Deleting, and Moving Files and Folders in an App-V Package.

If you are editing an InstallShield project (not a Windows Installer package), and you are adding a folder to this profile, you are prompted to choose whether you want to create a dynamic file link to the source folder.

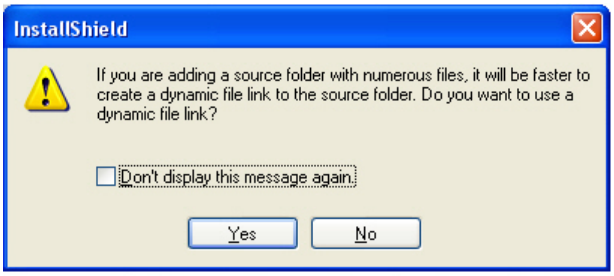


Figure 13-18: Dynamic File Link Option Dialog Box

Indicate whether you want to create a dynamic file link by selecting one of the following:

- **No**—For more flexibility with Citrix options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
- **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

Delete Files and Folders

You can delete files and folders from the Citrix profile by selecting the file or folder you want to delete, and selecting **Delete** from the context menu. For more information, see Deleting Files and Folders.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains.*

Note • *You cannot delete predefined folders. You can only turn of the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Set Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation environments are set on the Citrix XenApp, and those defaults are adequate for most environments. However, in the Citrix Assistant, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a file or folder and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting Isolation Options.

Modifying the Display of Predefined Folders

You can specify which of the Windows Installer predefined folders are listed in the **Citrix Profile** tree. See Controlling the Display of Predefined Folders.

Profile Shortcuts Page

You define profile shortcuts to enable users to launch a Citrix profile from within the isolation environment.

By default, the **Citrix Assistant** creates shortcuts to all of the executable (.exe) files that were added to the profile. These shortcuts are listed in a checklist on the **Profile Shortcuts** page.

Note • *Only shortcuts to executables are included in the profile. The Citrix Administrator chooses which of these shortcuts will be available to their users. When publishing, the Citrix Administrator chooses where to place the shortcut for their users to see.*

Tip • *Citrix currently only supports 16 color icons for shortcuts. Therefore, if you specify an **Icon File** on the **Shortcuts** view of the Installation Designer, be sure to select an icon that includes only 16 colors.*

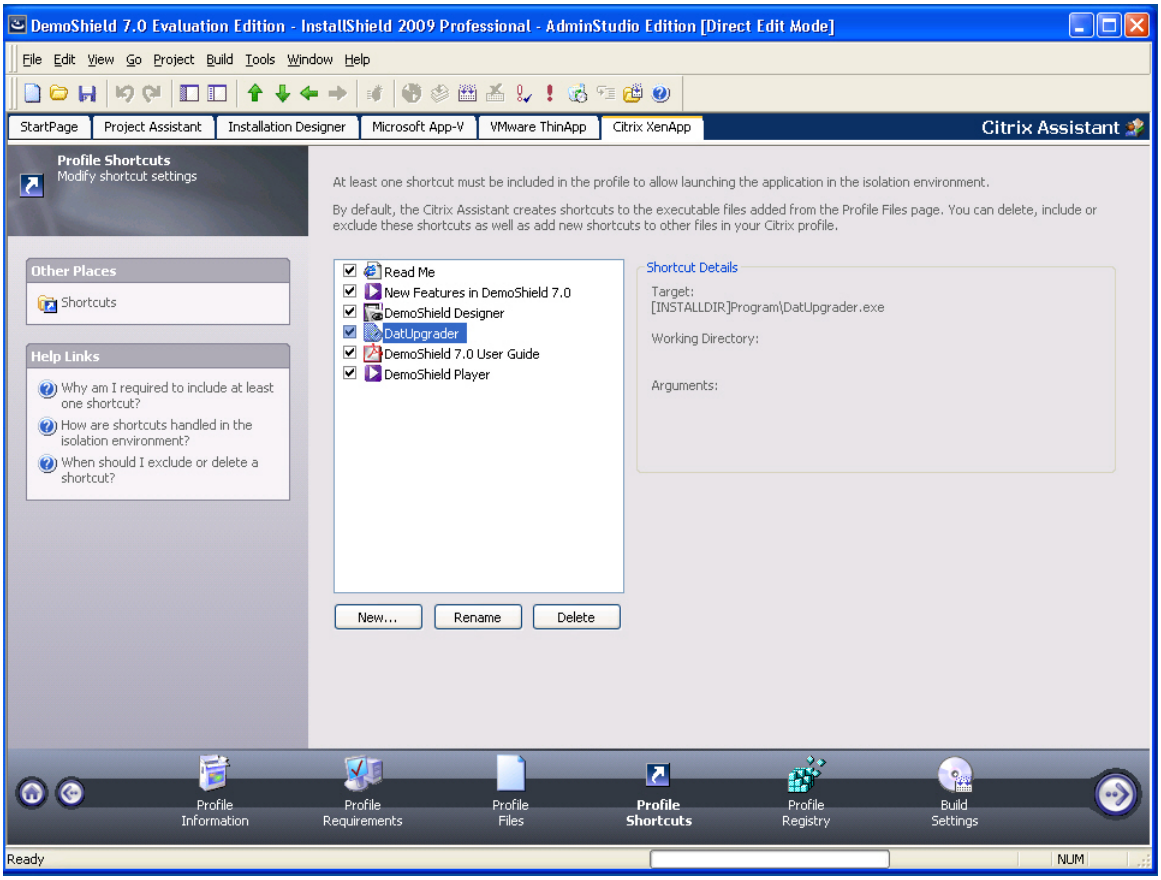


Figure 13-19: Profile Shortcuts Page

Shortcut Requirements

For each Citrix profile, you are required to define **at least one** shortcut. Profile shortcuts enable users to access the isolation environment and launch the application. If you build a Citrix profile that does not contain any shortcuts, users will not be able to launch the application.

Difference Between Deleting and Excluding a Profile Shortcut

To prevent a shortcut from being created in the Citrix profile, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Managing Shortcuts

On the **Profile Shortcuts** page, you can create, delete, include, exclude, or rename a profile's shortcuts. For step-by-step instructions, see the following topics:

- Creating a New App-V Package
- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Renaming a Shortcut

Profile Registry Page

On the **Profile Registry** page, you can view existing registry items, and add or delete registry items. You can also override the Citrix default isolation options for a registry key. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

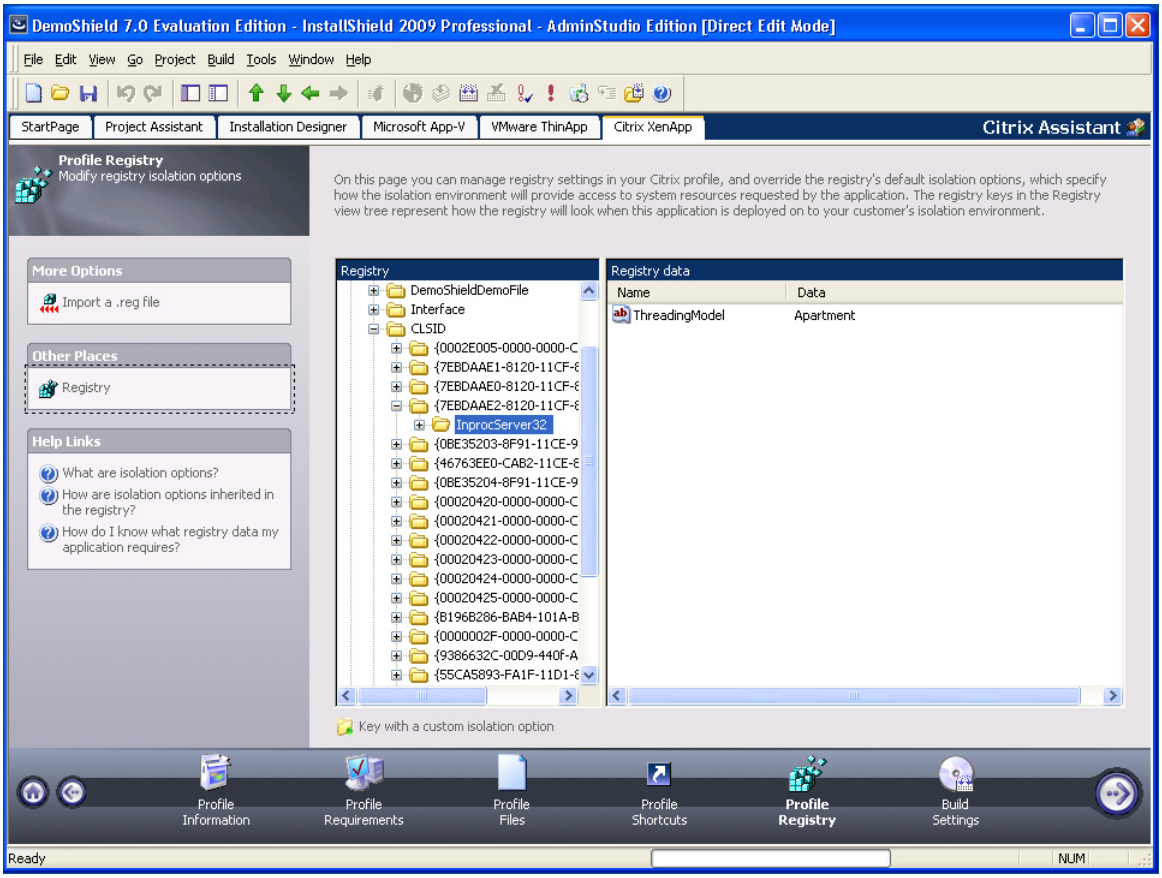


Figure 13-20: Citrix Assistant Profile Registry Page

Registry items that are listed on this page will be included in the Citrix profile, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the **Profile Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

For information on how to override a registry key's default isolation options, see Setting App-V Package Registry Isolation Options.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Build Settings Page

On the **Build Settings** page, you can perform the following tasks:

- Selecting Releases to Build
- Digitally Signing a Citrix Profile
- Including Additional Windows Installer Packages in a Citrix Profile
- Enabling Citrix Profile Building When in Direct Edit Mode
- Building a Citrix Profile

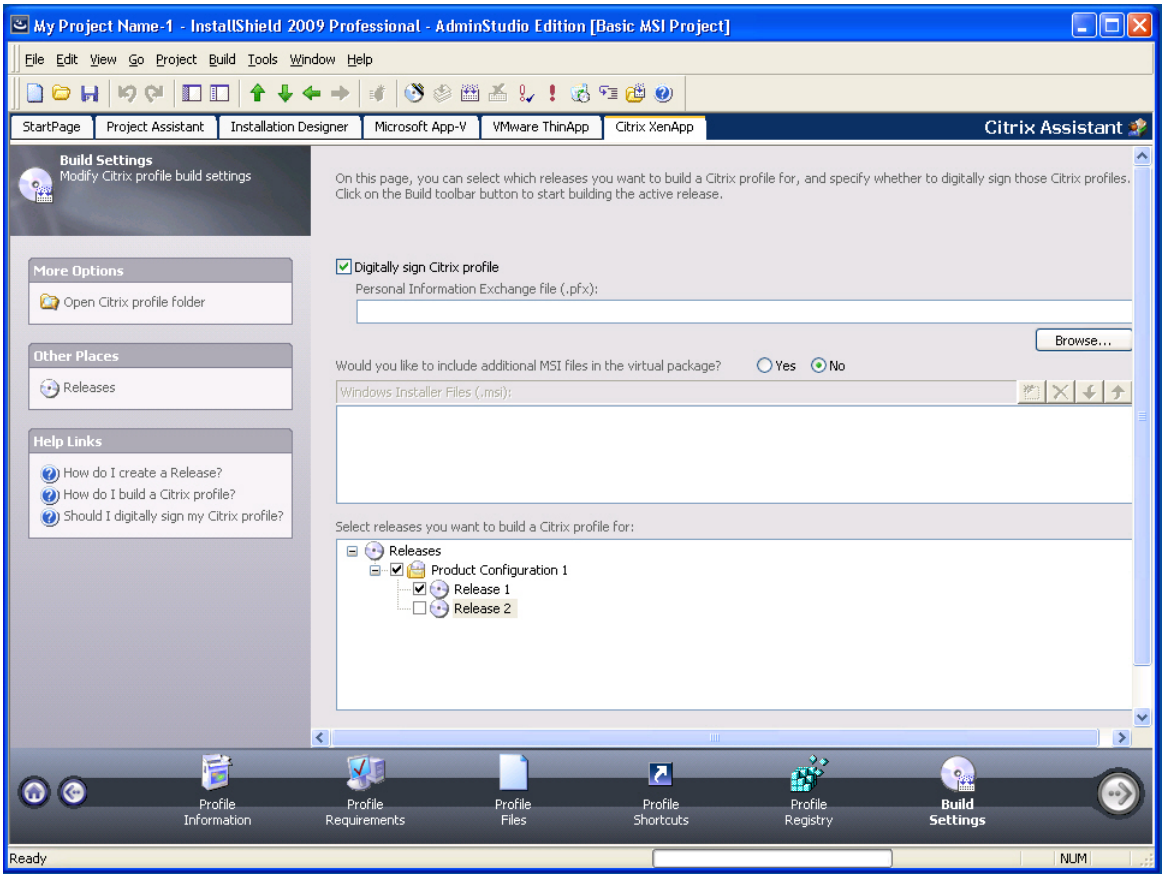


Figure 13-21: Citrix Assistant Build Settings Page

Selecting Releases to Build

You select the releases that you want to build a Citrix profile for on the **Releases** tree of the **Build Settings** page. By selecting a release, you are specifying that whenever that particular release is built, a Citrix profile will also be built.

Note • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the* ***Releases*** *tree on the* ***Build Settings*** *page is not displayed.*

About Building Releases

When you select a release on the Releases tree on the **Build Settings** page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the **Build Settings** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Settings** page:

- **Active release selected**—A Windows Installer package and a Citrix profile would be built.
- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

About Creating Releases

You create and edit releases on the **Releases** view of the InstallShield Installation Designer. You cannot create or edit a release in the Citrix Assistant.

If no releases exist, or if you want to create a new release, open the **Releases** view of the Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see Creating and Building Releases.

Digitally Signing a Citrix Profile

You can digitally sign your Citrix profile to assure end users that neither your installation nor the code within your application has been tampered with or altered since publication. When you digitally sign your application, end users are presented with a digital certificate when they run your installation.

To digitally sign a Citrix profile, select the **Digitally sign Citrix profile** option on the **Build Settings** page. When this option is selected, the **Personal Information Exchange file (.pfx)** field is enabled. A `.pfx` file is a standard file format for digital certificates. You then click **Browse** and select the `.pfx` file that you want to use to digitally sign this Citrix profile.

Including Additional Windows Installer Packages in a Citrix Profile

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the Citrix Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a Citrix profile, set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**, and then select the packages that you want to add.

- Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
- The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
- Use the Delete button ( ) to delete a package from the list.

Enabling Citrix Profile Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Citrix Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a Citrix profile for this Windows Installer package. To enable the **Build** button to build the Citrix profile, select the **Build Citrix Profile** option on the **Build Settings** page.

After you select this option, the **Build Citrix Profile** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Building a Citrix Profile

The method for building a Citrix profile depends upon what file you have open—an InstallShield project or a Windows Installer package. For detailed instructions, see one of the following topics:

- Building an App-V Package from Within an InstallShield Project
- Building an App-V Package from Within a Windows Installer Package in InstallShield

Dialog Boxes

The Citrix Assistant includes the following dialog boxes:

- Script Execution Dialog Box
- Diagnostic Tools Dialog Box
- File Isolation Options Dialog Box
- Folder Isolation Options Dialog Box
- Isolation Options Dialog Box (for Registry Keys)
- Service Packs Requirement Dialog Box

Script Execution Dialog Box

On the **Script Execution** dialog box, which is opened by clicking **Script Execution** in the **More Options** list on the **Profile Requirements** page, you can choose to include scripts that must execute for your application to run properly. From this dialog box, you can view and manage all of the **Before Profile Launch** and **After Profile Exit** script files you are including with your Citrix profile.

- Files can be marked to run inside or outside of the isolation environment.
- Only files with `.exe`, `.com`, `.cmd`, or `.bat` extensions are allowed to execute.

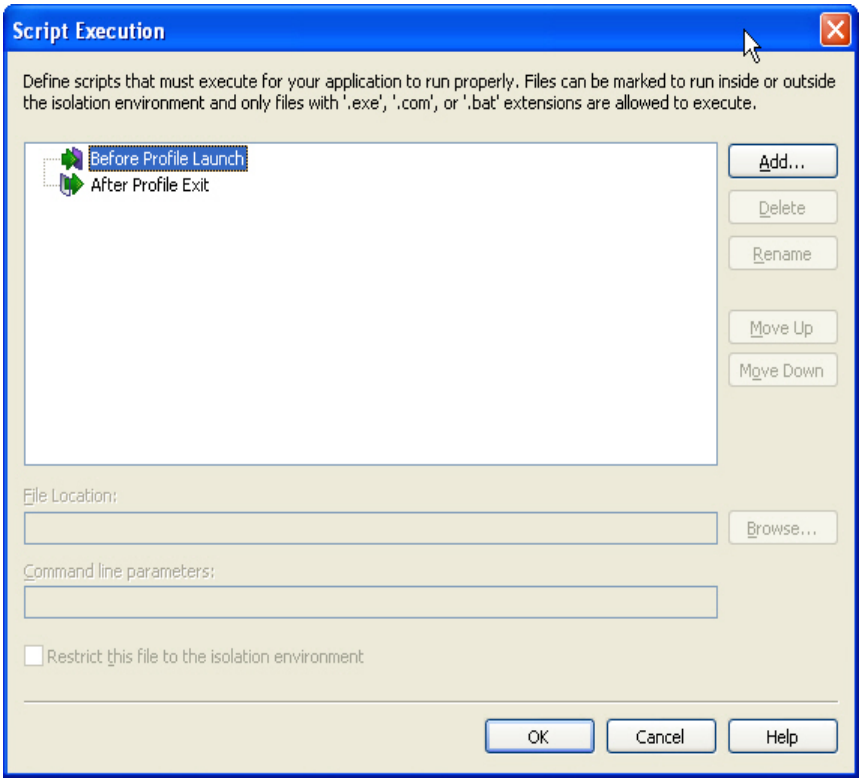


Figure 13-22: Custom Execution Dialog Box

For step-by-step instructions on using this dialog box, see Adding Pre-Launch and Post-Exit Scripts.

Diagnostic Tools Dialog Box

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Profile Information** page, you can choose to include the Windows Command Prompt and Registry Editor diagnostic tools with your Citrix profile.

If you include diagnostic tools with your Citrix profile, you will be able to look at the registry or file system for the application while it is running in its isolation environment. For example, if you were running a Citrix profile and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the Citrix profile. However, these tools may not be compatible with other operating systems.*

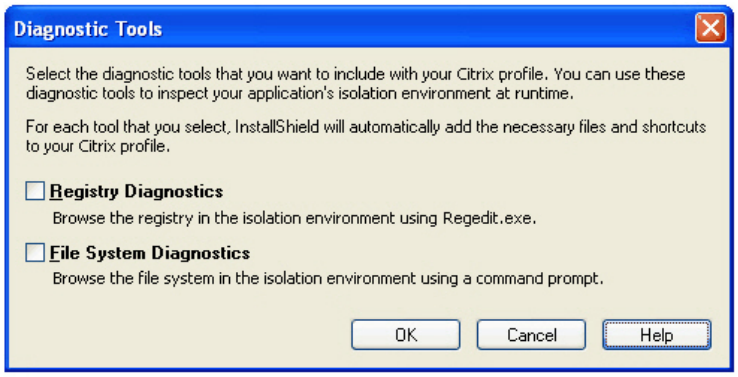


Figure 13-23: Diagnostic Tools Dialog Box

You can use these diagnostic tools to inspect your application's isolation environment at runtime. You have the following options:

Table 13-36 • Diagnostic Tools Dialog Box Options

Option	Description
Registry Diagnostics	Select this option if you want to include `regedit.exe` with your Citrix profile so that you can browse the profile registry.
File System Diagnostics	Select this option if you want to be able to browse the Citrix profile's isolation environment file system using a command prompt.

Launching the Diagnostic Tools Within the Isolation Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the profile.

When the user runs this Citrix profile application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's Citrix isolation environment.

File Isolation Options Dialog Box

On the **File Isolation Options** dialog box, you can override the default Citrix isolation option for the selected file.

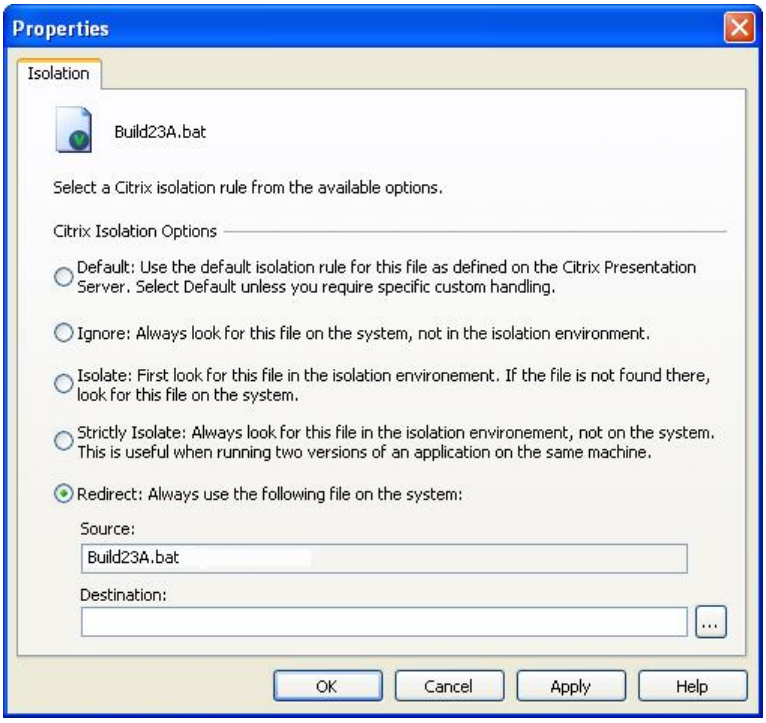


Figure 13-24: File Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The File Isolation Options dialog box includes the following options;

Table 13-37 • File Isolation Options

Option	Description
Default	Use the default isolation option for this file as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this file on the system, not in the isolation environment.
Isolate	First look for this file in the isolation environment. If the file is not found there, look for this file on the system.
Strictly Isolate	Always look for this file in the isolation environment, not on the system. ***Note*** • *This is useful when running two versions of an application on the same machine.*

Table 13-37 • File Isolation Options

Option	Description
Redirect	Always use the following file on the system, not the one in the isolation environment. If you select this option, also specify the following: • **Source**—Name of the selected file. • **Destination**—Select the file on the system that you want the application to use instead of the selected file.

Folder Isolation Options Dialog Box

On the **Folder Isolation Options** dialog box, you can override the default Citrix isolation option for the selected folder.

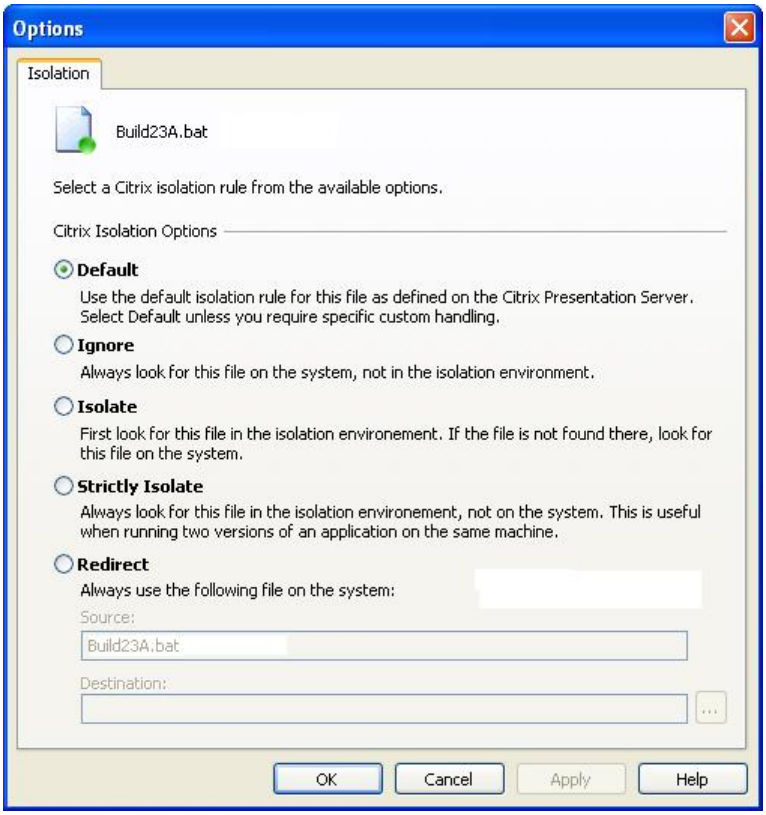


Figure 13-25: Folder Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The **Folder Isolation Options** dialog box includes the following options;

Table 13-38 • Folder Isolation Options

Option	Description
Default	Use the default isolation option for this file as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this file on the system, not in the isolation environment.
Isolate	Look for this folder in both the isolation environment and on the system. If the folder exists in both places, list both in the search results.
Strictly Isolate	Always look for this folder in the isolation environment, not on the system. ***Note*** • *This is useful when running two versions of an application on the same machine.*
Redirect	Always look in the following folder on the system, not in the one in the isolation environment. If you select this option, also specify the following: • **Source**—Name of the selected folder. • **Destination**—The directory on the system where you want the application to look instead of looking in the selected folder in the isolation environment.

Registry Isolation Options Dialog Box

On the **Registry Isolation Options** dialog box, you can override the default Citrix isolation option for the selected registry key.

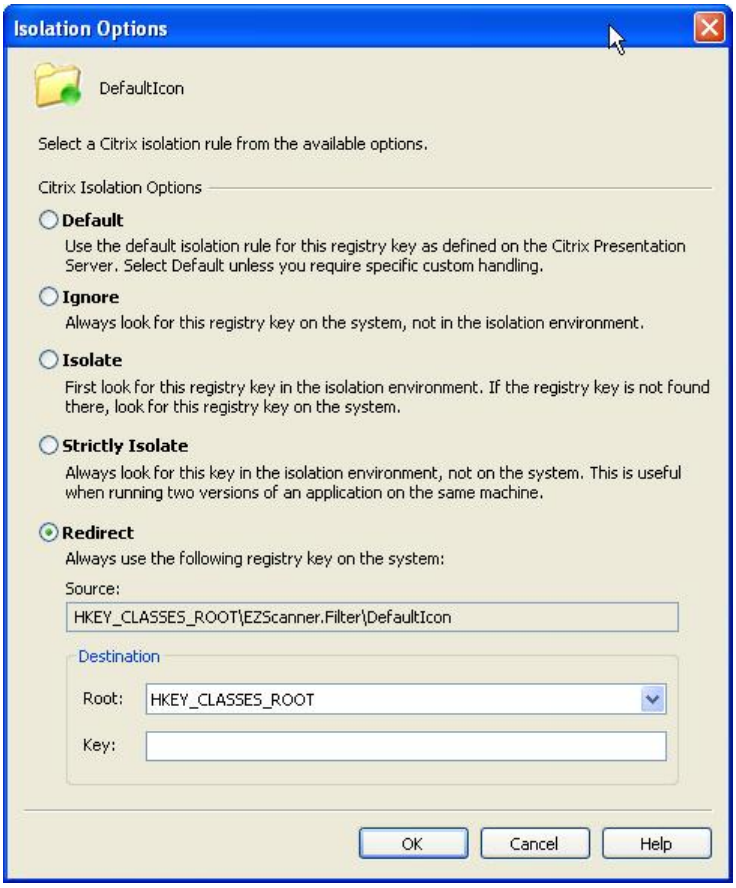


Figure 13-26: Registry Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The Registry Isolation Options dialog box includes the following options;

Table 13-39 • Registry Isolation Options

Option	Description
Default	Use the default isolation option for this registry key as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this registry key on the system, not in the isolation environment.
Isolate	Look for this registry key in both the isolation environment and on the system. If the registry key exists in both places, list both in the search results.

Table 13-39 • Registry Isolation Options (cont.)

Option	Description
Strictly Isolate	Always look for this registry key in the isolation environment, not on the system. ***Note*** *• This is useful when running two versions of an application on the same machine.*
Redirect	Always use the following registry key on the system. If you select this option, also specify the following: • **Source**—Lists the name of the selected registry key. • **Destination Root**—Select the registry root of the registry key on the system that you want to redirect to. • **Destination Key**—Select the registry key on the system that you want to redirect to.

Service Packs Requirement Dialog Box

The **Service Packs Requirement** dialog box is opened by selecting an operating system on the **Profile Requirements** page of the Citrix Assistant and selecting **Service Packs Requirement** from the context menu.

On this dialog box, you can specify which, if any, Service Packs are required for the application to run on the selected operating system

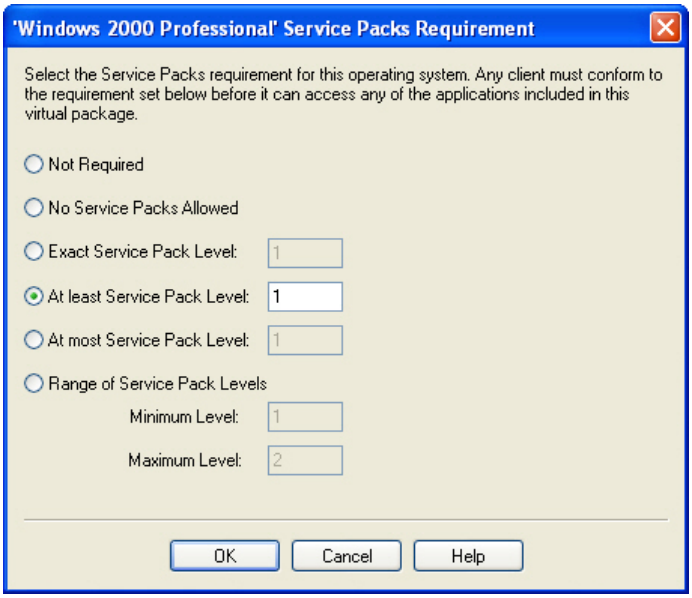


Figure 13-27: Service Packs Requirements Dialog Box

The **Service Packs Requirement** dialog box includes the following options:

Table 13-40 • Service Packs Requirement Options

Option	Description
No Service Pack Requirement	Select this option if this application supports all versions of this operating system, regardless of the number of Service Packs installed.
No Service Pack Allowed	Select this option if this application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly.
Exact Service Pack Level	Select this option if this application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box.
At Least Service Pack Level	Select this option if, to run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box.
At Most Service Pack Level	Select this option if, to run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box.
Range of Service Pack Levels	Select this option if, to run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the **Minimum Level** and **Maximum Level** in the boxes.

Building Citrix Profiles Using the Command Line

When you configure a Citrix profile in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the Citrix profile are built. When you use the standard InstallShield command line build, you do not need to add any additional command line parameters. All of the Citrix profile settings are saved within the InstallShield project.

Citrix Profile Conversion Error and Warning Messages

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a Citrix profile. Therefore, some additional pre- or post-conversion actions must be taken in order for the application profile to run on Citrix XenApp.

One action you could take to try to include ignored features in a Citrix profile is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a Citrix profile.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

Creating ThinApp Applications

Important • *ThinApp support requires a separate purchase of VMware® ThinApp™.*

ThinApp (formerly Thinstall Virtualization Suite) is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine.

You can use the InstallShield **ThinApp Assistant** to configure and build a ThinApp application. Information about creating ThinApp applications using the ThinApp Assistant is presented in the following sections:

- Overview of the Citrix Assistant
- Using the Microsoft App-V Assistant to Create an App-V Package
- Microsoft App-V Assistant Reference
- ThinApp Application Configuration File: package.ini

Overview of the ThinApp Assistant

A ThinApp application is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine. You can use the InstallShield **ThinApp Assistant** to configure and build a ThinApp application.

- About Microsoft Application Virtualization (App-V) and the Microsoft App-V Assistant
  - The ThinApp Virtual Operating System
  - Benefits of Deploying Citrix Profiles
- About the ThinApp Assistant
  - Process for Authoring an App-V Package Using the Microsoft App-V Assistant
  - Components of an App-V Package
  - Supported InstallShield Project Types
  - How Transforms are Included in an App-V Package
  - About Sandboxes

About ThinApp Applications

ThinApp applications can be deployed on a machine without modifying the local operating system or file system. They run in a "sandbox" (or virtual environment) which protects the local operating system from installation modifications that could affect stability or security. Also, ThinApp applications can be run safely from restricted user accounts without local installation.

Information about ThinApp applications is presented in the following sections:

- The ThinApp Virtual Operating System
- Benefits of Deploying Citrix Profiles

The ThinApp Virtual Operating System

A ThinApp application runs in a virtual operating system—a small light-weight component which is embedded with each ThinApp application—that consists of a virtual file system and a virtual registry. When the ThinApp application is run, the virtual operating system environment is merged with the real system environment.

The virtual operating system technology enables entire applications to be packaged into a single `.exe` file that can be run without an installation process, and without modifying the resident operating system.

A ThinApp application can be run from a network or offline on the local machine.

Benefits of Deploying ThinApp Applications

Deploying ThinApp applications provides the following benefits:

- **Reduces time to deployment and costs associated with testing**—Applications can be deployed and run in independent sandboxes, eliminating the need for expensive and time-consuming multi-application regression testing. This reduces the time to deployment and the costs associated with testing.
- **Fast, lightweight virtualization** —ThinApp does not use emulation, so all processes are executed natively at full speed.
- **Reduces the cost of maintaining secure locked-down desktops**—ThinApp applications can run in restricted user accounts without requiring any host modifications.
- **Enhances work-force mobility, business continuity and disaster recovery**—ThinApp applications can be run off-line, directly from any external media including USB Flash, CD-ROM, and off-line laptops.
- **No infrastructure changes needed**—ThinApp applications can be deployed using any existing software deployment systems including Active Directory and SMS. ThinApp has no client or server components to manage or maintain and ThinApp can transparently stream large applications from any network attached storage devices without server software.
- **Sandboxing prevents modifications**—ThinApp redirects all changes intended for the host computer's file system and registry to a private per-user sandbox. Sandboxes can be located on a network share, allowing application settings to follow users as they move from machine to machine. For mobile users, sandboxes can be stored on local USB flash drives, thus preventing damage to the host computer or accidental host storage of sensitive data.

About the ThinApp Assistant

Information about the ThinApp Assistant is organized into the following sections:

- Process for Authoring an App-V Package Using the Microsoft App-V Assistant
- Components of an App-V Package
- Supported InstallShield Project Types
- How Transforms are Included in an App-V Package
- About Sandboxes

Process for Authoring a ThinApp Application Using the ThinApp Assistant

You can use the ThinApp Assistant to convert a Windows Installer package into a ThinApp application. During this process, you:

- **General Settings**—Specify sandbox and Active Directory settings.
- **Files, Folders, Shortcuts, Registry Settings**—Specify the files, folders, shortcuts, and registry settings that will be included in the ThinApp application.
- **Isolation Options**—Override the default isolation options for selected folders and registry keys.
- **Build**—Specify build options and build a ThinApp application.

The following diagram illustrates the ThinApp application creation process:

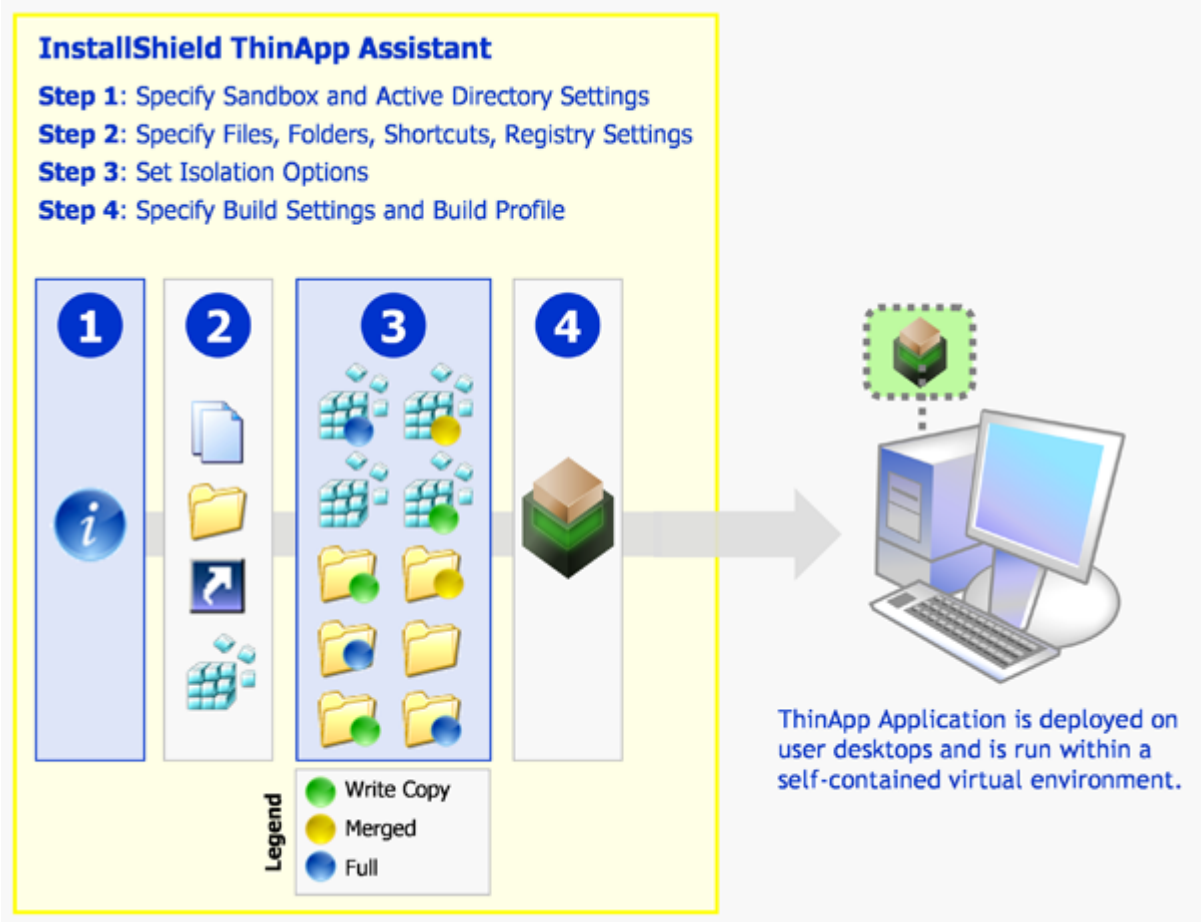


Figure 13-28: Creating a ThinApp Application

The process for authoring a ThinApp application using the ThinApp Assistant is as follows:

Table 13-41 • Steps to Convert a Windows Installer Package to a ThinApp Application

Step	Go To:	Actions
Getting Started	**InstallShield Start Page**	Create or open one of the following project types: • Basic MSI • MSI Database (Direct Edit Mode) • Transform (Direct MST Mode)
	InstallShield Start Page	Click on the **VMware ThinApp** tab to open the **ThinApp Assistant Home** page.
	ThinApp Assistant Home Page	Click **General Settings** in the navigation bar to open the **General Settings** page.

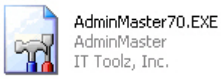
Table 13-41 • Steps to Convert a Windows Installer Package to a ThinApp Application

Step	Go To:	Actions
Specifying Package Information and Deployment Options	**General Settings Page**	Specify the sandbox name and sandbox options for the ThinApp application, control access to the ThinApp application via Active Directory, and specify whether to include diagnostic tools with the ThinApp application.
Managing Files in an App-V Package	**Files & Folders Page**	View existing files and folders, add and delete files.
Setting Isolation Options	**Files & Folders Page**	Override the default isolation options for selected folders. Isolation options specify how the virtual environment will provide access to folders requested by the ThinApp application.
Modifying Shortcuts to the App-V Package's Executable Files	**Applications Page**	Create, delete, include, exclude, or rename ThinApp application executables, which are derived from the shortcuts in its Windows Installer package.
Modifying App-V Package Registry Settings	**Registry Page**	Add, delete, or modify the registry settings in your ThinApp application, and override the default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the ThinApp application.
Modifying Build Options	**Build Options Page**	[Basic MSI Project mode] Select the releases that you want to build. [Direct Edit or Direct MST mode] To enable the Build function for a ThinApp application, select the **Build ThinApp application** option.
Building an App-V Package	**Build on the Toolbar** OR **Build Virtual Package Button**	Click **Build** to build the active Release and create a ThinApp application. When you are in Direct Edit mode, click the **Build Virtual Package** button to save the Windows Installer package and create a ThinApp application.

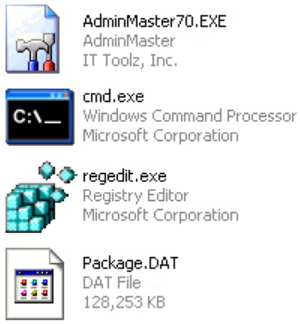
Components of a ThinApp Application

When you use the ThinApp Assistant to build a ThinApp virtual package, the resources you generate are called ThinApp applications. The number of files included in a ThinApp application depends upon how many shortcuts are defined in the project (or Windows Installer package) and whether you choose to include diagnostic tools with the ThinApp application.

Table 13-42 • Components of a ThinApp Application

Number of Shortcuts	ThinApp Application Components	Description
1 shortcut	`ProductName.exe` 	The ThinApp application consists of a single executable (`.exe`) file: • **Launching the application**—This executable file is used to launch the ThinApp application. • **Location of application data**—This executable file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

More than 1 shortcut	`ProductName.exe`
	`FeatureName.exe`
	`Package.DAT`
	
		The ThinApp application consists of two or more executable files and a `Package.DAT` file:
		• **Launching the application**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

1 shortcut with diagnostic tools	`ProductName.exe`
	`cmd.exe`
	`regedit.exe`
	`Package.DAT`
	 
		The ThinApp application consists of three executable files and a `Package.DAT` file:
		• **Launching the application**—The package executable is used to launch the ThinApp application.
		• **Launching the diagnostic tools**—The `cmd.exe` and regedit.exe executables are used to launch the Command Prompt and Registry Editor diagnostic tools.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the ThinApp application and the diagnostic tools to run.

ThinApp application files are saved in a directory named `ThinAppPackage`. The location of the `ThinAppPackage` directory depends upon the type of file you are editing in InstallShield:

- **InstallShield project**—The `ThinAppPackage` directory will be located in a subdirectory of the directory that contains this InstallShield project file, such as:

  `C:\InstallShield 2009 Projects\ProductName\ConfigurationName\ReleaseName\ThinAppPackage`

- **Windows Installer package**—The `ThinAppPackage` directory will be located in the same directory as the Windows Installer file, such as:

  `C:\FolderContainingMSI\ThinAppPackage`

Caution • *Modifying these files directly is* **not recommended**. *To make any modifications, use the InstallShield ThinApp Assistant.*

Intermediate Data Files: Interm Directory

When a ThinApp application is built, files that support the ThinApp application build process are extracted out of the Windows Installer package and saved in a subdirectory of the `ThinAppPackage` directory named the `Interm` directory.

Figure 13-29: Interm Subdirectory of the ThinAppPackage Directory

The data in this directory is then compiled into ThinApp application as part of the build process. The data in the `Interm` directory *does not* need to be distributed with the ThinApp application.

Supported InstallShield Project Types

The **VMware ThinApp** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in a ThinApp Application

The ThinApp Assistant supports the inclusion of transform files with Windows Installer packages in a ThinApp application.

- **How transforms are applied when building a ThinApp application**—When building a ThinApp application, transforms that you have specified are automatically applied to the base Windows Installer (`.msi`) package to create a temporary package, and then the ThinApp application is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build a ThinApp application from that transform file. When you create a new transform file in InstallShield, you specify the root `.msi` file in the **Open Transform** wizard. The steps you take to generate a ThinApp application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the ThinApp application, you need to open one of the *transforms* in InstallShield (rather than the `.msi` file). The **Open Transform** wizard will open, and you will be prompted to specify the root `.msi` file and which of the existing `.mst` files you want to include. The steps you

take to generate a ThinApp application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to a ThinApp application must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the ThinApp application is built.*

About Sandboxes

A ThinApp application runs in a *sandbox*, a virtual operating system—consisting of a virtual file system and a virtual registry—which is embedded with each ThinApp application. Running an application in a sandbox protects the local operating system from installation modifications that could affect stability or security. In a sandbox, system resources (such as files and registry keys) are redirected from the physical operating system files to the sandbox.

Many applications fail to run if the user does not have administrative rights because they expect to be able to write to global locations like `HKEY_LOCAL_MACHINE` and `C:\Program Files`. Using sandbox technology makes applications believe they have the ability to make global changes when they are actually writing to user and application-specific locations, and allows applications that require administrative rights to run without additional privileges. This feature allows ThinApp applications to run in security-restricted environments such as Windows Vista and Terminal Server.

What is a Sandbox Cache?

When you run a ThinApp application, additional files or registry keys may be produced. Depending upon the isolation options, some of this run time data will need to be stored locally in a sandbox cache, a local per-user directory.

When the ThinApp application is built, the *local Sandbox cache* is created in the following location, using the **Sandbox Name** that was entered on the **General Settings** page.

`c:\Documents & Settings\`*`USER_NAME`*`\Application Data\ThinApp\`*`SANDBOX_NAME`*

If the user's `Application Data` directory is stored on a network share, the ThinApp application's settings will automatically migrate when the same user logs in on another machine. You can also choose to create the sandbox cache in an external storage device such as a USB flash drive.

Using the ThinApp Assistant to Create a ThinApp Application

The steps you need to take to create a ThinApp application are the following:

Table 13-43 • Steps to Take to Create a ThinApp Application Using the ThinApp Assistant

Step #	Description
Step 1	Specifying Package Information and Deployment Options
Step 2	Managing Files in an App-V Package
Step 3	Setting Isolation Options
Step 4	Modifying Shortcuts to the App-V Package's Executable Files

Table 13-43 • Steps to Take to Create a ThinApp Application Using the ThinApp Assistant

Step #	Description
Step 5	Modifying App-V Package Registry Settings
Step 6	Modifying Build Options
Step 7	Building an App-V Package

Specifying ThinApp General Settings

When creating a ThinApp application, you can, optionally, specify sandbox and Active Directory settings. You can also specify whether to include diagnostic tools with the ThinApp application. The following tasks are performed on the **General Settings** page of the **ThinApp Assistant**:

- Specifying Sandbox Information
- Specifying Control Access via Active Directory
- Prerequisites for Building a ThinApp Application
- Including Diagnostic Tools in an App-V Package

Specifying Sandbox Information

In this step, you have the option of entering a name for the Sandbox cache that is created when the ThinApp application is built.

Note • *For information on sandboxes and sandbox caches, see About Sandboxes.*

Task ***To specify sandbox information:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. When a ThinApp application is built, a Sandbox cache is created in the following location:

   `c:\Documents & Settings\`*`USER_NAME`*`\Application Data\ThinApp\`*`SANDBOX_NAME`*

   By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the **Sandbox Name** field.

   Note • *The **Sandbox Name** you enter here is also recorded in the* `Package.ini` *file.*
3. If you want changes for Network mapped drives to be saved in the sandbox, then select the **Mapped Network Drive Changes go to Sandbox** option.
4. If you want changes for removable disks to be saved in the sandbox, then select the **Removable Disk Changes go to Sandbox** option.

5. If you want to delete the sandbox content when the ThinApp application exits, then select the **Reset Sandbox on Exit** option.

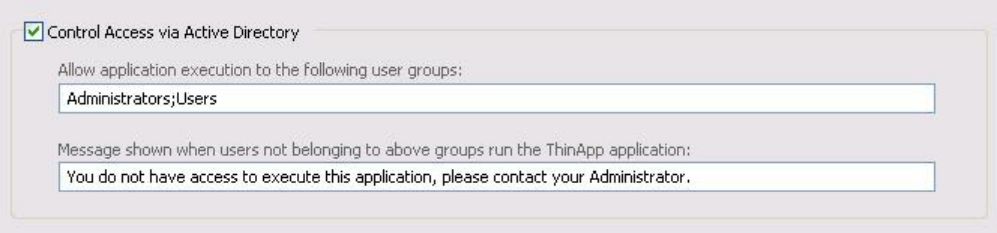
Specifying Control Access via Active Directory

You can control the access of users to a ThinApp application by specifying Active Directory groups on the **General Settings** page. At build-time, ThinApp assigns a unique GUID-like number to uniquely identify each Active Directory Group that you identify. Members of those groups will have access to the ThinApp application. For more detailed information about how Active Directory permissions are assigned, see About Controlling Access to ThinApp Applications.

To specify control access via Active Directory on the **General Settings** page, perform the following steps:

Task ***To specify control access via Active Directory:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. Select the **Control Access via Active Directory** option. The fields below are enabled.



3. In the **Allow application execution to the following user groups** field, enter the names of all of the Active Directory groups that you want to have permission to run this ThinApp application, separated by semi-colons, such as:

   ```
   GroupOne;GroupTwo;GroupThree
   ```

4. In the **Message shown when users not belonging to above groups run the ThinApp application**, enter the message that will be displayed when users that do not belong to the specified groups attempt to launch a ThinApp application.

Caution • *If you do not select the **Control Access via Active Directory** option, anyone who has access to a directory containing a ThinApp application will be able to run the application.*

About Controlling Access to ThinApp Applications

Note the following about controlling access to ThinApp applications via Active Directory:

- **You must be connected**—You must be connected to your Active Directory domain when you build the ThinApp application.
- **Groups must exist**—The Active Directory groups that you specify must exist when the ThinApp application is built.
- **If you delete a group and then recreate it, you must rebuild**—If you delete a group and recreate it, you will need to rebuild the ThinApp application in order to authenticate against the “new” group.

- **Offline users can authenticate using cached credentials**—When users are offline, they can authenticate using cached credentials. Assuming that the user can log into their laptop, ThinApp Active Directory authentication will still work.
- **Sometimes you may need to update credentials manually**—Cached credentials may not refresh on clients until the next Active Directory refresh cycle. To manually refresh the cached group policy credentials, you can use the `gpupdate` command. Sometimes the user may need to log-off before the credentials are recached.
- **"Administrators" and "Everyone" Groups use same credentials**—Special groups like Administrators and Everyone have the same SID on every Active Directory domain and Workgroup. Other groups you create will have a domain-specific SID, meaning a user cannot create their own local group with the same name to bypass authentication.

Prerequisites for Building a ThinApp Application

AdminStudio will convert the package installation into a format compatible with ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools.

As a prerequisite to building a ThinApp application from AdminStudio, you must have installed ThinApp and accepted any and all license agreements.

Note • *For more information, see ThinApp on the VMware Web site.*

Including Diagnostic Tools With a ThinApp Application

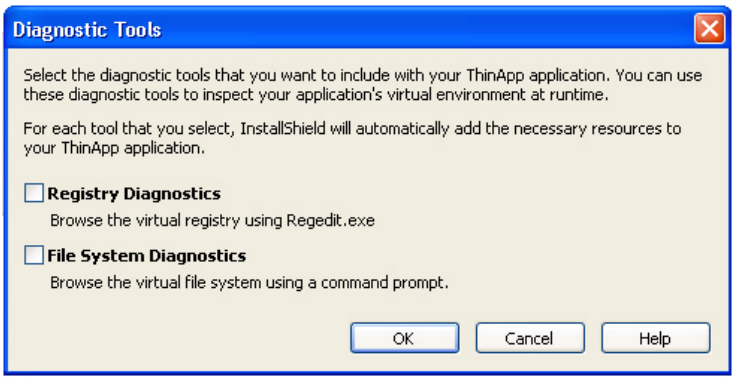
On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **General Settings** page, you can choose to include the Registry Editor and the Windows Command Prompt diagnostic tools with your ThinApp application.

If you include diagnostic tools with your ThinApp application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running a ThinApp application and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the ThinApp application. However, these tools may not be compatible with other operating systems.*

Task ***To include diagnostic tools with a ThinApp application:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. In the **More Options** list, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.



3. If you want to include the Registry Editor (regedit.exe) with your ThinApp application so that you can browse the registry at runtime from within the virtual environment, select the **Registry Diagnostics** option.
4. If you want to include the Windows Command Prompt application with your ThinApp application so that you can browse the virtual file system at runtime from within the virtual environment, select the **File System Diagnostics** option.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the ThinApp application.

When the user runs this ThinApp application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Managing Files and Folders in a ThinApp Application

The next step in creating a ThinApp application is to view existing files and folders, add and delete files and folders, and override the default isolation options for folders.

The following tasks are performed on the **Files & Folders** page.

- Adding, Deleting, and Moving Files and Folders in an App-V Package
- Controlling the Display of Predefined Folders

Adding, Deleting, and Moving Files and Folders in a ThinApp Application

The directories in the destination tree on the **Files & Folders** page of the ThinApp Assistant represent how your application will look when it is installed on to your customer's machine.

On the **Files & Folders** page, you can view all of the files and folders that are currently in your ThinApp application, add new files and folders to include in the ThinApp application, and delete files and folders from the ThinApp application.

- Adding Files to a Citrix Profile

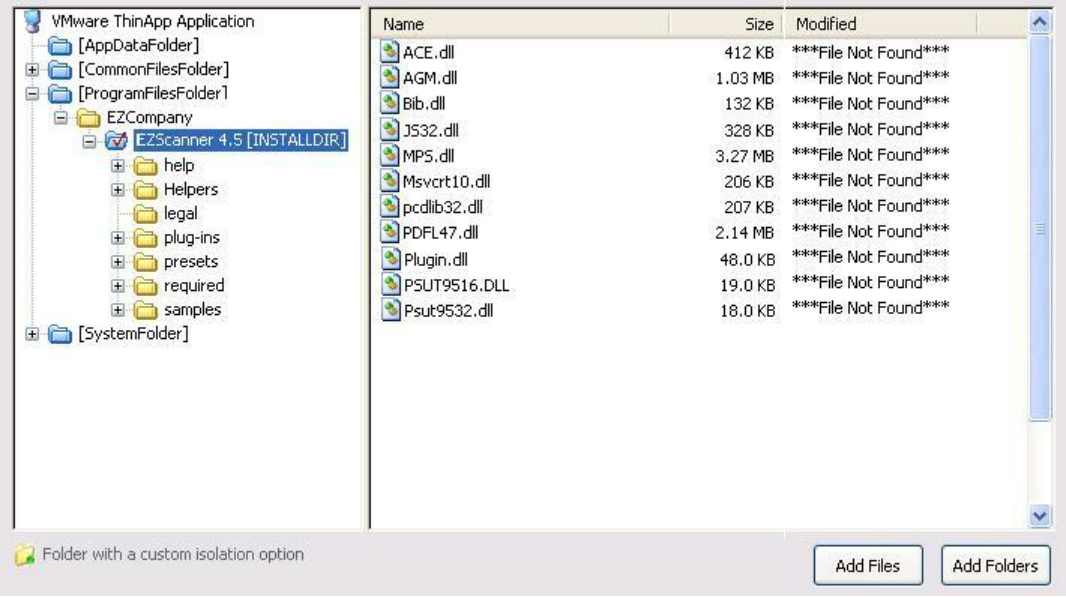
- Adding an Existing Folder (and its Contents) to a Citrix Profile
- Creating a New Folder
- Moving Files and Folders
- Deleting Files and Folders

Adding Files to a ThinApp Application

To add files to a ThinApp application, perform the following steps:

Task ***To add a files to a ThinApp application:***

1. In the **ThinApp Assistant**, open the **Files & Folders** page. The files and folders are listed in the **VMware ThinApp Application** tree, organized by installation directory.



   Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files From Your System

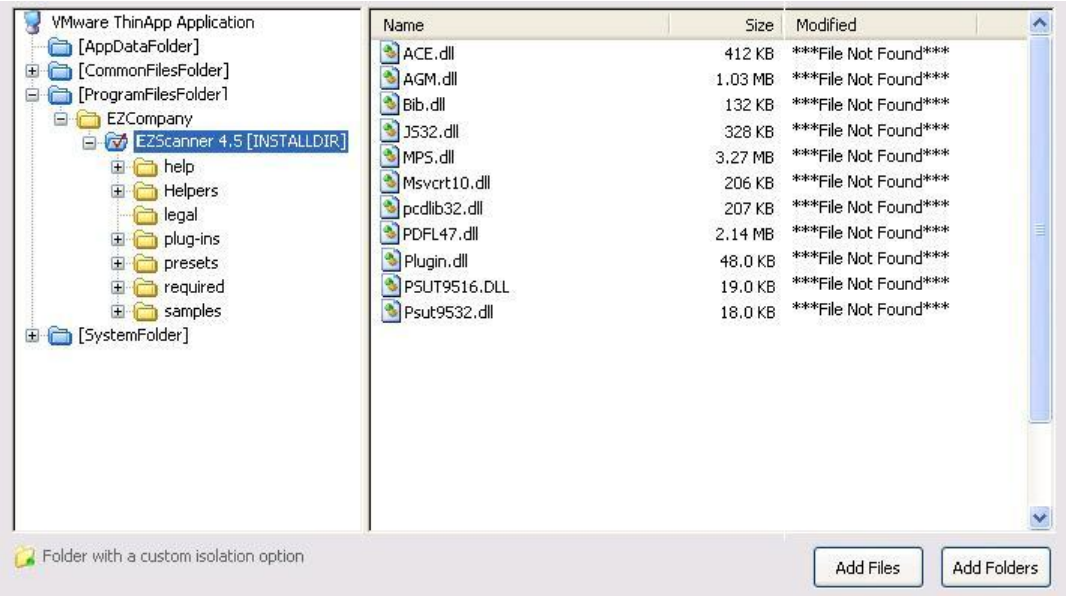
You can also add files or folders to your ThinApp application on the **Files & Folders** page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and its Contents) to a ThinApp Application

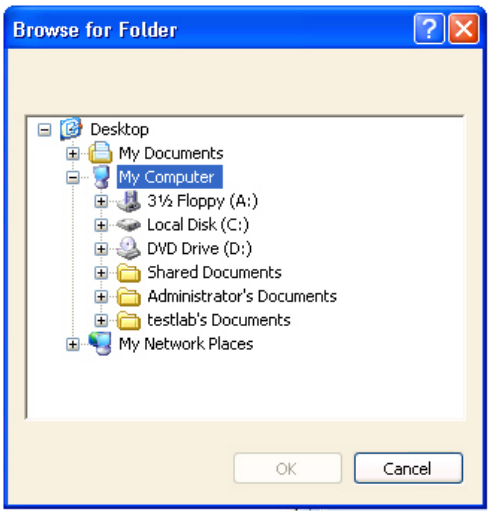
To add an existing folder and all of the files and subfolders within it to a ThinApp application, perform the following steps:

Task ***To add an existing folder to a ThinApp application:***

1. In the **ThinApp Assistant**, open the **Files & Folders** page. The files and folders are listed in the **ThinApp Application** tree, organized by installation directory.

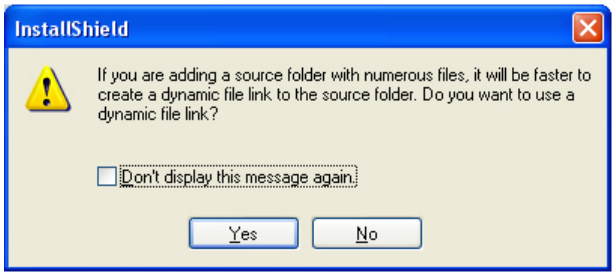


2. Browse through the folder tree to find the folder that you would like to add a folder into.
3. Select the folder and click the **Add Folders** button.The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.



4. Select a folder and click **OK**.

If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.



5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with ThinApp options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

You can create a new, empty folder by selecting an existing folder in the tree and selecting **New Folder** from the context menu.

Task ***To create a new folder:***

1. Right-click on a folder in the **VMware ThinApp Application** tree and select **New Folder**. A new folder is created as a subfolder of the selected folder:

2. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the ThinApp application folder tree structure, perform the following steps:

Task ***To move a file or folder:***

1. Select the file or folder that you want to move.
2. With the mouse button down, drag the file or folder to the new location.

3. Release the mouse button.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from a ThinApp application, perform the following steps:

Task ***To delete a file or folder:***

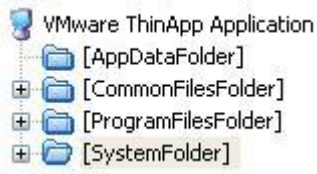
1. Select the file or folder in the **VMware ThinApp Application** tree that you want to delete.
2. Select **Delete** from the context menu. You are prompted to confirm the deletion.
3. Click **Yes**. The selected file or folder is deleted.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire Project, not just from the ThinApp application.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the **Files & Folders** page, the **VMware ThinApp Application** tree initially displays the more commonly used predefined folders, such as `[ProgramFilesFolder]` and `[CommonFilesFolder]`.

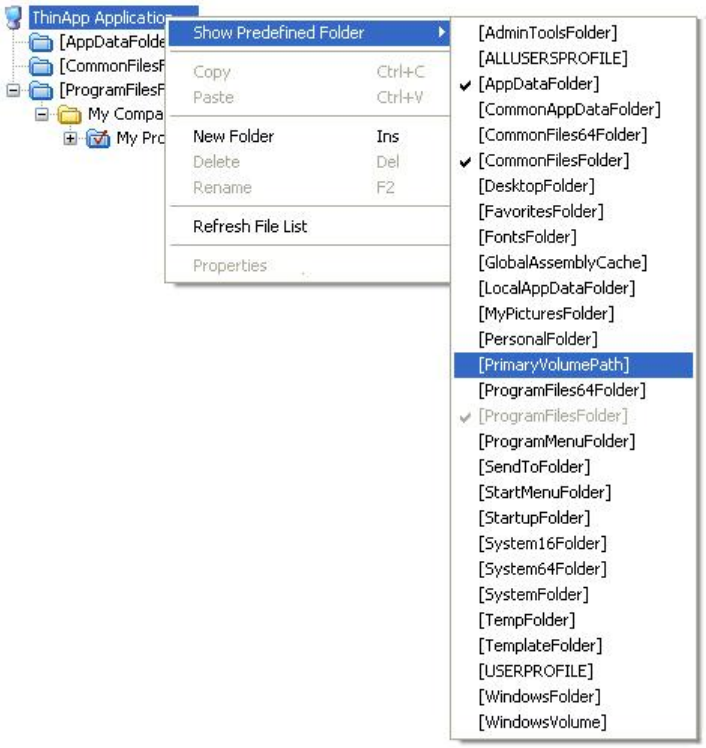


These predefined folders are dynamic, meaning that they do not use hard-coded paths. The value for each destination folder is obtained from the operating system of the target machine.

You can control which predefined folders are listed in this tree.

Task ***To change which predefined folders are listed:***

1. In the **VMware ThinApp Application** tree, select the **ThinApp Application** node (or any of the files or folders that are listed, point to **Show Predefined Folder**. A list of predefined folders opens.



   Those folders that are already displayed are preceded by a check mark, and those that are not displayed do not have a check mark.

2. To add a folder to the tree listing, select a folder that is not currently listed in the tree.

   Note • *These predefined folders are always added to the root of the* ***VMware ThinApp Application*** *tree, no matter what file or folder you had selected when you selected it from the Predefined Folders list.*

3. To remove a folder from the tree listing, select that folder name in this list (which is preceded by a check mark.

   Note • *You cannot turn off the display of the* `[ProgramFilesFolder]`.

Setting ThinApp Isolation Options

ThinApp uses a sandbox virtual environment to control application compatibility and accessibility. The isolation option that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application. You can use isolation options to control what an application can read and write on the local machine.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, in the ThinApp Assistant, you can override the default settings for selected folders or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is open by selecting a folder or registry key and then selecting **Isolation Options** from the context menu.

Information about setting isolation options is presented in the following topics:

- Overview of Citrix Isolation Options
- Setting Isolation Options for Folders and Files
- Inheritance of Isolation Options from Folders to Files

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

Overview of ThinApp Isolation Options

ThinApp uses virtual environments to control application compatibility and accessibility. The *isolation option* that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, in the ThinApp Assistant, you can override the default settings for selected folders or registry keys to exert control over application interactions with client operating system resources.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings. Select the **Default** isolation option unless you require specific custom handling.*

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a folder selected on the **Files & Folders** page or a registry key selected on the **Registry** page.

Information about isolation options is presented in the following sections:

- Available ThinApp Isolation Options
- ThinApp Isolation Option Use Scenarios
- ThinApp Assistant Default Isolation Options

Available ThinApp Isolation Options

On the **Isolation Options** dialog box, you can choose one of the following isolation options:

Table 13-44 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				

Table 13-44 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 13-45 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

ThinApp Assistant Default Isolation Options

If you do not set any isolation options on a folder or registry key in the ThinApp Assistant, the following default isolation options are applied:

Table 13-46 • ThinApp Assistant Default Isolation Options

Isolation Option	Condition
Write Copy Isolation	All other directories and subkeys associated with the product are assigned **Write Copy** isolation.
Merged Isolation	User-specific storage areas like the Desktop and My Documents, are set to Merged Isolation so that application has direct Write access to these locations

Note • *Network shares are not affected by isolation modes. Read and write operations to network shares occur unchanged by ThinApp.*

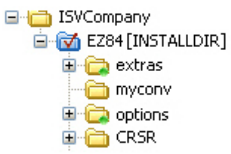
Note • *These default isolation options are built into the ThinApp Assistant.*

Setting Isolation Options for Folders

To override a folder's default isolation options, perform the following steps:

Task ***To set an isolation option on a folder.***

1. Open the **Files & Folders** page.
2. Browse through the folder tree to find the folder that you would like to modify.
3. Select the folder and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.
4. Select one of the following options, as described in ThinApp Isolation Options.
   - Default
   - Write Copy
   - Merged
   - Full
5. Click **OK**. Folders that have an isolation setting other than default are marked with a special icon:



Inheritance of Isolation Options from Folders to Files

Isolation options for files and subfolders are always inherited. The ThinApp virtual environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

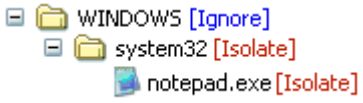


Figure 13-30: Example of Inheritance of Isolation Options from Folders to Files

Modifying Shortcuts to the ThinApp Application's Executable Files

You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment.

By default, the **ThinApp Assistant** creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

When you select each shortcut, details about it are displayed:

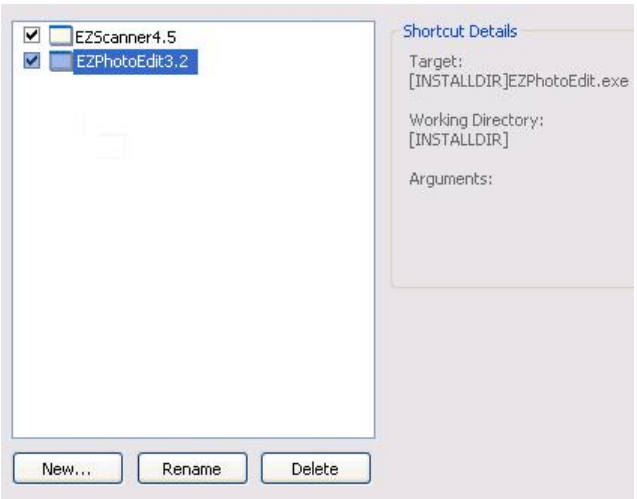


Figure 13-31: List of Shortcuts for an Application

Caution • *You must define at least one shortcut to enable users to launch the application from the isolation environment.*

On the **Applications** page, you can create, delete, include, exclude, or rename ThinApp application executables, which are derived from the shortcuts in its Windows Installer package.

- App-V Packages and the Virtual Environment
- App-V Shortcut Requirements
- Creating a New App-V Package

- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Renaming a Shortcut

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

ThinApp Applications and the Virtual Environment

On the **Applications** page of the ThinApp Assistant, you define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project.

To deploy a ThinApp application—on a local drive or a network share—systems administrators just need to give users access to the ThinApp application.

Compressing a ThinApp Application

A ThinApp application consists of either:

- **One executable file (.exe)**—This file is used to both launch the ThinApp application and also contain all of the data that is required for the application to run. In this scenario, this executable file would be a large file.

  or

- **Several executable files (.exe) and a Package.DAT file**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application and `Package.DAT` contains all of the data that is required for the application to run. In this scenario. `Package.DAT` would be a large file.

Each time a user launches a ThinApp application, its data (from either the executable file or from `Package.DAT`) is read into the computer's memory. To reduce the application size, you can select a **Compression Type** on the **Build Options** page to compress all of the data.

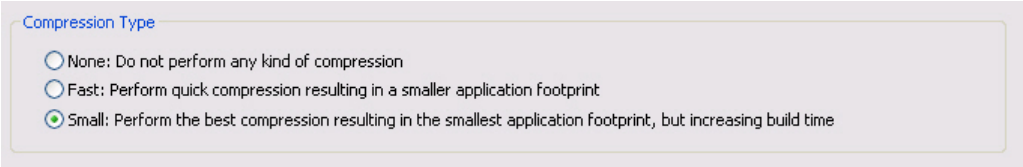


Figure 13-32: Compression Types on the Build Options Page

The following **Compression Types** are available:

Table 13-47 • ThinApp Compression Types

Type	Description
None	Do not perform any type of compression.

Table 13-47 • ThinApp Compression Types

Type	Description
Fast	Perform quick compression resulting in a smaller application footprint.
Small	Perform the best compression resulting in the smallest application footprint, but increasing build time.

Application startup time is most effected by compression options used:

- **No compression**—Without compression enabled, startup speeds are comparable to normal application startup times.
- **Fast compression**—With fast compression options enabled, applications may startup faster than normal when the disk cache is empty and slightly slower than normal when the disk cache has been pre-filled, depending on processor speed and disk speeds.

You may also want to compress a ThinApp application to make it easier to distribute it throughout your organization.

When you perform compressed builds, large temporary files are saved in a cache location. To delete all of these temporary files, select the **Clear the VMware ThinApp Cache** option in the **More Options** list on the **Build Options** page

ThinApp Shortcut Requirements

For each ThinApp application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build a ThinApp application that does not contain any shortcuts, users will not be able to launch the application.

Creating a New ThinApp Application

On the **Applications** page of the ThinApp Assistant, you specify the executables that you want to create ThinApp applications for.

Task ***To create a new ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project (or Windows Installer package) are listed:
   - Those that are currently included in the ThinApp application are selected.
   - Those that are currently excluded from the ThinApp application are not selected.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this ThinApp application.
3. Select the file that you want to create a shortcut to.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.
5. To include this shortcut in the ThinApp application, make sure that its check box is selected.

Including an Existing ThinApp Application

If you want to include a previously excluded shortcut in a ThinApp application, perform the following steps:

Task ***To include an existing ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding or Deleting an Existing ThinApp Application

By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

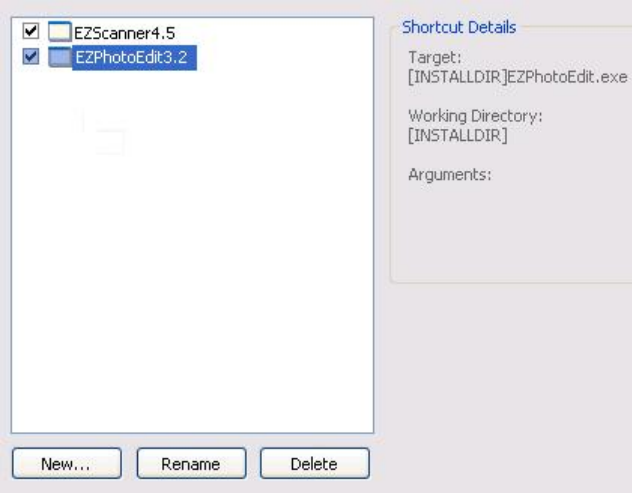


Figure 13-33: Initial List of Shortcuts for an Application

To prevent the shortcut from being created in the ThinApp application, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. See Deleting a Shortcut.

Caution • *If you delete a shortcut on the* ***Applications*** *page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

If you have any unnecessary shortcuts in your project, you can simply exclude them from the ThinApp application by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding a ThinApp Application

If you want to exclude one of these shortcuts from being created in the ThinApp application, perform the following steps:

Task ***To exclude a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To exclude a shortcut, select the shortcut and clear the check box.

Note • *When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project.*

Deleting a ThinApp Application

To delete a ThinApp application, perform the following steps.

Task ***To delete a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut and click **Delete**.

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Excluding vs. Deleting ThinApp Application Shortcuts

To prevent a shortcut from being created in the ThinApp application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Renaming a ThinApp Application

To rename a ThinApp application, perform the following steps:

Task ***To rename a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut that you want to rename and click **Rename**. A box appears around the shortcut name, and the shortcut name becomes an editable field.
3. Enter a new name for the shortcut.

Modifying ThinApp Application Registry Settings

Using the **ThinApp Assistant**, you can view existing registry keys, values, and data, and add or delete registry items in your ThinApp application.

You can also override the default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

Information about modifying registry settings on the **Registry** page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values
- Setting App-V Package Registry Isolation Options

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_USER

- HKEY_CLASSES_ROOT

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the Registry page is performed much like it is performed on the InstallShield Registry view. See Editing the Registry.

Setting ThinApp Isolation Options on Registry Keys

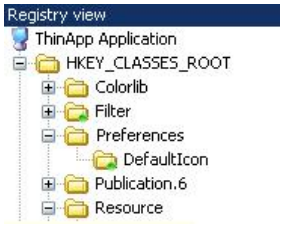
To override a registry key's default isolation options (which are built into the ThinApp Assistant), perform the following steps:

Task ***To set an isolation option on a registry key:***

1. Open the **Registry** page.
2. Browse through the registry tree to find the key that you would like to modify.
3. Select the folder or key and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.

   Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

4. Select one of the following options, as described in ThinApp Isolation Options.
   - Default
   - Write Copy
   - Merged
   - Full
5. Click **OK**. Registry keys that have an isolation setting other than default are marked with a special icon:



***Tip** • To import an existing registry (.reg) file, click the **Import a .reg file** option on the **More Options** list to open the Registry Import Wizard.*

Inheritance of ThinApp Isolation Options in the Registry

Isolation options for registry keys are always inherited. The ThinApp virtual environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

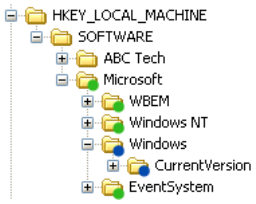


Figure 13-34: Example of Inheritance of Isolation Options from Folders to Files

Modifying Build Options

On the **Build Options** page, you choose which releases of this InstallShield project you want to build a ThinApp application for when the project is built, specify the type of compression, disable the ThinApp Log Monitor tracing capabilities, and specify whether you want to include additional Windows Installer packages in the virtual package.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the **Build ThinApp Application** option on the **Build Options** page before you will be able to build a ThinApp application for that Windows Installer package.

- Selecting the Releases for Which You Want to Build App-V Packages
- Enabling App-V Package Building When in Direct Edit Mode
- Including Additional Windows Installer Packages in a ThinApp Application
- Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application
- Setting ThinApp Log Monitor Tracing Options
- Setting AppLink Options
- Setting AppSync Options

***Important** • You must create at least one Release (on the **Releases** view of the Installation Designer) before you will be able to select a Release on the **Build Options** page.*

Selecting Releases to Build

You select the releases that you want to build a ThinApp application for on the **Releases** tree of the **Build Options** page.

Important • *You cannot create or edit a release in the ThinApp Assistant. If no releases exist, you can simply click the* ***Build*** *toolbar button to create a new release or open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see Creating and Building Releases.*

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the ***Releases*** *tree on the* ***Build Options*** *page is not displayed.*

Task ***To select releases to build:***

1. Open the **Build Options** page.
2. Select the releases in the **Releases** tree that you want to build a ThinApp application for.

   Important • *When you select a release on the* ***Build Options*** *page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected on the* ***Build Options*** *page have no bearing upon which release is built when you click the* ***Build*** *button on the toolbar. When you initiate a build by clicking the* ***Build*** *button, a build is initiated for the* **active** *release—the release that was most recently selected on the Installation Designer* ***Releases*** *view. The output of that build would depend upon what releases were selected on the* ***Build Options*** *page:*

   - **Active release selected**—*A Windows Installer package and a ThinApp application would be built.*
   - **Active release not selected**—*Only a Windows Installer package would be built.*

   Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Enabling ThinApp Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **ThinApp Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a ThinApp application for this Windows Installer package. To do this, perform the following steps:

Task ***To enable ThinApp application building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. Open the **Build Options** page of the ThinApp Assistant.

3. Select the **Build ThinApp application** option. After you select this option, the **Build ThinApp application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Including Additional Windows Installer Packages in a ThinApp Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the ThinApp Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a ThinApp application, set the **Would you like to include additional MSI files in the virtual package?** option on the **Build Options** page to **Yes**, and then select the packages that you want to add.

Task ***To include additional Windows installer packages in a ThinApp application:***

1. Open the **Build Options** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up () and Move Down () buttons.
   - Use the Delete button ( ) to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application

You can choose to build a Windows Installer package to assist in the distribution of a ThinApp application. This simplifies the deployment of a ThinApp application by enabling you to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management.

To build a Windows Installer file with your ThinApp application, select the **Generate a Windows Installer (MSI) file as part of the build output** option on the **Build Options** page of the ThinApp Assistant. By default, this option is not selected.

This Windows Installer file can be run to properly install the ThinApp application on an end-user's desktop. A ThinApp application installed using a Windows Installer package can be uninstalled using **Add or Remove Programs** in the Control Panel.

Setting ThinApp Log Monitor Tracing Options

ThinApp Log Monitor is an application in the ThinApp Suite that allows you to record detailed information about any application's execution history for later review. The following events are recorded:

- **API calls**—Win32 API calls with parameter and result information made by applications running in the ThinApp virtual operating system
- **Errors**—A list of potential errors, exceptions, and security events within the application

- **Loaded DLLs**—A list of all DLLs loaded by the application and address ranges

Log Monitor is launched by selecting a shortcut in the ThinApp Suite group on the Windows Start menu.

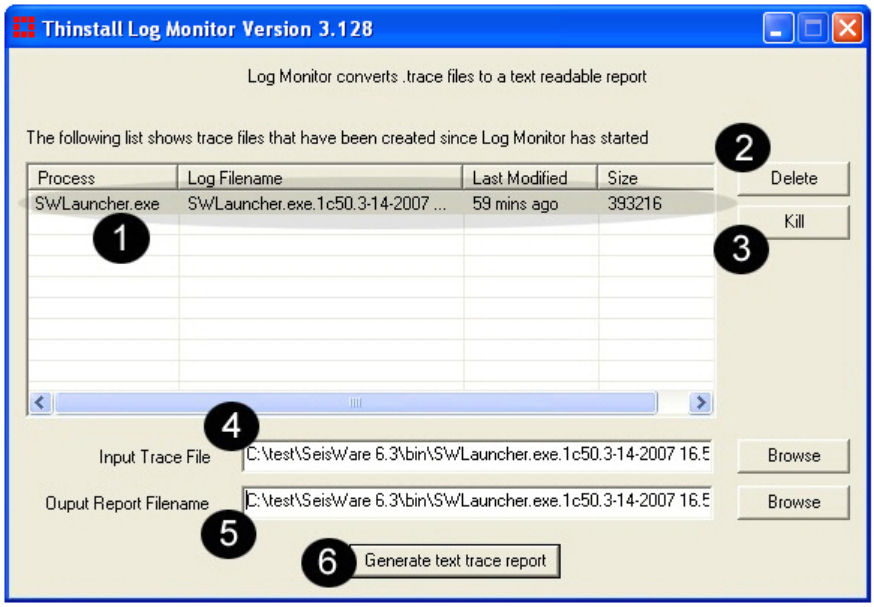


Figure 13-35: ThinApp Log Monitor

Log Monitor displays the following information:

Table 13-48 • ThinApp Log Monitor Interface

#	Name	Description
1	**Process List**	Any new ThinApp process which has been started after Log Monitor begins will be listed. If you click on one of the processes, the **Input Trace File** and **Output Report Filename** fields are automatically populated. ***Note*** • *If the application was built with the **Disable Log Monitor Tracing** option on the **Build Options** page selected, it will not be listed.*
2	**Delete**	Click to delete trace files for the selected processes in the Process List.
3	**Kill**	Click to kill currently running process that is selected in the Process List. You would do this to stop a process from logging additional entries once an error condition has been reached.
4	**Input Trace File**	Click Browse to manually browse for a trace file to convert.
5	**Output Report File**	The file listed in this field is generated when you click **Generate text trace report**. This report should be viewed with a text editor that supports UNIX-style line breaks such as Wordpad or Word (not Notepad).

Disabling Log Monitor Tracing

If you do not want to allow ThinApp Log Monitor tracing in a ThinApp application, select the **Disable Log Monitor Tracing** option on the **Build Options** page.

Task ***To disabling the ThinApp Log Monitor tracing capabilities:***

1. Open the **Build Options** page.
2. Select the **Disable Log Monitor Tracing** option.
3. Build the ThinApp application.

Setting AppLink Options

Note • *The AppLink Settings feature requires ThinApp 4.x. If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. You can set AppLink settings for the current ThinApp application on the **AppLink Settings** dialog box, which is opened by clicking the **AppLink Settings** option in the **More Options** menu of the ThinApp Assistant **Build Options** page.

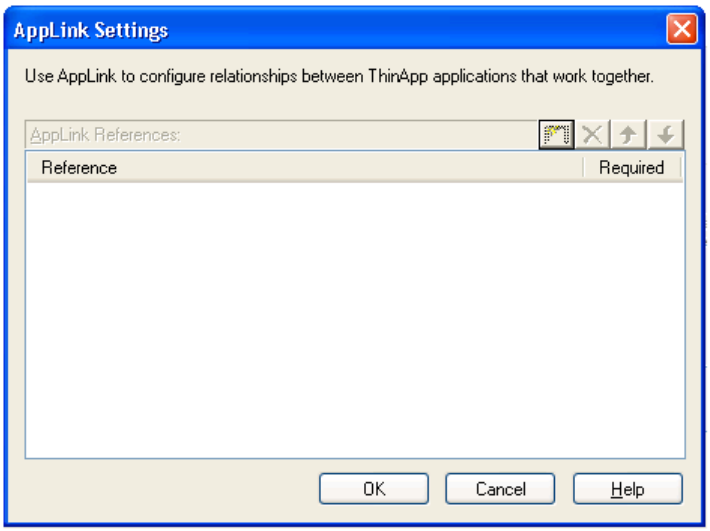
You can use the AppLink feature to perform the following tasks:

- **Linking runtime components to applications**—You can link runtime components to the applications that use them. For example, you can link a package containing the Java runtime environment (JRE) or ODBC drivers to a package containing a browser application.
- **Linking add-ons and plug-ins to applications**—You can link add-ons and plug-ins to applications. For example, Microsoft Office add-ons can be linked to applications or Adobe Photoshop plug-ins can be linked to a package containing Photoshop.
- **Linking packaged applications to service packs**—You can link packaged applications to service packs. By using AppLink, you can upgrade or roll back your service packs by changing the service pack that you capture and link to its parent application.

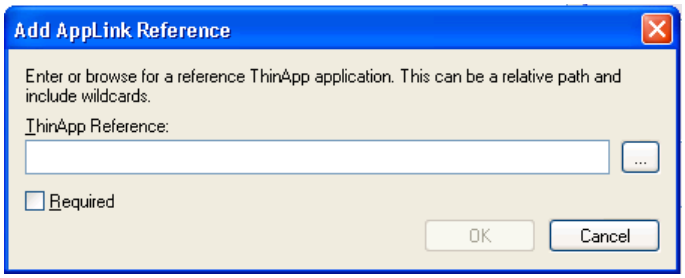
To set AppLink options for a ThinApp application, perform the following steps.

Task ***To configure AppLink settings for your ThinApp application:***

1. On the **Build Settings** page of the ThinApp Assistant, click the **AppLink Settings** option in the **More Options** menu. The **AppLink Settings** dialog box opens.



2. Click the Browse button to open the **Add AppLink Reference** dialog box.



3. In the **ThinApp Reference** box, enter the relative (runtime) path to the existing ThinApp application that you want to link to.
   - If you want to add multiple applications, repeat the procedure as necessary.
   - You can also use wild cards. See Security and Authorization
   - The order in which packages are imported can be changed by selecting a package and clicking the up and down arrows. See Collisions and Order of Import for more information on order.

Note • *Required and Optional links are listed on the AppLink Settings dialog box together and the order can be changed using the up and down arrows. However, at runtime, all of the applications in the Required category are read first, before those in the Optional category, even if applications in the Optional category were originally higher in the list. When the AppLink Settings dialog box is reopened, the AppLink References will be grouped by category rather than be in the order that was arranged prior to closing the dialog box. In other words, the category order (Required and Optional) overrides the order set by the user.*

- To delete a package you have added, select the package and click the Delete ( ) button.

Important • *When linking to a ThinApp application that has only one shortcut, select its .EXE file. When linking to a ThinApp application that has more than one shortcut, select either its Package.DAT file (if the ThinApp application was built with AdminStudio) or its primary executable file (if the ThinApp application was built with ThinApp).*

Important • *On the* ***Add AppLink Reference*** *dialog box, if you click Browse and browse for a ThinApp application, the absolute path to that application is entered, such as* `C:\Program Files\AppName\filename.exe`. *In that case, the parent ThinApp application needs that linked application to be found at the specified absolute path location at runtime, which is unlikely. Therefore, it is recommended that you enter a relative path name.*

4. If you want this package to be required, select the **Required** option. If a required package is missing from the virtual package, it will fail to run. Note the following about required packages:
   - If any specified package fails to import, an error message will be displayed and the parent executable file will exit.
   - If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
   - To continue even if load errors occur, make the package references optional instead.
5. Click **OK** to return to the **AppLink References** dialog box. The item you selected is now listed in the **AppLink References** list.
6. Click **OK** to return to the **Build Options** page.

Setting AppSync Options

Note • *The AppSync Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppSync settings that you define will be ignored.*

AppSync (Application Sync) enables you to automatically keep deployed ThinApp applications up to date. When an application starts up, AppSync can query a Web server to see if an updated version of the package is available. If an update is available, the differences between the existing package and the new package will be downloaded and used to construct an updated version of the package. The updated package will be used for future deployments.

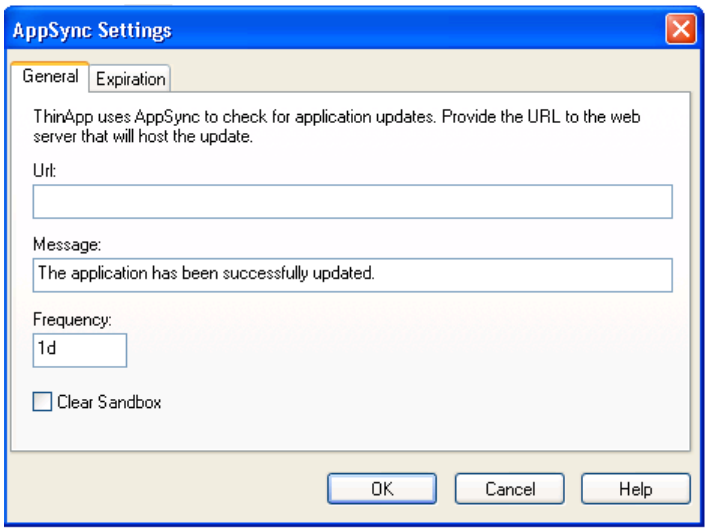
You can use the AppSync feature to perform the following tasks:

- **Distribute runtime components separately**—You can use AppSync to distribute runtime components separately from the applications that use them. For example, the Java Runtime Environment (JRE) or ODBC drivers.
- **Apply layered service packs to applications**—You can use AppSync to apply layered service packs to your applications. Application Sync enables you to distribute service packs and roll back to previous versions, if necessary.

On the AppSync Settings dialog box, you specify the location of the update, the message displayed to the user, and the expiration settings. You set AppSync settings for the current ThinApp application on the **Build Options** page of the ThinApp Assistant. To configure AppSync settings for a ThinApp application, perform the following steps:

Task ***To configure AppSync settings for your ThinApp application,***

1. On the **Build Settings** page, click the **AppSync Settings** option in the **More Options** menu. The **General** tab of the **AppSync Settings** dialog box opens.



2. In the **Url** field, specify the location of the Web server that hosts application updates. When entering the URL, consider the following:

   - **Supports HTTP and HTTPS**—Application Sync works over both the HTTP (unsecure) and HTTPS (secure) protocol.
   - **Can include login information**—You can include a user name and password in the URL that will be used for basic authentication. The standard Windows/Internet Explorer proxy setting is respected. For example:

     ```
     https://www.example.com/some/path/PackageName.exe
     ```

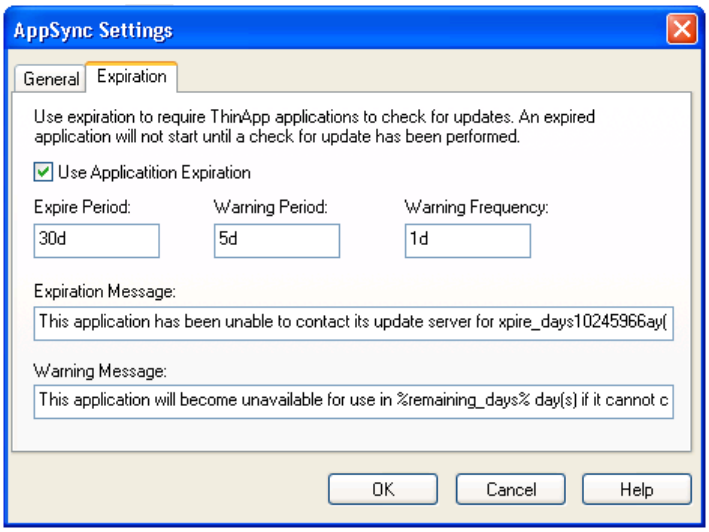
3. In the **Message** field, enter the information you want to display to the user when the ThinApp application is updated. By default, the following is entered:

   ```
   The application has been successfully updated.
   ```

4. By default, a package will connect to the Web server once per day to see if an updated version is available. You can set the frequency by modifying the **Frequency** setting. For example, to set the **Frequency** to 2 days, enter **2d**. For 2 weeks, enter **2w**, etc.

5. If you want to automatically clear the sandbox after an update, select the **Clear Sandbox** option. By default, this option is not selected.

6. Click **Expiration** to open the **Expiration** tab. On this tab, you can specify that a ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run.



7. To require that an application has to check for updates at a specified frequency, select the **Use Application Expiration** option

8. In the **Expire Period** box, enter the update frequency in minutes (`m`), hours (`h`), or days (`d`). For example:

   - To set the period to 30 days, enter **`30d`.**
   - To set the period to 72 hours, enter **72h**.
   - If you do not want the package to expire, clear the **Use Application Expiration** check box.

   If the Web server cannot be reached, meaning that the update fails, the package will continue to work until the **Expire Period** is reached. The default setting is 30 days.

9. In the **Warning Period** box, enter the amount of time prior to expiration that the user is first warned. For example, to set the period at 5 days, enter **5d**.

10. In the **Warning Frequency** box , enter the frequency that a warning message will be displayed to the user before the package expires. With the default of one day, the warning message will be displayed once per day only. To configure the warning to pop up on every application launch, enter **`0`**. To configure it to pop up every 4 days, enter **4d**.

    Note the following about warning frequency:

    - After the warning period has started, the Web server will be checked on every launch of an application, overriding any previous setting.
    - As long as a package has not expired, this parameter checks for new versions and downloads will occur in the background. The user can continue to use the old version.
    - If the application is terminated by the user before the download is complete, the download will resume when a virtual application is launched again. After the download completes, the new version will be activated on the next launch.
    - When the package has expired, the version check and download will happen in the foreground. A progress bar will be shown during the download phase.

11. Before the expiration limit has been reached and a ThinApp application is started, it will try to connect to the Web server and check for a new version. If the connection fails, a message box will be shown. The default message is:

```
This application will become unavailable for use in Warning_Period days if it cannot contact its
    update server. Check your network connection to ensure uninterrupted service
```

12. After the expiration limit has been reached and a ThinApp application is started, it will try to connect to the Web server and check for a new version. If the connection fails, the message entered in the **Expiration Message** box will be shown and execution will be terminated. The default message is:

```
This application has been unable to contact its update server for Expire_Period days, so it is
    unavailable for use. Check your network connection and try again.
```

Note • *If you use AppSync, VMware recommends that you disable automatic application updates that are configured in your virtual application. Conflicts might occur between the linked packages and the software that is automatically updated. If an automatic update feature updates an application, it stores the updates in the sandbox. If AppSync then updates the application to a different version, the updates stored in the sandbox take precedence over the files contained in the version that AppSync created. The order of precedence for the update files are those in the sandbox, then the virtual operating system, and then the physical machine.*

Building a ThinApp Application

The method for building a ThinApp application depends upon what file you have open—an InstallShield project or a Windows Installer package.

- Building an App-V Package from Within an InstallShield Project
- Building an App-V Package from Within a Windows Installer Package in InstallShield

Building a ThinApp Application for an InstallShield Project

To build a ThinApp application for an InstallShield project, perform the following steps:

Task ***To build a ThinApp application for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

   Important • *You cannot create or edit a release in the ThinApp Assistant. If no releases exist, or if you want to create a new release, open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see Creating and Building Releases.*
3. Open the **Build Options** page of the ThinApp Assistant.
4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build a ThinApp application for.

   Important • *When you select a release on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected*

*on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what was selected on the **Build Options** page:*

- **Active release selected**—*A Windows Installer package and a ThinApp application would be built.*
- **Active release not selected**—*Only a Windows Installer package would be built.*

To build more than one release at a time, perform a batch build. See Performing Batch Builds.

5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

   The output of the build will be a Windows Installer package and a ThinApp application. For information on the files included in a ThinApp application, see Components of an App-V Package.

Building a ThinApp Application for a Windows Installer Package

To build a ThinApp application for a Windows Installer package, perform the following steps:

Task ***To build a ThinApp application for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).
   - On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
   - On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the ThinApp Assistant to set ThinApp application options.
3. Save the edits to the Windows Installer package or transform file by selecting **Save** on the **File** menu.
4. On the **Build Options** page of the ThinApp Assistant, select the **Build ThinApp application** option. The **Build Virtual Package button is enabled.**
5. Click the **Build Virtual Package button** to start building the ThinApp application.

   The output of the build will be a ThinApp application. For information on the files includes in a ThinApp application, see Components of an App-V Package.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

ThinApp Assistant Reference

Reference information about the ThinApp Assistant is organized into the following sections:

- Microsoft App-V Assistant Pages

- Microsoft App-V Assistant Dialog Boxes
- Building App-V Packages Through the Command Line
- Troubleshooting the Builds of App-V Packages
- Application Features that Require Pre- or Post-Conversion Actions

Pages

The ThinApp Assistant is comprised of the following pages:

- Home Page
- Profile Information Page
- Profile Files Page
- Profile Shortcuts Page
- Profile Registry Page
- Build Settings Page

ThinApp Assistant Home Page

The ThinApp Assistant Home page displays a diagram that illustrates the process of creating a ThinApp application.

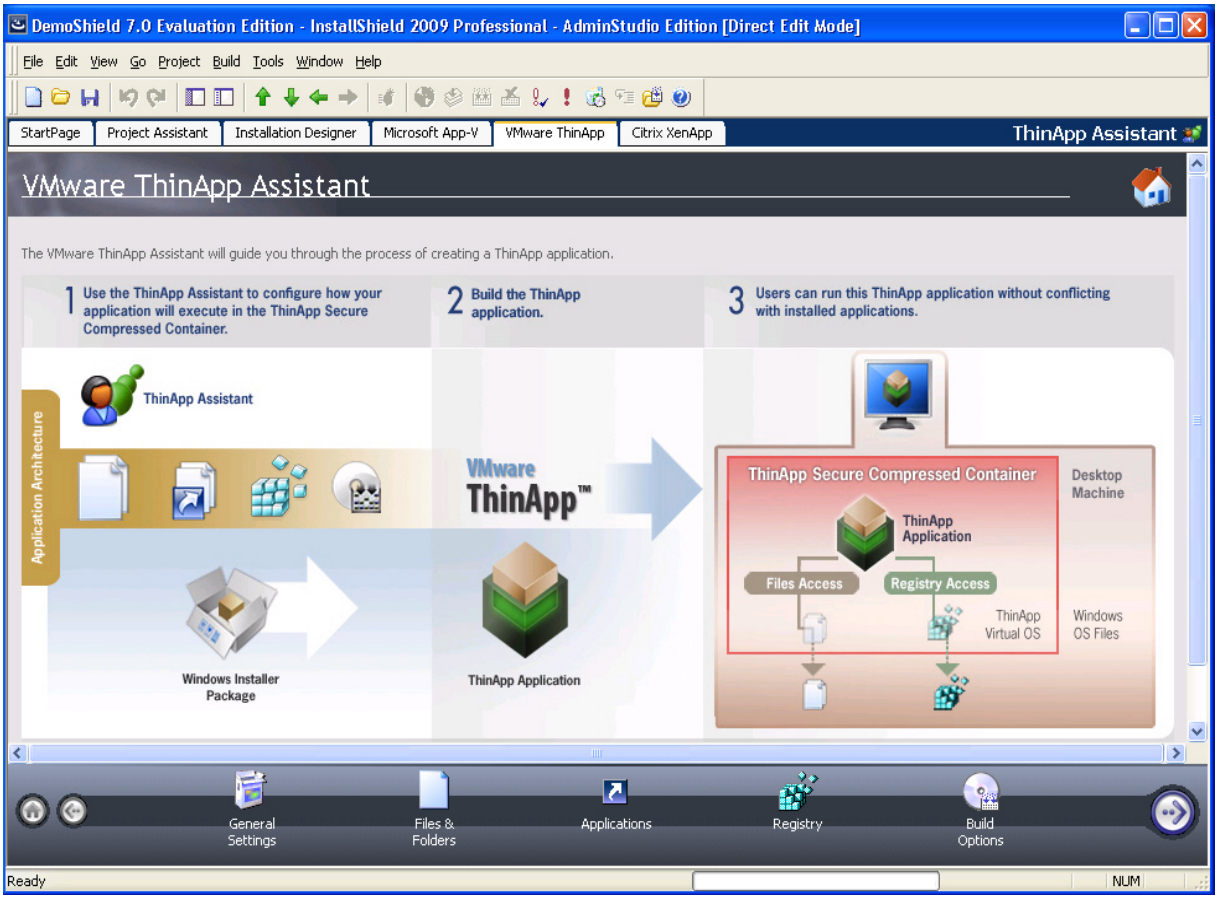


Figure 13-36: ThinApp Assistant Home Page

Click the following icons in the navigation bar at the bottom of the page to navigate through the ThinApp Assistant interface:

Table 13-49 • Navigation Bar Icons

Icon	Destination
	Profile Information Page
	Profile Files Page
	Profile Shortcuts Page
	Profile Registry Page

Table 13-49 • Navigation Bar Icons

Icon	Destination
	Build Settings Page
	Go to next page.
	Jump back to previous page.
	Home Page

General Settings Page

On the **General Settings** page in the **ThinApp Assistant**, you specify Sandbox options, including options to control access to the ThinApp application using Active Directory.

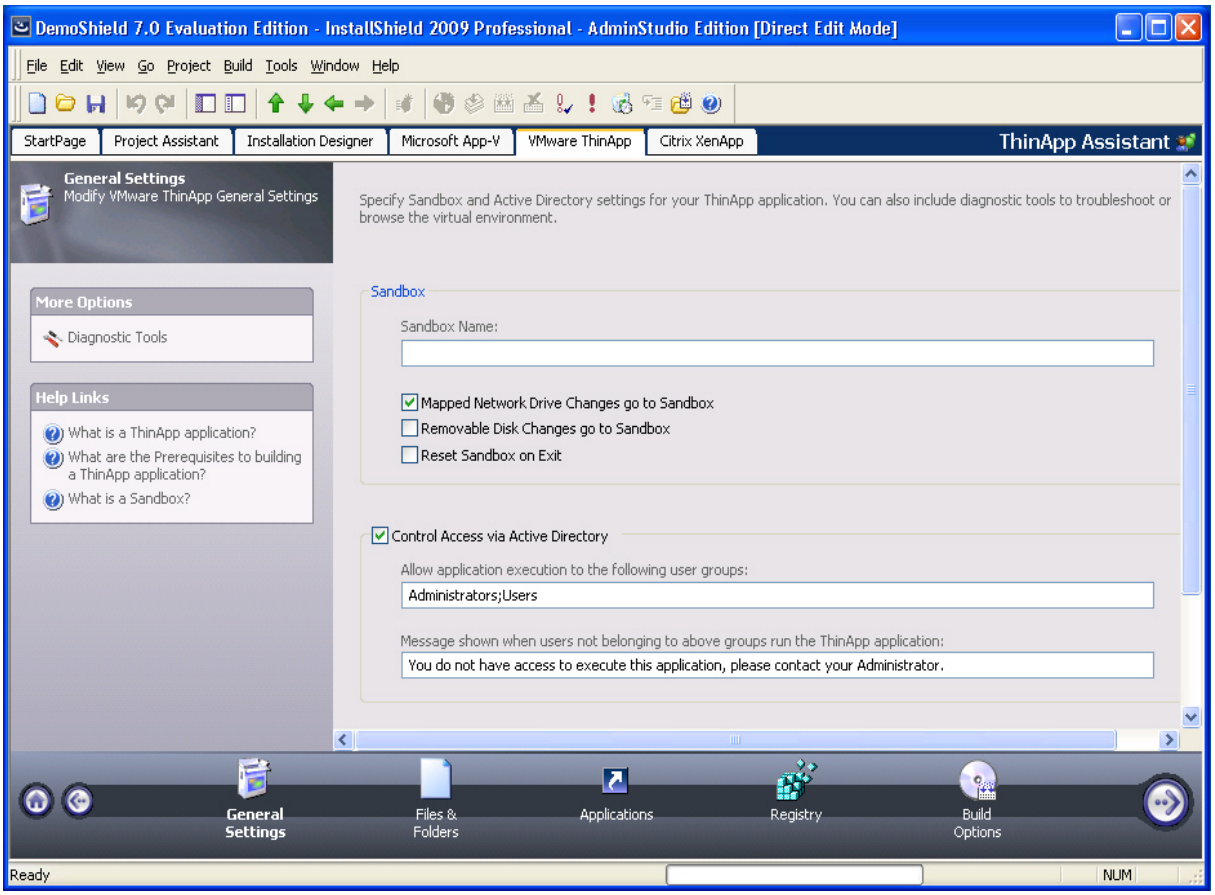


Figure 13-37: ThinApp Assistant General Settings Page

The **General Settings** page includes the following options:

Table 13-50 • General Settings Page

Option	Description
Sandbox Name	When a ThinApp application is built, a Sandbox cache is created in the following location: `c:\Documents & Settings\USER_NAME\Application Data\ThinApp\SANDBOX_NAME` By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the **Sandbox Name** field.
Mapped Network Drive Changes go to Sandbox	Enable this option if you want changes for Network mapped drives to be saved in the sandbox. By default, users can read and write normally to network mapped drives.
Removable Disk Changes go to Sandbox	Enable this option if you want changes for removable disks to be saved in the sandbox. By default users can read and write normally to removable disks.
Reset Sandbox on Exit	Select this option to delete the sandbox content when the application exits. This resets the ThinApp application to its original captured state.
Control Access via Active Directory	If you want to control the access of users to a ThinApp application by specifying Active Directory groups, select this option and enter the names of those groups. At build-time, ThinApp would then assign a unique GUID-like number to uniquely identify each Active Directory Group that you have identified. Members of those groups will have access to the ThinApp application. • **Allow application execution to the following user groups**—Enter the names of all of the Active Directory groups that you want to have permission to run this ThinApp application, separated by semi-colons, such as: `GroupOne;GroupTwo;GroupThree` • **Message shown when users not belonging to above groups run the ThinApp application**—Enter the message that will be displayed when users that do not belong to the specified groups attempt to launch a ThinApp application. ***Caution*** • *If you do not select the **Control Access via Active Directory** option, anyone who has access to a directory containing a ThinApp application will be able to run the application.* ***Note*** • *For more information, see About Controlling Access to ThinApp Applications.*

For testing purposes, you can also choose to include diagnostic tools in your ThinApp application by selecting the **Diagnostic Tools** link in the **More Options** list. For more information, see Diagnostic Tools Dialog Box.

Files & Folders Page

On the **Files & Folders** page of the ThinApp Assistant, you can perform the following tasks:

- View Files and Folders
- Add Files and Folders
- Delete Files and Folders
- Set Isolation Options
- Modifying the Display of Predefined Folders

View Files and Folders

On the **Files & Folders** page, you can view all of the files and folders that are currently in your ThinApp application.

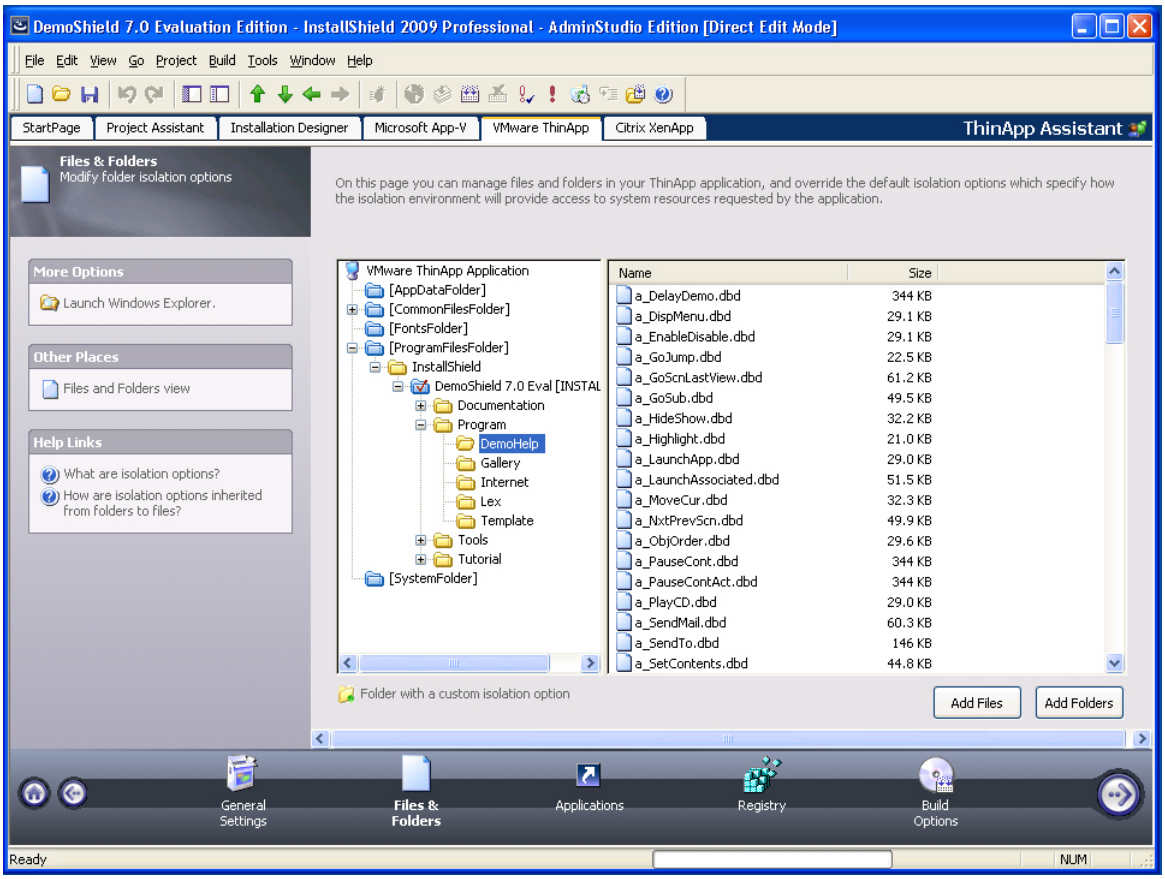


Figure 13-38: ThinApp Assistant Files & Folders Page

Folders are listed in the **VMware ThinApp Application** tree on the left, and all of the files in the selected folder are listed on the right.

- The directories in the tree represent how your application will be organized within its secure compressed container.
- Blue folders are the supported MSI standard folders.
- The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

Add Files and Folders

On the **Files & Folders** page, you can use the **Add Files** and **Add Folders** buttons to add new files and folders to include in the ThinApp application. See Adding, Deleting, and Moving Files and Folders in an App-V Package.

If you are editing an InstallShield project (not a Windows Installer package), and you are adding a folder to this ThinApp application, you are prompted to choose whether you want to create a dynamic file link to the source folder.

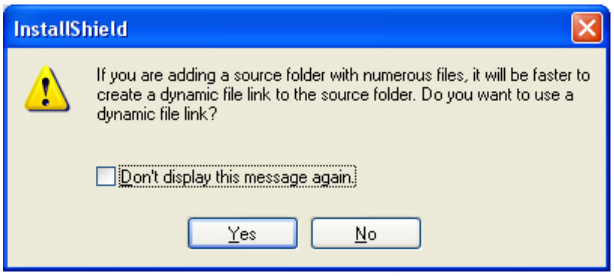


Figure 13-39: Dynamic File Link Option Dialog Box

Indicate whether you want to create a dynamic file link by selecting one of the following:

- **No**—For more flexibility with ThinApp options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
- **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

Delete Files and Folders

You can delete files and folders from the ThinApp application by selecting the file or folder you want to delete, and selecting **Delete** from the context menu. For more information, see Deleting Files and Folders.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains.*

Note • *You cannot delete predefined folders. You can only turn of the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Set Isolation Options

ThinApp uses a sandbox virtual environment to control application compatibility and accessibility. The isolation option that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a file or folder and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting Isolation Options.

Modifying the Display of Predefined Folders

You can specify which of the Windows Installer predefined folders are listed in the **VMware ThinApp Application** tree. See Controlling the Display of Predefined Folders.

Applications Page

You define shortcuts to enable users to launch a ThinApp application from within the sandbox virtual environment.

By default, the **ThinApp Assistant** creates ThinApp applications for all of the executable shortcut that exist in your project. The project's shortcuts are listed in a checklist on the **Applications** page.

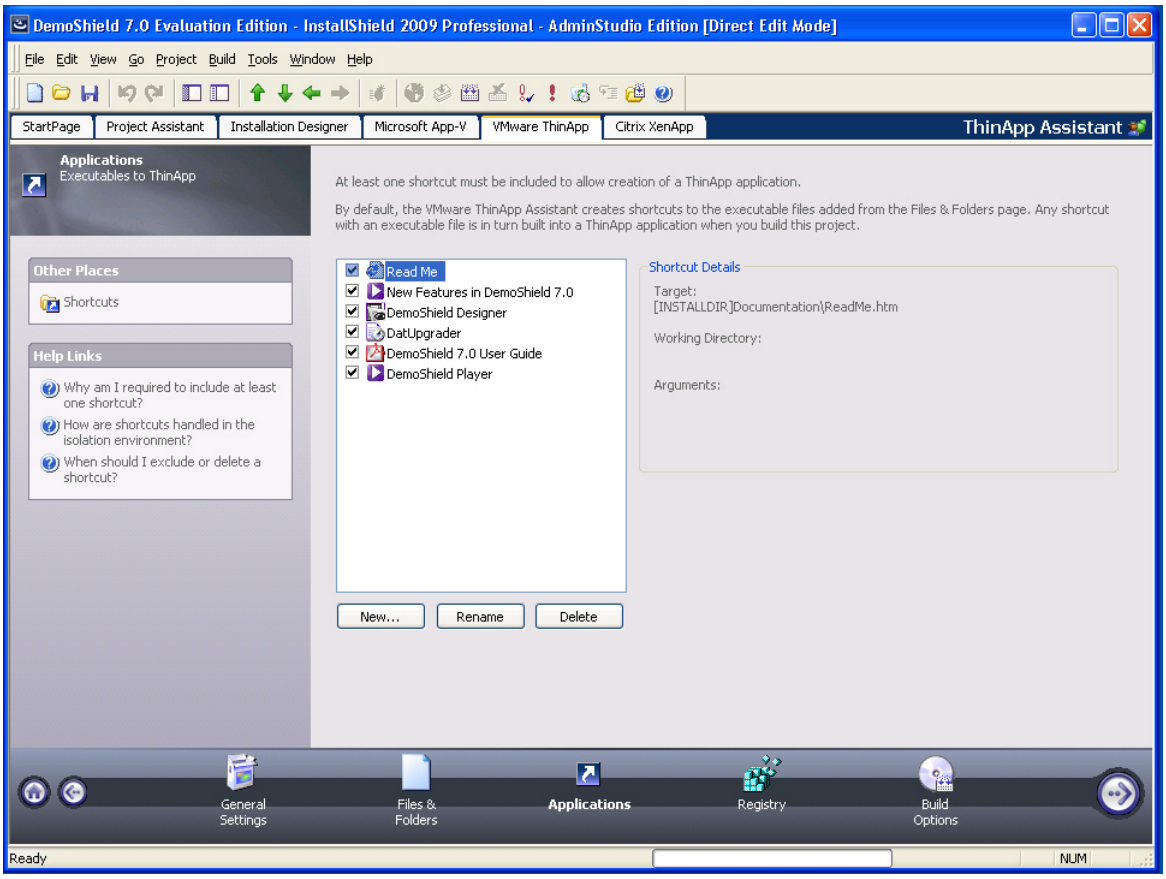


Figure 13-40: ThinApp Assistant Applications Page

Shortcut Requirements

For each ThinApp application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build a ThinApp application that does not contain any shortcuts, users will not be able to launch the application.

Difference Between Deleting and Excluding a Shortcut

To prevent a shortcut from being created in the ThinApp application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Managing Shortcuts

On the **Applications** page, you can create, delete, include, exclude, or rename a ThinApp application. For step-by-step instructions, see the following topics:

- Creating a New App-V Package
- Including an Existing App-V Shortcut
- Excluding or Deleting an Existing App-V Package
- Renaming a Shortcut

Registry Page

On the **Registry** page, you can view existing registry keys, values, and data, and add or delete registry items. You can also override the default isolation options for a registry key. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a registry key and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting Isolation Options.

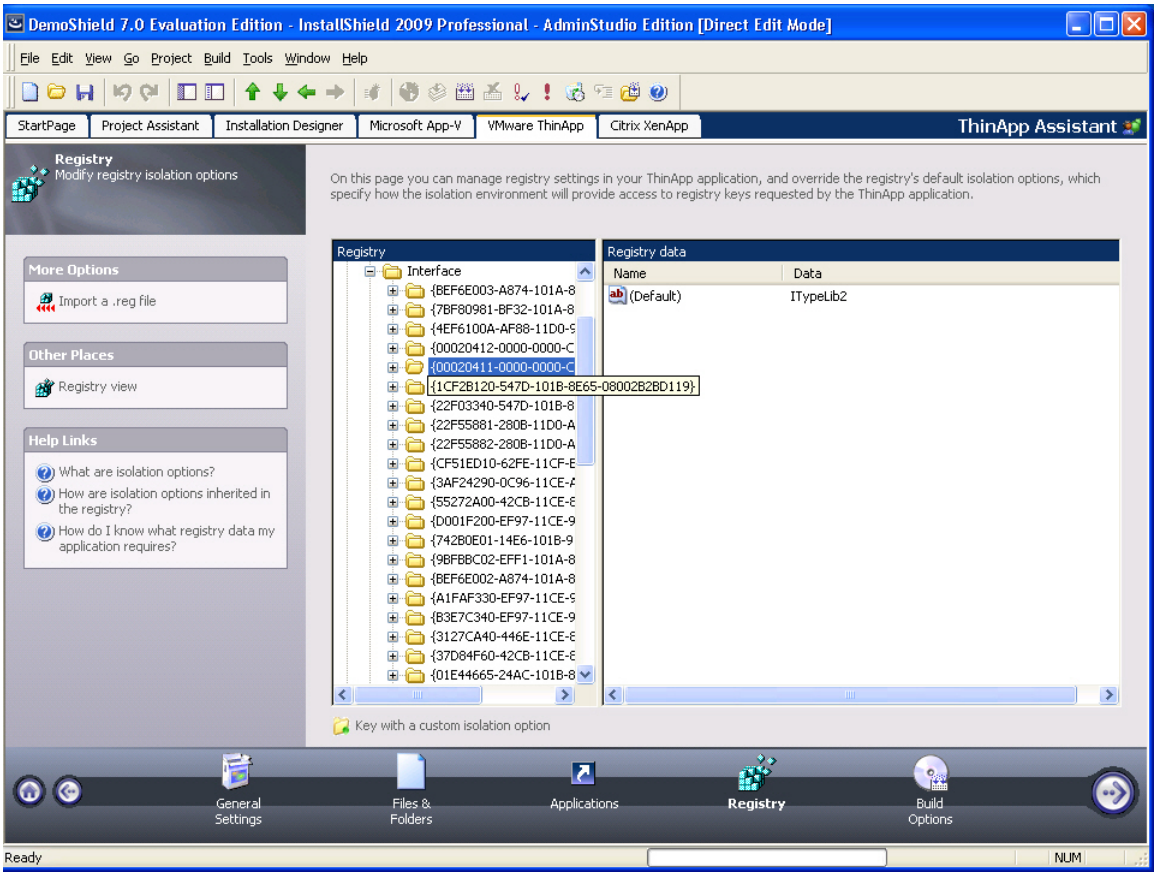


Figure 13-41: ThinApp Assistant Registry Page

Registry items that are listed on this page will be included in the ThinApp application, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the Registry page is performed much like it is performed on the InstallShield Registry View. See Editing the Registry.

For information on how to override a registry key's default isolation options, see Setting App-V Package Registry Isolation Options.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Build Options Page

On the **Build Options** page, you can perform the following tasks:

- Specifying Build Options
- Including Additional Windows Installer Packages in a ThinApp Application
- Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application
- Selecting Releases to Build
- Enabling Citrix Profile Building When in Direct Edit Mode
- Clearing the ThinApp Cache
- Opening the ThinApp Application Folder
- Building a Citrix Profile
- Supporting AppSync and AppLink

The options on the Build Options page vary depending upon whether you are editing an InstallShield project or a Windows Installer package:

InstallShield Project

When you open an InstallShield project in InstallShield:

- The **Build Options** page includes a releases tree, and you select the release that you want to build.
- To build the ThinApp application, you click the **Build** button on the toolbar.

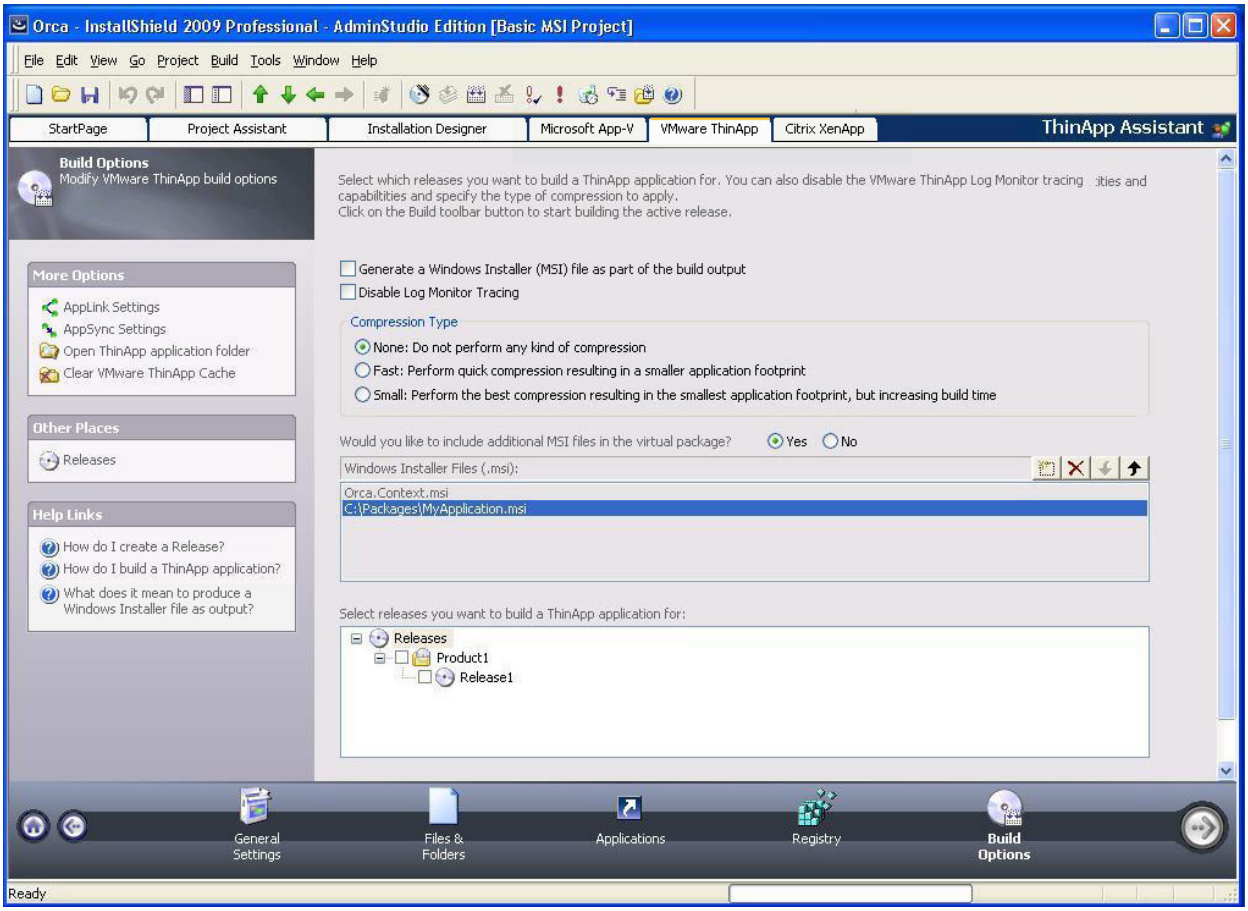


Figure 13-42: Build Settings Page, When in Basic MSI Project Mode

Windows Installer Package [Direct Edit Mode]

When you open a Windows Installer package in InstallShield:

- Because you do not have to select a release for a Windows Installer package, there is no releases tree.
- Because a Windows Installer package has already been built, InstallShield's standard build functionality is disabled. To build the ThinApp application, select the **Build ThinApp application** option and click the **Build Virtual Package** button.

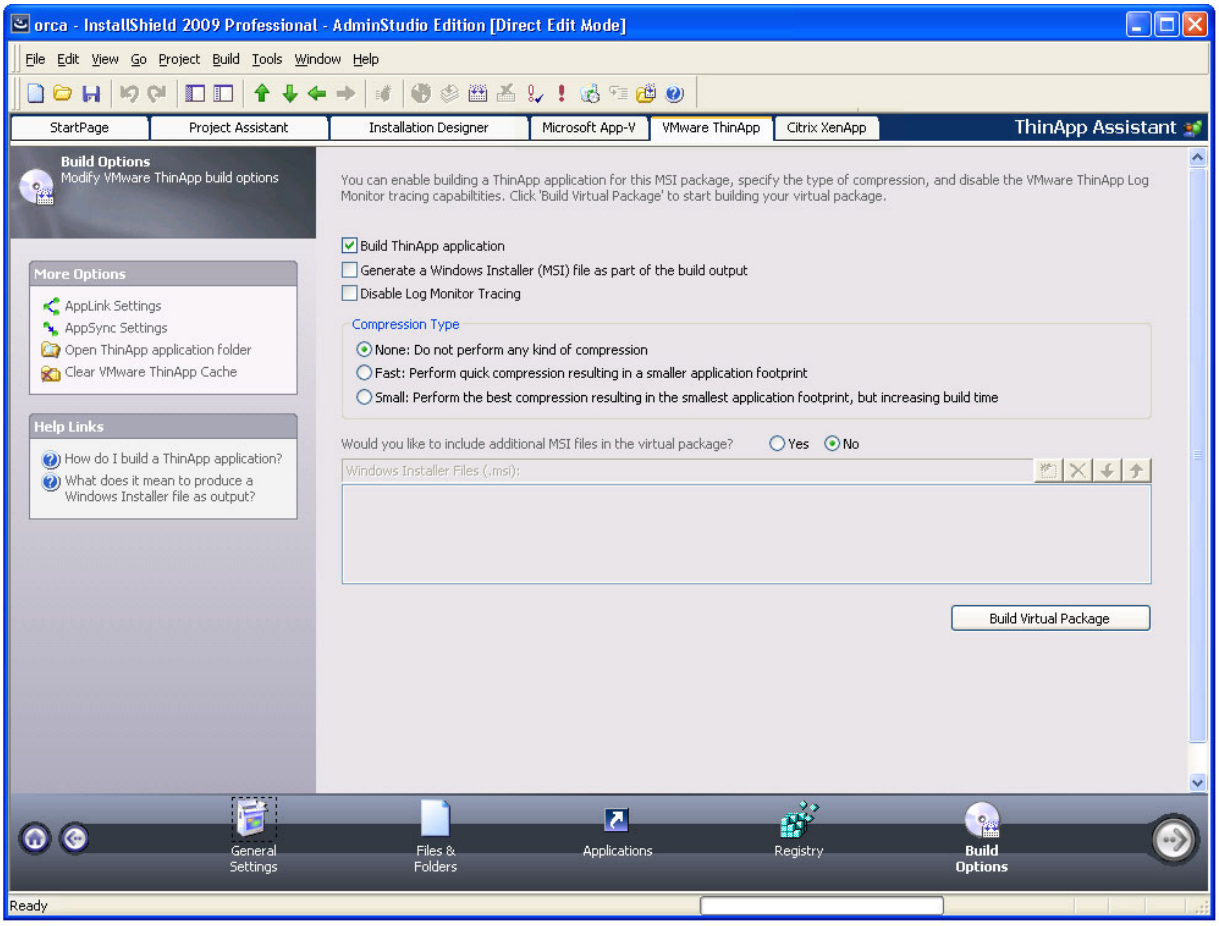


Figure 13-43: Build Settings Page, When in Direct Edit Mode

Specifying Build Options

On the **Build Options** page, you can specify the following options:

Table 13-51 • ThinApp Application Build Options

Option	Description
Build ThinApp Application	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, it is not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled. Select the **Build ThinApp Application** option to enable the Build function. When this option is selected, the **Build Virtual Package** button is enabled. For more information, see Enabling Citrix Profile Building When in Direct Edit Mode.

Table 13-51 • ThinApp Application Build Options

Option	Description
Build Virtual Package	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, if you select the **Build ThinApp Application** option, this button is enabled. Click it to build the ThinApp application. ***Note*** *• This button will also be enabled if the **Build Citrix profile** option is selected on the **Build Settings** page of the Citrix Assistant. In this scenario, if you click this button without also selecting the **Build ThinApp application** option on this page, the ThinApp application will not be built.*
Generate a Windows Installer (MSI) file as part of the build output	You can choose to build a Windows Installer package with your ThinApp application. This enables you to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management to distribute your ThinApp application. To build a Windows Installer file with your ThinApp application, select this option. By default, this option is not selected. For more information, see Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application.
Disable Log Monitor Tracing	Select this option if you do not want to allow ThinApp Log Monitor tracing for a ThinApp application. ThinApp Log Monitor is an application in the ThinApp Suite that allows you to record detailed information about any application's execution history for later review. For more information, see Setting ThinApp Log Monitor Tracing Options.
Compression Type	Select one of the following options to specify the ThinApp application's compression type: • **None**: Do not perform any type of compression • **Fast**: Perform quick compression resulting in a smaller application footprint • **Small**: Perform the best compression resulting in the smallest application footprint, but increasing build time. ***Note*** *• For more information, see Compressing a ThinApp Application.*
Would you like to include additional MSI files in the virtual package?	Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. To include additional Windows Installer packages in a ThinApp application, set this option to **Yes**, and then select the packages that you want to add. ***Note*** *• For more information, see Including Additional Windows Installer Packages in a ThinApp Application.*

Including Additional Windows Installer Packages in a ThinApp Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the ThinApp Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a ThinApp application, set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**, and then select the packages that you want to add.

- Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
- The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
- Use the Delete button ( ) to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application

You can choose to build a Windows Installer package to assist in the distribution of a ThinApp application by selecting the **Generate a Windows Installer (MSI) file as part of the build output** option on the **Build Options** page. By default, this option is not selected.

The Windows Installer file can be run to properly install the ThinApp application on an end-user's desktop. This simplifies the deployment of a ThinApp application by enabling you to use enterprise distribution tools such as ConfigMgr (Formerly called as System Center Configuration Manager) or Microfocus ZENworks Configuration Management.

A ThinApp application installed using a Windows Installer package can be uninstalled using **Add or Remove Programs** in the Control Panel.

Selecting Releases to Build

You select the releases that you want to build a ThinApp application for on the **Releases** tree of the **Build Options** page. By selecting a release, you are specifying that whenever that particular release is built, a ThinApp application will also be built.

Note • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.*

About Building Releases

When you select a release on the Releases tree on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Options** page:

- **Active release selected**—A Windows Installer package and a ThinApp application would be built.
- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

About Creating Releases

You cannot create or edit a release in the ThinApp Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see Creating and Building Releases.

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.

Enabling ThinApp Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **ThinApp Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a ThinApp application for this Windows Installer package. To enable the **Build** button to build just the ThinApp application, select the **Build ThinApp application** option on the **Build Options** page.

After you select this option, the **Build ThinApp application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Clearing the ThinApp Cache

When you perform compressed builds, large temporary files are saved in a cache location. To delete all of these temporary files, select the **Clear the VMware ThinApp Cache** option in the **More Options** list on the **Build Options** page

Opening the ThinApp Application Folder

To quickly open the folder containing the ThinApp application files that were generated when this InstallShield project or Windows Installer package was built, click **Open ThinApp application folder** in the **More Options** menu.

Building a ThinApp Application

The method for building a ThinApp application depends upon what file you have open—an InstallShield project or a Windows Installer package. For detailed instructions, see one of the following topics:

- Building an App-V Package from Within an InstallShield Project
- Building an App-V Package from Within a Windows Installer Package in InstallShield

Supporting AppSync and AppLink

To configure AppSync and AppLink settings for your ThinApp application, click the **AppSync Settings** or **AppLink Settings** option in the **More Options** menu. For more information, see the AppSync Settings Dialog Box or the AppLink Settings Dialog Box.

Dialog Boxes

The ThinApp Assistant includes the following dialog boxes:

- Diagnostic Tools Dialog Box
- Folder Isolation Options Dialog Box
- Isolation Options Dialog Box (for Registry Keys)
- AppSync Settings Dialog Box
- AppLink Settings Dialog Box
- Add AppLink Reference Dialog Box

ThinApp Diagnostic Tools Dialog Box

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **General Settings** page, you can choose to include the Windows Command Prompt and Registry Editor diagnostic tools with your ThinApp application.

If you include diagnostic tools with your ThinApp application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running a ThinApp application and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the ThinApp application. However, these tools may not be compatible with other operating systems.*

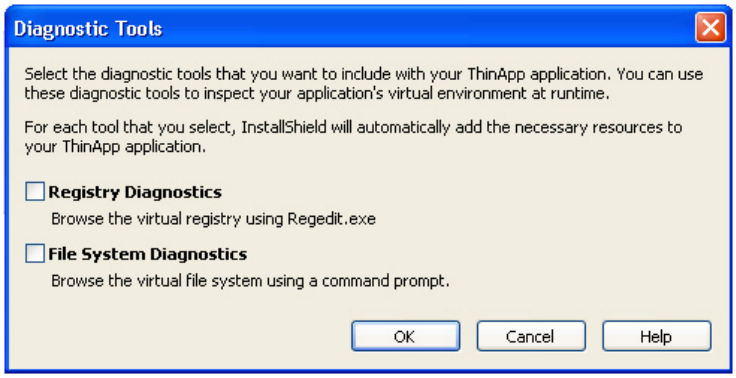


Figure 13-44: Diagnostic Tools Dialog Box

You can use these diagnostic tools to inspect your application's virtual environment at runtime. You have the following options:

Table 13-52 • Diagnostic Tools Dialog Box Options

Option	Description
Registry Diagnostics	Select this option if you want to include `regedit.exe` with your ThinApp application so that you can browse the registry.
File System Diagnostics	Select this option if you want to be able to browse the ThinApp application's virtual environment file system using a command prompt.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the ThinApp application.

When the user runs this ThinApp application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Folder Isolation Options Dialog Box

On the **Folder Isolation Options** dialog box, you can override the default isolation options for the selected folder.

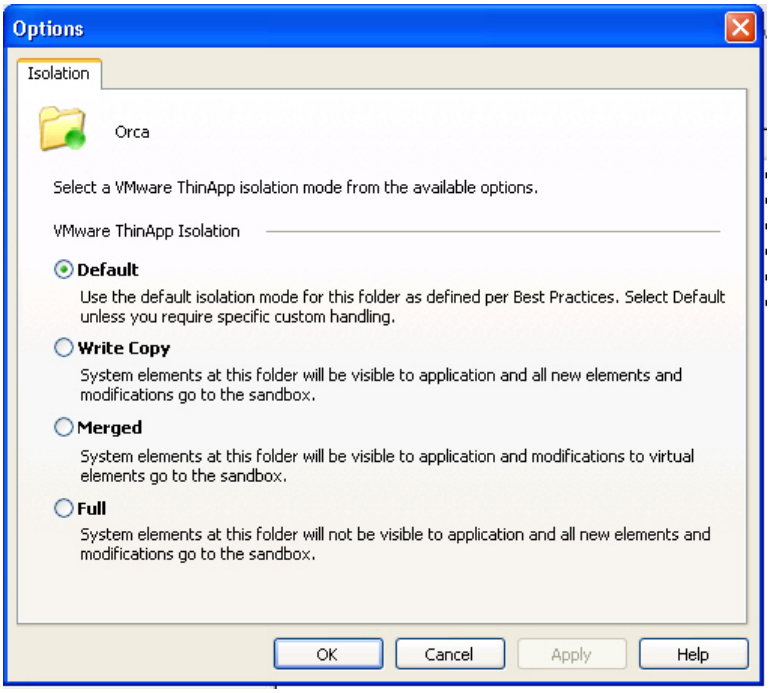


Figure 13-45: Folder Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

The **Folder Isolation Options** dialog box includes the following options:

Table 13-53 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 13-54 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

Registry Isolation Options Dialog Box

On the **Registry Isolation Options** dialog box, you can override the default isolation options for the selected registry key.

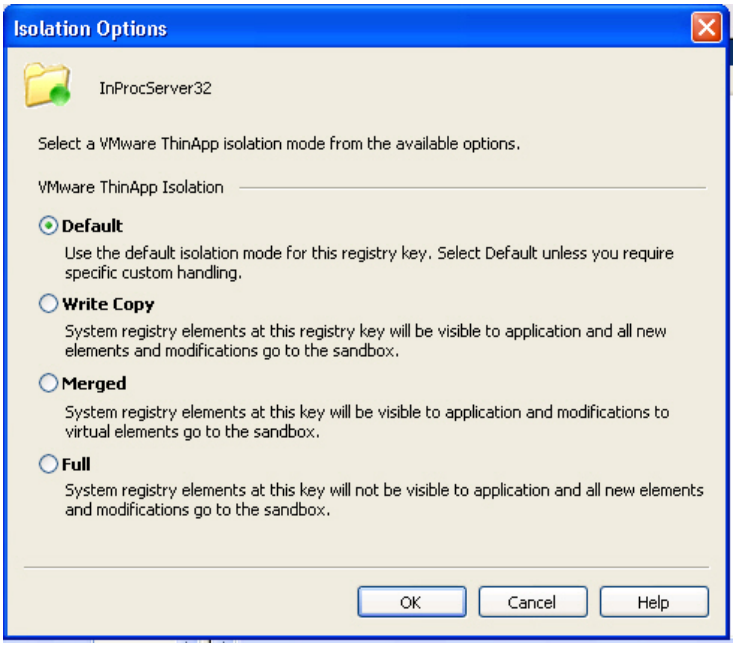


Figure 13-46: Registry Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

The **Registry Isolation Options** dialog box includes the following options:

Table 13-55 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 13-56 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

AppLink Settings Dialog Box

Note • *The AppLink Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. You can set AppLink settings for the current ThinApp application on the **AppLink Settings** dialog box, which is opened by clicking the **AppLink Settings** option in the **More Options** menu of the ThinApp Assistant **Build Options** page.

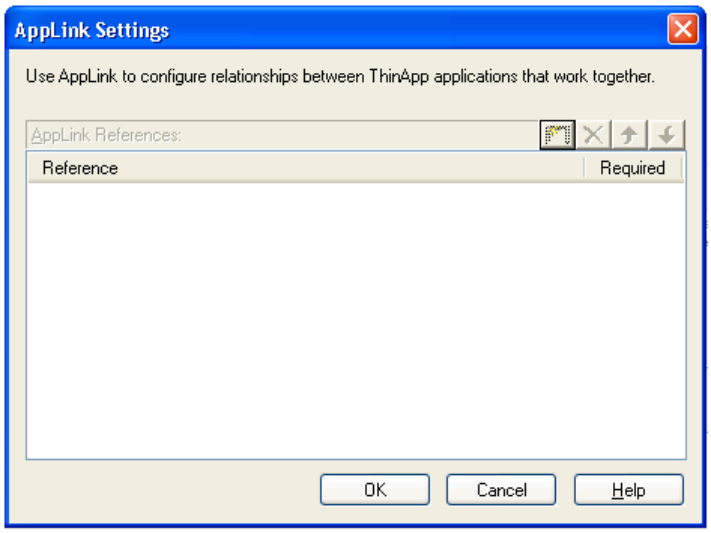


Figure 13-47: AppLink Settings Dialog Box

You can use the AppLink feature to perform the following tasks:

- **Linking runtime components to applications**—You can link runtime components to the applications that use them. For example, you can link a package containing the Java runtime environment (JRE) or ODBC drivers to a package containing a browser application.
- **Linking add-ons and plug-ins to applications**—You can link add-ons and plug-ins to applications. For example, Microsoft Office add-ons can be linked to applications or Adobe Photoshop plug-ins can be linked to a package containing Photoshop.
- **Linking packaged applications to service packs**—You can link packaged applications to service packs. By using AppLink, you can upgrade or roll back your service packs by changing the service pack that you capture and link to its parent application.

The **AppLink Settings** dialog box has the following options:

Table 13-57 • AppLink Settings Dialog Box

Option	Description
AppLink References	List of ThinApp applications that are linked to the open ThinApp application. The following information is listed: • **Reference**—List of linked ThinApp applications, including the application location and name. • **Required**—If **Yes** is listed in this column, the linked application must be available in order for the ThinApp application to run. If the linked application cannot be found, the ThinApp application will fail to run. See Required and Optional Linked Applications for more information.
Browse Button	Click the Browse button to open the **Add AppLink Reference** dialog box, where you can add a linked application to the **AppLink Reference** list. For more information, see Add AppLink Reference Dialog Box.

Table 13-57 • AppLink Settings Dialog Box

Option	Description
Up and Down Arrows	ThinApp uses a "last import wins" policy to determine what happens when two packages are imported that have the same files or registry keys. Therefore, you can use the Up and Down arrows to order the list of linked applications. See Collisions and Order of Import for more information. ***Note*** *• Initially, the Required and Optional linked applications are listed on this dialog box together, and you can change the order of these applications using the Up and Down arrows. However, at runtime, the linked applications in the Required category are read first, before those in the Optional category, even though an Optional application might have been listed before a Required application in the AppLink References list. Also, each time the* ***AppLink Settings*** *dialog box is reopened, the Required linked applications will be grouped at the top of the list, before all Optional applications.*

Required and Optional Linked Applications

When an application is linked to a ThinApp application, it can be designated to be either Required or Optional:

Required Applications

If a package is required, it has a mark in the **Required** column. If this package is missing from the virtual package, it will fail to run.

- If any specified package fails to import, an error message will be displayed and the parent executable file will exit.
- If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
- To continue even if load errors occur, make the package references optional instead.

Optional Applications

If a package does not have a mark in the **Required** column, it is optional. An optional package operates the same as a required package except that if an import fails to load, the error is ignored and the main application will start executing.

Collisions and Order of Import

ThinApp uses a "last import wins" policy to determine what happens when two packages are imported that have the same files or registry keys.

For example, if `PackageA.exe` has `c:\myinfo.txt` in its virtual file system and `PackageB.exe` also has `c:\myinfo.txt` in its virtual file system, ThinApp will determine what happens based on which package is imported last.

- **Package order in the AppLink References list**—If `PackageA.exe` is listed before `PackageB.exe` on the AppLink References list, `PackageB.exe`'s copy of `c:\myinfo.txt` will be used. But if `PackageB.exe` is listed before `PackageA.exe` on the AppLink References list, `PackageA.exe`'s copy of `c:\myinfo.txt` will be used.
- **Wild cards**—When wild cards are used, alphabetical order is used to load packages, so if you enter `Package*.exe` in the AppLink References list, `PackageB.exe` will be loaded last (after `PackageA.exe`), so its copy of `c:\myinfo.txt` will be used.

- **VB scripts**—If two or more packages include VB scripts, the order of execution for the VB Scripts will be alphabetical order by the name of the package. If two packages contain a VB script with the same name, the "last import wins" policy will be used to execute only the version of the VB script from the last imported package containing a script with that name.

Caution • *Because VB Script name collisions could cause scripts from other packages not to be executed, it is important to use unique name for VB Script filenames.*

Security and Authorization

The user running the ThinApp application must be a member of all `PermittedGroups` sections for all of the linked (imported) ThinApp applications. If this is not the case, an `Access Denied` message will be displayed and the main ThinApp application will fail to load.

The following are limitations of the AppLink feature:

- ThinApp supports importing up to 250 packages at a time, and each package may be any arbitrary size.
- Packages that have been updated via AppSync will not have updates visible to the parent executable.
- Sandbox changes from packages being imported will not be visible to the parent executable.

Add AppLink Reference Dialog Box

Note • *The AppLink Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. On the **Add AppLink Reference** dialog box, which is opened by clicking the Browse button on the **AppLink Settings** dialog box, you specify the name and location of a ThinApp application and indicate whether that application is Required or Optional.

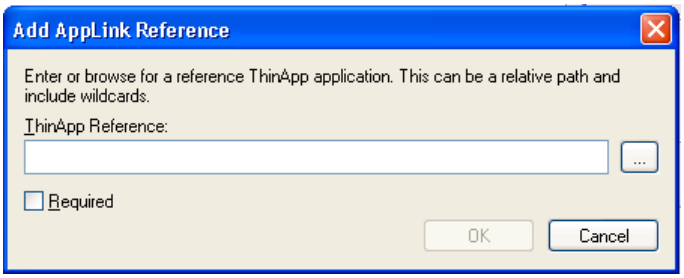


Figure 13-48: Add AppLink Reference Dialog Box

In the **ThinApp Reference** box, enter the relative (runtime) path to the existing ThinApp application that you want to link to. For more information on how to specify a ThinApp Reference, see the following:

- Enter a Relative Path
- Path Name Format
- Which ThinApp File Should Be Specified in an AppLink Reference?

- Required vs. Optional
- Examples of AppLink References

Enter a Relative Path

On the **Add AppLink Reference** dialog box, if you click Browse and browse for a ThinApp application, the absolute path to that application is entered, such as `C:\Program Files\AppName\filename.exe`. In that case, the main ThinApp application needs that linked application to be found at the specified absolute path location at runtime, which is unlikely. Therefore, it is recommended that you enter a relative path name.

Path Name Format

AppLink supports both URL and UNC path names.

Which ThinApp File Should Be Specified in an AppLink Reference?

If a ThinApp application has only one shortcut, it consists of a single executable. Therefore, you would obviously specify that executable file when creating an AppLink Reference.

However, when a ThinApp application has more than one shortcut, the ThinApp file that you specify in an AppLink Reference depends upon what tool you used to build the ThinApp application:

Table 13-58 • File to Specify in an AppLink Reference

Tool Used to Build ThinApp Application	# of Shortcuts	ThinApp Application File to Specify
AdminStudio or ThinApp	Only one	Specify the executable file (`.EXE`).
AdminStudio	More than one	When built with AdminStudio, a ThinApp application that has more than one shortcut consists of two or more executable files and a `Package.DAT` file (as described in Components of an App-V Package). In this situation, specify the `Package.DAT` file.
ThinApp	More than one	When built with ThinApp, a ThinApp application that has more than one shortcut consists of multiple executable files, with one primary executable. In this situation, specify the primary executable file (`.EXE`).

Required vs. Optional

If you want this package to be required, select the **Required** option. If a required package is missing from the virtual package, it will fail to run. Note the following about required packages:

- If any specified package fails to import, an error message will be displayed and the parent executable file will exit.
- If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
- To continue even if load errors occur, make the package references optional instead.

Examples of AppLink References

The following are examples of how packages can be added to the **AppLink References** list:

Table 13-59 • AppLink References Examples

Example	Description
`Plugin.exe`	This will import a single package located in the same directory as the parent executable.
`plugins\Plugin.exe`	This will import a single package located in the `plugins` subdirectory of the parent executable.
`plugins\*.exe`	This will import all executables located in the `plugins` directory. ***Important*** • *If any executable fails to import because it is not a proper ThinApp package or because of a security issue, the parent executable will fail to load.*
`n:\plugins\*.exe`	This will import all EXEs located at the absolute path `n:\plugins`.
`%PLUGINS%\*.exe`	This expands the environment variable, `PLUGINS`, and imports all executables found at this location.
`plugin1.exe;plugin2.exe;plugins\*.exe`	This loads two specified plugins and a list of executables found in the `plugins` subdirectory.

AppSync Settings Dialog Box

Note • *The AppSync Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppSync settings that you define will be ignored.*

AppSync (Application Sync) enables you to automatically keep deployed virtual applications up to date. When an application starts up, AppSync can query a Web server to see if an updated version of the package is available. If an update is available, the differences between the existing package and the new package will be downloaded and used to construct an updated version of the package. The updated package will be used for future deployments.

You can use the AppSync feature to perform the following tasks:

- **Distribute runtime components separately**—You can use AppSync to distribute runtime components separately from the applications that use them. For example, the Java Runtime Environment (JRE) or ODBC drivers.
- **Apply layered service packs to applications**—You can use AppSync to apply layered service packs to your applications. Application Sync enables you to distribute service packs and roll back to previous versions, if necessary.

On the **AppSync Settings** dialog box, which is opened by clicking **AppSync Settings** on the M**ore Options** menu of the **Build Options** page, you can configure AppSync settings for your ThinApp application.

On the **Expiration** tab, you can specify that the ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run. Note that the update does not expire, the ThinApp application expires, and cannot be used until it is updated. successfully.

The **AppSync Settings** dialog box includes two tabs:

- General Tab
- Expiration Tab

General Tab

On the **General** tab, you specify the location of the Web server that hosts application updates.

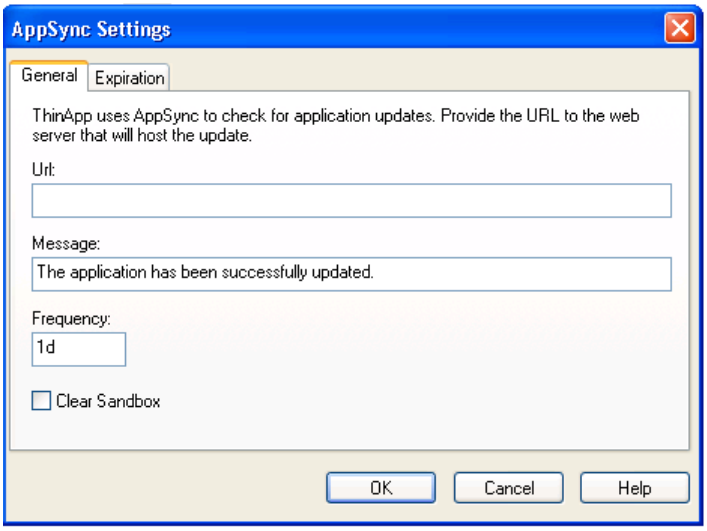


Figure 13-49: General Tab of the AppSync Settings Dialog Box

The following options are included:

Table 13-60 • General Tab of the AppSync Settings Dialog Box

Option	Description
Url	URL of the Web server where updates are stored. Application Sync works over both the HTTP (unsecure) and HTTPS (secure) protocol. Part of HTTPS is that the identity of the Web server is checked. You can include a user name and password in the URL that will be used for basic authentication. The standard Windows/Internet Explorer proxy setting is respected. For example: `https://example.com/some/path/PackageName.exe`
Message	When an updated package is first launched, an information message can be shown. For example: `Your application has been updated.`
Frequency	By default, a package will connect to the Web server once per day to see if an updated version is available. You can set the frequency by modifying this setting. For example, to set the **Frequency** to 2 days, enter **2d**. For 2 weeks, enter **2w**, etc.

Table 13-60 • General Tab of the AppSync Settings Dialog Box

Option	Description
Clear Sandbox	Gives you the option to clear the sandbox after an update. By default, the sandbox is not cleared. Select this option to clear the sandbox.

Expiration Tab

On the **Expiration** tab, you can specify that a ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run.

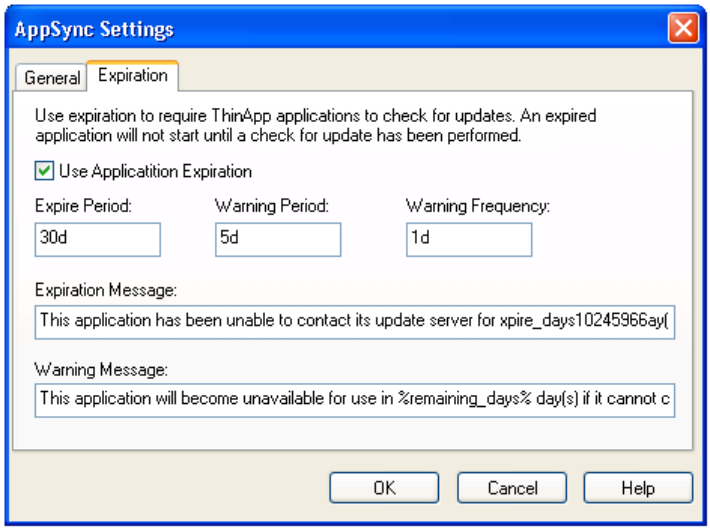


Figure 13-50: Expiration Tab of the AppSync Settings Dialog Box

The following options are included:

Table 13-61 • Expiration Tab of the AppSync Settings Dialog Box

Option	Description
Use Application Expiration	Select this option to require that an application has to check for updates at a specified frequency.
Expire Period	Sets the update frequency in minutes (`m`), hours (`h`), or days (`d`). If the Web server cannot be reached, the package will continue to work until the Expire Period is reached. This default setting is 30 days but you can change that setting by modifying this setting. For example: • To set the period to 30 days, enter **`30d`** • If you do not want the package to expire, clear the **Use Application Expiration** check box.
Warning Period	Sets the start of the warning period before a package expires. For example, to set the period at 5 days, enter **`5d`**.

Table 13-61 • Expiration Tab of the AppSync Settings Dialog Box

Option	Description
Warning Frequency	Sets the frequency of warnings before the package expires. With the default of one day, the warning message will be displayed once per day only. To configure the warning to pop up on every application launch, enter **0**. To configure it to pop up every 4 days, enter **4d**. After the warning period has started, the Web server will be checked on every launch of an application, overriding any previous setting. As long as a package has not expired, this parameter checks for new versions and downloads will occur in the background. The user can continue to use the old version. If the application is terminated by the user before the download is complete, the download will resume when a virtual application is launched again. After the download completes, the new version will be activated on the next launch. When the package has expired, the version check and download will happen in the foreground. A progress bar will be shown during the download phase.
Expiration Message	After the expiration limit has been reached and a virtual application is started, it will try to connect to the Web server and check for a new version. If the connection fails, a message box will be shown and execution will be terminated. The default message is shown in the example below. For example: `This application has been unable to contact its update server for` *`Expire_Period`* `days, so it is unavailable for use. Check your network connection and try again.`
Warning Message	If the connection to the Web server fails, a message box will be shown. The default message is: `This application will become unavailable for use in` *`Warning_Period`* `days if it cannot contact its update server. Check your network connection to ensure uninterrupted service`

Note • *If you use AppSync, VMware recommends that you disable automatic application updates that are configured in your virtual application. Conflicts might occur between the linked packages and the software that is automatically updated. If an automatic update feature updates an application, it stores the updates in the sandbox. If AppSync then updates the application to a different version, the updates stored in the sandbox take precedence over the files contained in the version that AppSync created. The order of precedence for the update files are those in the sandbox, then the virtual operating system, and then the physical machine.*

Building ThinApp Applications Using the Command Line

When you configure a ThinApp application in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the ThinApp application are built. When you use the standard InstallShield command line build, you do not need to add any additional command line parameters. All of the ThinApp application settings are saved within the InstallShield project.

ThinApp Application Conversion Error and Warning Messages

For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a ThinApp application. Therefore, some additional pre- or post-conversion actions must be taken in order for the ThinApp application to be created properly.

One action you could take to try to include ignored features in an ThinApp application is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a ThinApp application.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

ThinApp Not Found

To create a ThinApp application, you are required to have both AdminStudio and ThinApp installed on the same machine. If a user attempts to create a ThinApp application without this ThinApp component, a message is displayed and the build is unsuccessful.

To purchase the ThinApp, visit the VMware Web site:

http://www.vmware.com/products/thinapp/

ThinApp Application Configuration File: package.ini

ThinApp application configuration options that you set in the ThinApp Assistant interface are recorded in the `package.ini` file that is generated when the ThinApp application is built.

A `package.ini` contains the following groups of options:

- [BuildOptions]
- [Compression]
- [Isolation]
- [MainApp.exe]
- [Test.exe]

Note • *For the latest information on the ThinApp application configuration file, package.ini, consult your ThinApp documentation.*

[BuildOptions]

The `[BuildOptions]` section of the package`.ini` file specifies Global options which will be inherited by each child executable file. The following options are included:

Table 13-62 • [BuildOptions] Section of package.ini

Option	Description

SandboxName	When a ThinApp application is built, a Sandbox cache is created in the following location:
	`c:\Documents & Settings\USER_NAME\Application Data\ThinApp\SandboxName`
	The `SandboxName` entry in the package`.ini` file is used to name the directory where sandbox files are stored at runtime.
	`SandboxName=MyApplicationV3`
	By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the package.ini file using the `SandboxName` option.
	If no Sandbox Name is entered, a unique GUID is used, such as:
	`SandboxName={2BDBE10A-9E53-4B5E-811D-DF8019D0B13C}`
	
	Note • *This option corresponds to the **Sandbox Name** field on the **General Information** page.*

InventoryName	Used by desktop management systems to identify packages for usage reporting purposes. If you do not use a desktop management system or license-controlled system, this value has no effect `InventoryName=MainApp v1.0`
SandboxNetworkDrives	Enable this option if you want changes to data on Network-mapped drives to go into the sandbox. By default, the ThinApp application can read and write to network mapped drives with no changes. The value for **SandboxNetworkDrives** is set to either 0 (off) or 1 (on). `SandboxNetworkDrives=0` ***Note*** • *This option corresponds to the **Mapped Network Drive Changes go to Sandbox** option on the **General Settings** page.*

Table 13-62 • [BuildOptions] Section of package.ini (cont.)

Option	Description
SandboxRemovableDisk	Enable this option if you want changes to data on Removable disk (floppy/flash) to go into the sandbox. By default the application can read and write to removable disk with no changes. The value for **SandboxRemovableDisk** can be set to either 0 (off) or 1 (on). `SandboxRemovableDisk=0` ***Note*** *• This option corresponds to the* ***Removable Disk Changes go to Sandbox*** *option on the* ***General Settings*** *page.*
RemoveSandboxOnExit	Enable this option if you want to delete the sandbox when the ThinApp application exits. This resets the application to its original captured state If the application spawns child processes, the clean up will be postponed until all have quit. The value for **RemoveSandboxOnExit** can be set to either 0 (off) or 1 (on). `RemoveSandboxOnExit=0` ***Note*** *• This option corresponds to the* ***Reset Sandbox on Exit*** *option on the* ***General Settings*** *page.*
ExternalCOMObjects	This option allows you to specify that you want specific COM objects to be executed on the system instead of in the virtual environment. This option only applies to out-of-process COM objects (LocalServer32) and Services-based COM objects. To specify multiple objects, put a semicolon after each entry. Objects should always be specified in CLSID format The following class ID specifies the class ID for Microsoft Word: `ExternalCOMObjects={000209FF-0000-0000-C000-000000000046};{000209FF-0000-0000-C000-000000000047}` ***Caution*** *• This option is for advanced users.*

Table 13-62 • [BuildOptions] Section of package.ini (cont.)

Option	Description

VirtualizeExternalOutOfProcessCOM	Enable this option if you want all out-of-process COM objects to be loaded outside of the virtual environment . By doing this, the application may indirectly modify the machine—for example, the MSI installer service COM object could be modified.
	`VirtualizeExternalOutOfProcessCOM=0`
	The value for this option can be set to either:
	• **0**—Inside virtual environment
	• **1**—Outside the virtual environment
	`The default is to create all out-of-process COM objects inside the virtual environment.`

PermittedGroups	Using this option, you can specify the Active Directory groups which are allowed to use this ThinApp application. `PermittedGroups=Group1;Group2;Group3` ***Note*** *• This option corresponds to the* ***Allow application execution to the following user groups*** *option on the* ***General Settings*** *page.*
AccessDeniedMsg	Use this option to customize the message the user sees if they do not have permission to execute a ThinApp application. `AccessDeniedMsg=You do not have access to execute this application, please contact your Administrator` ***Note*** *• This option corresponds to the* ***Message shown when users not belonging to above groups run the ThinApp application*** *field on the* ***General Settings*** *page.*

Table 13-62 • [BuildOptions] Section of package.ini (cont.)

Option	Description

ChildProcessEnvironmentDefault and **ChildProcessEnvironmentExceptions**	Executables located in the virtual file system are always executed within the virtual environment. Executables located in the physical file system can be executed inside or outside the virtual environment.
	
	The default is determined by the **ChildProcessEnvironmentDefault** option, which can be set to **Virtual** or **External**. If this option is not present, the default is the **Virtual** environment.
	
	It is possible to override the default for specific applications by specifying a list of applications, separated by semicolons, using the **ChildProcessEnvironmentExceptions** option. If a complete path is specified, the full name of the executable is used for the comparison; otherwise, only the file name is used.
	
	For example:
	
	`ChildProcessEnvironmentDefault=Virtual`
	`ChildProcessEnvironmentExceptions=exec.exe;c:\path\file.exe`
	
	In this example, `c:\exec.exe`, `c:\Windows\exec.exe` and `c:\path\file.exe` would be executed externally.

AutoShutdownServices	Use this option to specify if virtualized services keep on running when the last non-service process exits. Permitted values are: • **0**—Keep on running. • **1**—Stop virtualized services (Default). `AutoShutdownServices=1`

Table 13-62 • [BuildOptions] Section of package.ini (cont.)

Option	Description
NetRelaunch	Under some conditions, Norton AntiVirus will try to perform a complete scan of an executable. This scan can have a big impact on launch times for large executable files located on network shares. Norton AntiVirus decides to perform a complete scan under these conditions: • **If the executable is launched from a network share or removable disk.** It skips the scan when the executable is located on the hard drive). • **When the executable makes its first network connection.** It does not scan the executable if the executable does not make any network connections. Because a large number of desktops have Norton AntiVirus installed, ThinApp automatically compensates for this by allowing applications to launch from a network share without incurring the lengthy scan times. It does so by creating a small stub executable in the user's sandbox which is then relaunched. Because the small executable can be scanned quickly, it will load the remainder of the application data from the original source location. You can disable ThinApp default behavior by adding the `NetRelaunch=1` option to disable full file scans. `NetRelaunch=1`

[Compression]

The [Compression] options specify the default compression options to use when building the ThinApp application.

Table 13-63 • [Compression] Section of package.ini

Option	Description
CompressionType	To reduce the application startup time, you can specify the CompressionType option to compress the ThinApp application. `CompressionType=Fast` Specify one of the following options: • **None**: Do not perform any type of compression • **Fast**: Perform quick compression resulting in a smaller application footprint • **Small**: Perform the best compression resulting in the smallest application footprint, but increasing build time. ***Note*** *• This option corresponds to the* ***Compression Type*** *options on the* ***Build Options*** *page.*

[Isolation]

The [Isolation] options specify the isolation options to use for folders and registry keys when building the ThinApp application.

Table 13-64 • [Isolation] Section of package.ini

Option	Description
DirectoryIsolationMode	This option specifies the default isolation options to use for folders when building this project. `DirectoryIsolationMode=WriteCopy|Merged` This option has the following possible values: • `WriteCopy`—System elements are visible, modifications to both virtual and system elements are made in the sandbox, new elements are created in the sandbox, and if a system element and a virtual element are at the same location, the application sees the virtual element. • `Merged`—System elements are visible, modifications to virtual elements are made in the sandbox, modifications to system elements are made on the system, new elements are created on the system, and if a system element and a virtual element are at the same location, the application sees the virtual element.

Table 13-64 • [Isolation] Section of package.ini (cont.)

Option	Description
RegistryIsolationMode	This option specifies the default isolation options to use for registry keys when building this project. `RegistryIsolationMode=WriteCopy|Merged` This option has the following possible values: • `WriteCopy`—System elements are visible, modifications to both virtual and system elements are made in the sandbox, new elements are created in the sandbox, and if a system element and a virtual element are at the same location, the application sees the virtual element. • `Merged`—System elements are visible, modifications to virtual elements are made in the sandbox, modifications to system elements are made on the system, new elements are created on the system, and if a system element and a virtual element are at the same location, the application sees the virtual element.

[MainApp.exe]

The `[MainApp.exe]` section specifies the source executable, the name of the file that contains read-only registry data to be bound, whether to perform logging, and the icon to use for the executable.

Table 13-65 • [MainApp.exe] Section of package.ini

Option	Description
Source	This option specifies the `.exe` which will be run to launch the ThinApp application. `Source=%ProgramFiles%\Test\MainApp.exe` This option also specifies the icon that will be used, if an icon is not explicitly specified using the Icon option.
ReadOnlyData	This option specifies the name of the file that contains read-only registry data to be bound. If the read-only registry also has an associated file-data, the file-data file should be in the same directory with the appended extension `TestMain.exe.ro.thfd`. `ReadOnlyData=bin\MainApp.exe.ro.tvr`
DisableTracing	This optional setting will disable logging/tracing capabilities for this application when Log Monitor is running. Possible values are 1 (logging is disabled) or 0 (logging is enabled). `DisableTracing=1` ***Note*** *• This option corresponds to the* ***Disable Log Monitor Tracing*** *option on the* ***Build Options*** *page.*

Table 13-65 • [MainApp.exe] Section of package.ini (cont.)

Option	Description

Icon	By default the icon is used from the executable identified in the Source option. You can change this to specify one of the following:
	`Icon=SomeOtherEXE.exe`
	`Icon=NULL`
	`Icon=SomeOtherIco.ico`

RetainAllIcons	By default, each application retains the main Group Icon from its Source executable and the individual icon resource pointed to by the Group Icon. Tlink will strip out extra icons that cannot be used directly by the system shell. However, you can force these extra icons to be included in the ThinApp executable by using the `RetainAllIcon=1` option. For example:
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`RetainAllIcons=1`
	Instead of using the Source option to identify your application icon, you can also use:
	1. **The value NULL.** In this case, the application will not have an icon and Windows will use the default application icon.
	`[myapp.exe}`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=NULL`
	2. **The path to another .exe file.** In this case, Tlink will load the icons from a different .exe file. If a full path is not specified, the path is relative to the project directory.
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\app2.exe`
	Executable files can contain multiple icon sets. You can optionally specify which set to use by appending ",1" ",2" to the end of the Icon path name like this:
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\app2.exe,1`
	3. **The path to an .ico icon file.** In this case, Tlink will load the icons from the specified .ico file. If a full path is not specified, the path is relative to the project directory.
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\myicon.ico`

[Test.exe]

The `[MainApp.exe]` section specifies the Shortcut and WorkingDirectory options.

Table 13-66 • [Test.exe] Section of package.ini

Option	Description
Shortcut	The **Shortcut** option specifies whether the `.exe` that is generated will contain any registry or file data. This information will be loaded from the `.exe` referenced by the **Shortcut** option. Shortcut applications can specify `WorkingDirectory` and `CommandLine` `Shortcut=MainApp.exe`
WorkingDirectory	The **WorkingDirectory** option specifies where the ThinApp application will start. If this option is not specified, the Current Working Directory will be inherited from the parent process `WorkingDirectory=%ProgramFiles%\Test`

14

Customizing and Authoring Installations Using InstallShield

Edition • *InstallShield Professional is included with AdminStudio Professional Editions. InstallShield Premier is included with AdminStudio Enterprise Edition.*

InstallShield Editor provides the most comprehensive and flexible setup-creation technology available for the Windows Installer. With the latest InstallShield installation development environment (Interface) you can create setup packages that utilize Windows Installer technology, while harnessing the flexibility provided by InstallScript, InstallShield's development language.

Administrators can also take advantage of InstallShield Editor to customize repackaged legacy setups, further enhancing them prior to deploying them in production environments. For AdminStudio Professional and Enterprise Editions, InstallShield Editor contains integrated Application Catalog functionality.

InstallShield Editor's documentation is divided into the following main areas:

Table 14-1 • InstallShield Editor Documentation

Section	Description
InstallShield Editor Integration with Application Catalog and the Software Repository	Explains InstallShield Editor's integration with Application Catalog.
Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support	Provides an overview of how Repackager and InstallShield Editor provide support for the conversion of Windows Installer packages to virtual applications.
Differences Between InstallShield Editor and InstallShield Professional and Premier Editions	Explains the differences between InstallShield Editor and InstallShield Professional and Premier Editions.
InstallShield Editor Help Library	Explains how to use InstallShield Editor to take advantage of its features to build Windows Installer packages.

AdminStudio-Specific Functionality in InstallShield Editor

When running InstallShield Editor from AdminStudio, some AdminStudio-specific functionality is enabled:

Table 14-2 • AdminStudio Functionality in InstallShield Editor

Functionality	Description
InstallShield Editor Integration with Application Catalog and the Software Repository	If you have the AdminStudio Enterprise Edition, which includes the Software Repository feature, you can add a package to the software repository via the InstallShield Editor Build process.
Repackager Integration	You can launch Repackager from the InstallShield Editor interface for conversion of the following: • Microfocus ZENworks projects (`.axt/.aot`) • Microsoft SMS Installer projects (`.ipf`) • WinINSTALL exported text files (`.txt`) • Wise Installation files (`.wse`) • Legacy Repackager projects (`.inc`) When you attempt to open any of these file types in InstallShield Editor, you can launch Repackager to perform the conversion.

Note • *AdminStudio Role permissions apply to Application Catalog functionality in InstallShield Editor. If you are not assigned to a Role with sufficient permissions, you may not be able to access some of these features.*

InstallShield Editor Integration with Application Catalog and the Software Repository

The following topics describe InstallShield's integration with Application Catalog:

- InstallShield Integration with Application Catalog Software Repository
- Quickly Opening Package in InstallShield Direct Edit Mode
- Quickly Creating and Opening a Transform File in InstallShield Direct MST Mode

InstallShield Integration with Application Catalog Software Repository

Edition • *The Software Repository feature is available in AdminStudio Enterprise Edition.*

When you have a Basic MSI project open in InstallShield Editor, AdminStudio-specific settings are listed in the **Publishing** area on the **Events** tab of the **Media > Releases** view in the Installation Designer.

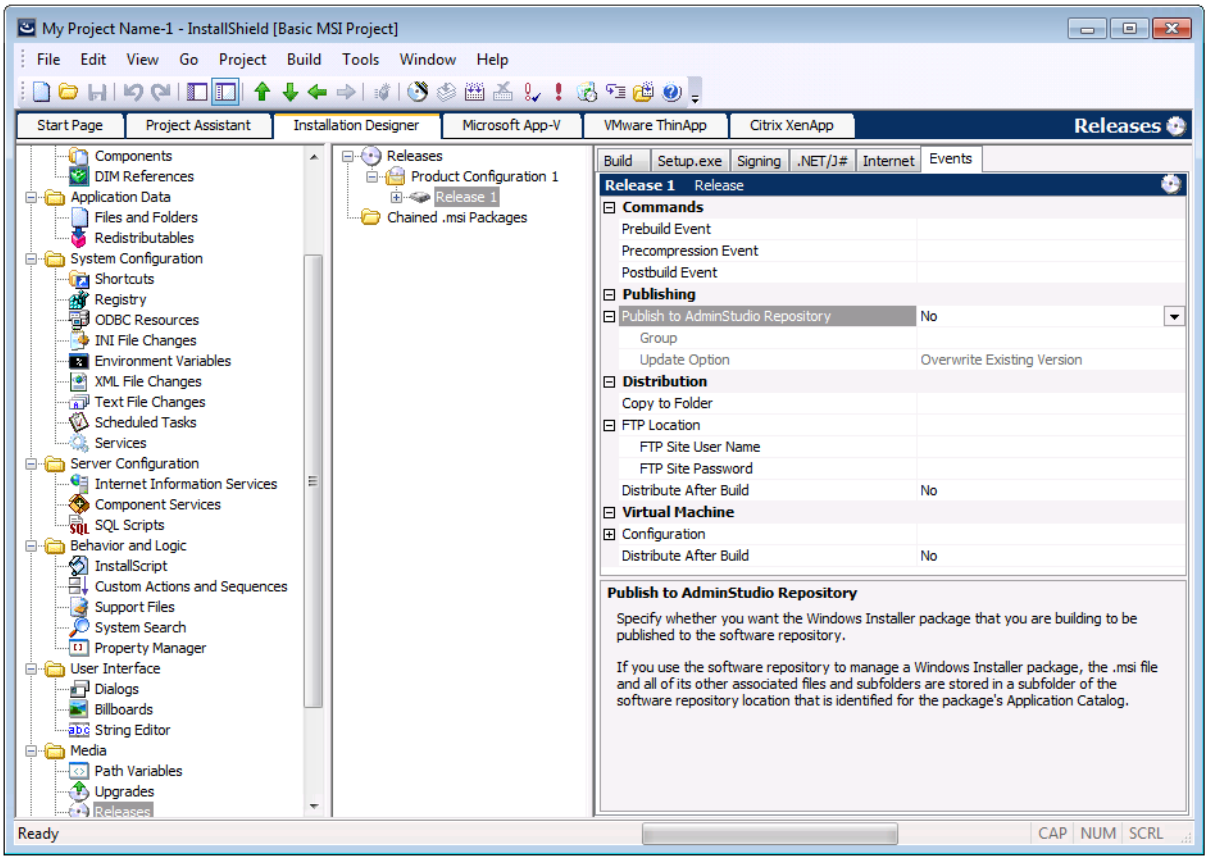


Figure 14-1: Events Tab Under Media > Releases

You can use these settings to specify information such as whether you want the Windows Installer package that you are building to be published to the software repository.

Table 14-3 • Publishing Settings on the Events Tab—for InstallShield with AdminStudio

Property	Project Type	Description
Publish to AdminStudio Repository	Basic MSI	Specify whether you want the Windows Installer package that you are building to be published to the software repository. If you use the software repository to manage a Windows Installer package, the `.msi` file and all of its other associated files and subfolders are stored in a subfolder of the software repository location that is identified for the package's Application Catalog.

Table 14-3 • Publishing Settings on the Events Tab—for InstallShield with AdminStudio

Property	Project Type	Description
Group	Basic MSI	If you select **Yes** for the **Publish to AdminStudio Repository** setting and you want to associate the Windows Installer package with one or more groups in the Application Catalog, click the ellipsis button (...) in this setting. The **Select Application Catalog Groups** dialog box opens. Select the check box for each group that you want to contain the Windows Installer package.
Update Option	Basic MSI	Specify how you want to handle the importing of the Windows Installer package that is built for the selected release into the Application Catalog. Available options are: • **New Package Version**—A new build is treated as a new package in the Application Catalog. • **Overwrite Existing Version**—A new build overwrites the existing version in the Application Catalog. • **New Package History Version**—A new build is treated as a new version of the package that exists in the Application Catalog. • **Ignore if Exists**—A new build is imported only if it does not already exist in the Application Catalog.

Adding a Package to the Software Repository via the InstallShield Editor Build Process

To add a package to the Software Repository via the InstallShield Editor Build process, perform the following steps.

Task ***To add a package to the Software Repository via the InstallShield Editor Build process:***

1. In AdminStudio, connect to an Application Catalog that has the Software Repository enabled.
2. Launch InstallShield Editor.
3. Open an InstallShield project (`.ism`).
4. In the Installation Designer, open the **Events** tab of the **Releases View**.
5. Set the **Publish to AdminStudio Repository** property to **Yes**.
6. Select the **Group** that you want the imported package to belong to.
7. Select one of the following for **Update Option**:
   - **New Package Version**
   - **Overwrite Existing Version**
   - **New Package History Version**
   - **Ignore if Exists**
8. Build the setup.

Upon completion of the build, the MSI package is published to the Software Repository. The progress messages are displayed in the Output window.

Note • *If the setup is compressed, an administrative image must be created before the package can be published to the Software Repository.*

Quickly Opening Package in InstallShield Direct Edit Mode

To quickly open a Windows Installer package in InstallShield Direct Edit Mode, open the Application Catalog **Home** tab, select the package in the Application Catalog tree and select **Edit with InstallShield** from the shortcut menu.

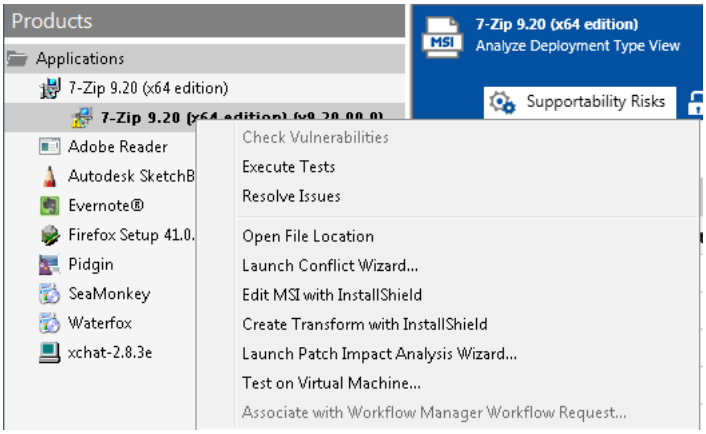


Figure 14-2: Opening a Package in InstallShield Direct Edit Mode from Application Catalog

After you have finished editing this package in InstallShield and have saved it, you can then reimport the Windows Installer package along into the Application Catalog by right-clicking on the Windows Installer package in the Application Catalog tree and selecting **Reimport Package** from the shortcut menu.

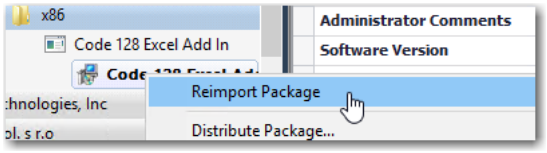


Figure 15: Reimport Package Command on Shortcut Menu

Quickly Creating and Opening a Transform File in InstallShield Direct MST Mode

You can quickly create a new transform project for a Windows Installer package by right-clicking on the package in the Application Catalog tree and then selecting **Create Transform with InstallShield** from the shortcut menu.

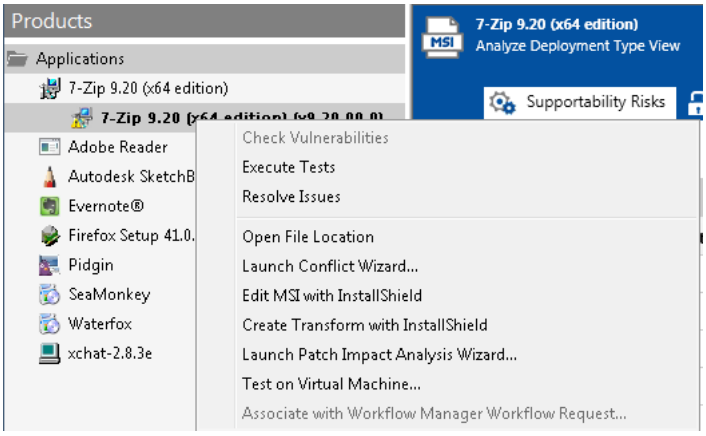


Figure 16: Create Transform with InstallShield

A new transform project (named `PackageName_ISTransform.mst`) for the selected package opens in InstallShield in Direct MST mode.

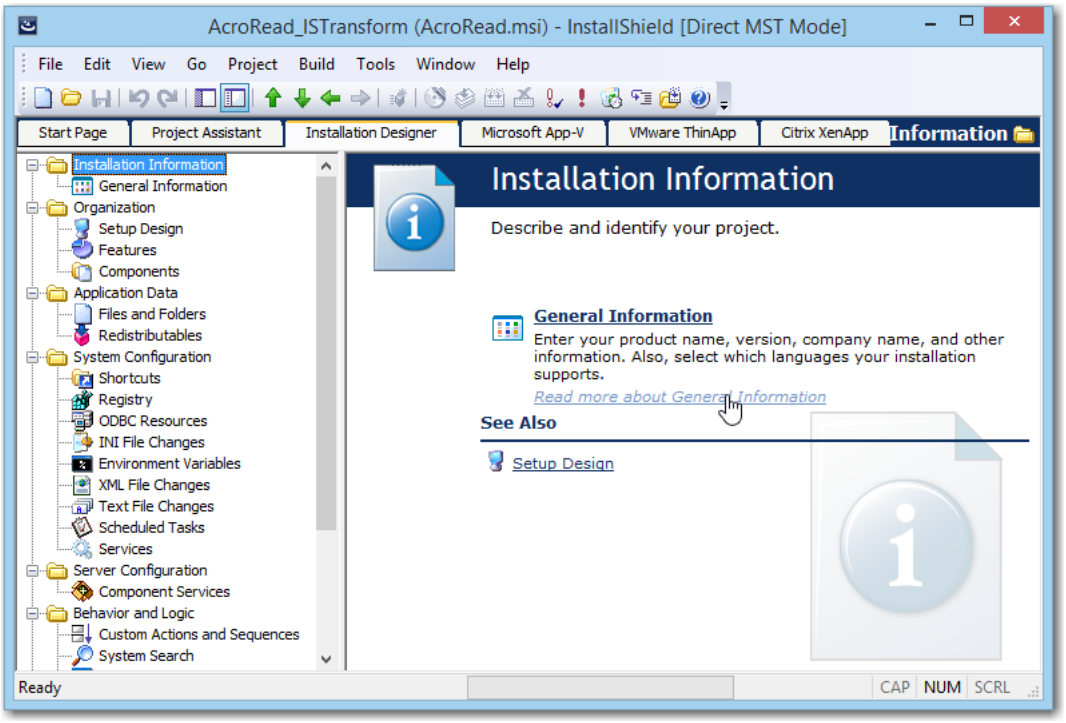


Figure 17: Transform File Open in InstallShield Direct MST Mode

After you have finished customizing this transform file in InstallShield and have saved it, you can then reimport the Windows Installer package along with its newly created transform file into the Application Catalog by right-clicking on the Windows Installer package in the Application Catalog tree and selecting **Reimport Package** from the shortcut menu. This lets Application Catalog know that you are done editing the transform file.

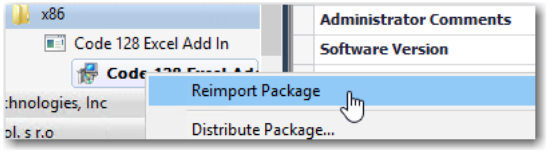


Figure 18: Reimport Package Command on Shortcut Menu

Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support

Important • *AdminStudio ThinApp support requires a separate purchase of VMware ThinApp™.*

Both AdminStudio and InstallShield Editor provide support for the conversion of Windows Installer packages to virtual packages:

- **Automated Application Converter**—You can use the Automated Application Converter to convert a Windows Installer or legacy package (or group of packages) into Microsoft App-V, VMware ThinApp, Citrix XenApp. Automated Application Converter can examine a group of selected setups and perform automated virtualization of those that can be cleanly virtualized. For those setups that cannot be cleanly virtualized (due to custom actions, etc.), Automated Application Converter can perform automated repackaging of those setups and then perform automated virtualization of those repackaged MSIs.
- **Repackager Interface**—By selecting an option on the **Repackaged Output** view, you can simultaneously build an InstallShield Editor project, a Windows Installer package, a Microsoft App-V application, a ThinApp application, and a Citrix profile from a Repackager project.
- **Microsoft App-V Assistant**—Using the Microsoft App-V Assistant, you can convert a Windows Installer package or an InstallShield project to a customized App-V application. You can modify a Microsoft App-V application's operating system requirements, files, folders, shortcuts, registry settings, isolation options, and build options.
- **ThinApp Assistant**—Using the VMware ThinApp Assistant, you can convert a Windows Installer package or an InstallShield project to a customized ThinApp application. You can configure a ThinApp application's files, folders, shortcuts, registry settings, isolation options, and build options.
- **Citrix Assistant**—Using the Citrix Assistant, you can convert a Windows Installer package or an InstallShield project to a customized Citrix XenApp profile. You can modify a Citrix XenApp profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

For more information on the capabilities of these features, see Getting Started With Application Virtualization.

Differences Between InstallShield Editor and InstallShield Professional and Premier Editions

Edition • *InstallShield Professional is included with AdminStudio Professional Editions. InstallShield Premier is included with AdminStudio Enterprise Edition.*

The InstallShield Editor that is included with AdminStudio 2022 R2 | 24.0 is based upon InstallShield Professional and Premier Editions, but it has a slightly different feature set. Those differences are explained here.

Default Project Types

In InstallShield Editor, all non-Windows Installer-based project types are disabled by default. You can enable these additional project types, such as InstallScript, on the InstallShield Editor **Options** dialog box, which is opened by clicking **Options** on the **Tools** menu.

Multilingual Runtime Language Support

InstallShield Editor includes InstallShield Premier Edition's multilingual runtime language support, which enables you to create a single installation that displays end-user text in multiple languages. If an installation will contain more than one language, you can specify whether to prompt the end user to select the run-time language, or to automatically display the language of the target system's operating system.

Using a Network Repository to Share Project Elements

A repository is a collection of common elements that can be shared and reused in different installation projects, enabling you to ensure consistency. InstallShield Editor includes not only InstallShield Professional Edition's *local* repository support, but also InstallShield Premier Edition's *network* repository support, which fosters collaboration among installation authors.

InstallShield Editor Help Library

Edition • *InstallShield Professional is included with AdminStudio Professional Edition. InstallShield Premier is included with AdminStudio Enterprise Edition.*

The InstallShield Editor Help Library gives you unified access to InstallShield Editor Help. This library's help topics contain information that assist you in finding answers with InstallShield Editor.

Open the InstallShield Editor Help Library to see a listing of the Help Library's contents.

Note • *You can also download the InstallShield documentation PDFs from https://docs.flexera.com.*

15

Customizing Installations Using Tuner

Using Tuner, you can add to, modify, or remove information from a Windows Installer package. This involves creating a transform file, where all the modifications are stored. When you install the package and transform together, your modifications are reflected in the installation.

Tuner allows you to configure the initial state of features, add or remove files from an installation, edit registry entries, configure setup properties, set Add/Remove Programs options, and configure servers for application resiliency. You can also validate Windows Installer packages and transform files to ensure they conform to Microsoft guidelines.

Tuner user documentation is presented in the following sections:

Table 15-1 • Tuner User Documentation

Section	Description
Working with Transforms	Explains how to create a transform file to customize a Windows Installer-based installation.
Validation	Explains how to compare a Windows Installer-based installation to a known set of guidelines (an evaluation file) to ensure it has been created to those guidelines.
Setup Organization	Explains how to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.
Configuring Package Content	Explains how to use a transform file to manipulate the original package contents, including files and folders, registry entries, shortcuts, INI files, ODBC resources, and NT services.
Working with Dialogs	Explains how to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences.
Configuring Additional Server Locations	Explains how to configure additional server locations.

Table 15-1 • Tuner User Documentation (cont.)

Section	Description
Changing Add/Remove Program Settings	Explains how to configure the Windows setup to give the user the option of removing, repairing, or changing the installation with the click of a button.
Customizing Setup Properties	Explains how to edit property values in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties.
Preparing Packages for Distribution	Explains how to postvalidate your transform and base Windows Installer package, and how to package the transform and base package for distribution.
Directly Editing Packages	Explains how to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.
Documenting Response Transform Creation Using the Microsoft Step Recorder Tool	Explains how to use the Microsoft Steps Recorder documentation tool to record the steps taken during transform creation.
Tuner Reference	Information on Tuner Views and dialog boxes.

When to Use Tuner vs. InstallShield Editor

Most system administrators use InstallShield Editor to import repackaged setups and convert them into Windows Installer packages. InstallShield Editor is also ideal for making changes to the package that you want reflected in all deployments of the package. However, it is recommended that you use Tuner to create transforms for changes that you only want to affect a particular deployment, rather than every installation.

Working with Transforms

The Microsoft-designated term transform refers to a specific file type used to customize a Windows Installer-based installation. A transform contains all modification information, such as whether features are installed, how they are installed, which files, shortcuts, and registry entries are included, and Windows 2000 and XP Add/Remove Programs information. Transform files use an .mst extension.

For example, you may need to customize an installation for different departments in your company. Typical business productivity suites come with a spreadsheet program, a word processor, and a presentation tool. Your accounting department may only need the spreadsheet and the presentation programs. On the other hand, the writing department may need only the word processor and the spreadsheet. A third department may need the entire suite of applications. Instead of manually setting up every person in the company, you can take the original setup of the entire suite, and create a customization project in the form of a transform to meet the needs of each department. A transform would need to be created for every configuration that you plan to use.

Once you have created a transform, you can apply it at runtime, depending on whose machine the application is being installed on. For example, in the accounting department, the transform limits the installation to include only the spreadsheet and presentation programs.

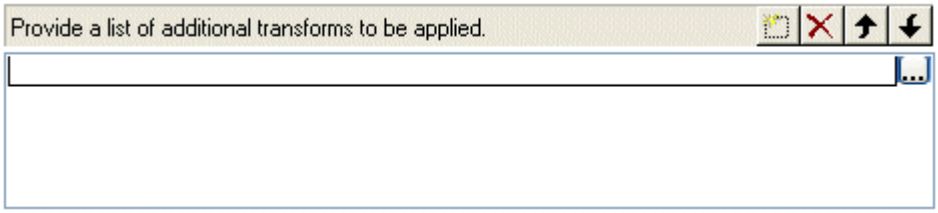
In some cases, it may be necessary to have multiple transform files for an installation. For example, a vendor may use a transform file for language-specific information. When you want to customize that MSI file, you need to include the preexisting transform so your modifications affect the entire existing package.

Creating New Transform Files

Task ***To create a new transform file:***

1. Launch **Tuner** from the AdminStudio interface. The **Tuner Start Page** opens.
2. Click **Create a new transform** on the left side of the view or select **New** from the **File** menu. The right side of the view changes to display the fields necessary to create a new transform.
3. In the **Select an MSI File** field of the **Base Windows Installer Package** area, enter the name and location of the Windows Installer package that you are customizing, or click **Browse** to locate it.
4. If there are transforms already associated with the Windows Installer package, (for example, previous customizations or transforms containing language-specific information), go to the **Provide a list of additional transforms to be applied** area and click the New button:

   A new entry appears in the list.

   When an entry appears in the list, click the Browse button (...) to the right of it and locate the transform. If multiple transforms are associated with this package, use the Move Up and Move Down buttons to specify the order in which the transforms are applied.

   Caution • *When using multiple transforms, keep in mind that the order in which they are applied is critical. For example, if you create a transform for a Windows Installer package that creates a new value for a property, and then create a second transform that changes the value created in the first transform, everything works correctly. However, if you apply the second transform first, that transform is attempting to modify the property's value, instead of creating it. That will result in an error.*

   One simple example of where this may be a problem is with the default company name. If the value is not set by default, and you set it in using the first transform, a new value for the property is created. If you create a second transform that modifies the combined original package and first transform, and the second transform changes the default company name, it is only changing the property. However, if you try to apply the second transform without the first one, Windows Installer interprets this as trying to change a null value to another value, which will result in an error.

5. By default, the transform will be created in the same directory as the Windows Installer package, and named the same as the base package with an `.mst` extension. However, if you want to change the name and/or location of the

transform, you can also do so in the **Windows Installer Transforms** area. Click **Browse** to open the **Save Customization File** dialog box.

Navigate to the directory in which you want to store the transform file you are creating. Provide the name of the transform with an .mst extension (for example, `MyTransform.mst`) and click **Save**. The dialog box closes and the path and file name appear in the edit field.

6. If you want to create a **Response Transform**, check the appropriate box. If you are using a response transform, you can specify additional command-line properties (in property name/value pairs separated by semicolons) to pass to the response transform. These must be PUBLIC properties, and only control how the dialogs are displayed during creation of the response transform. They are not persisted outside of the UI sequence during creation. For example, you can pass the property/value pair `ARPHELPTELEPHONE=1-111-111-1111` to set the value of the **Help Telephone** field of **Add/Remove Programs**. See Using Response Transforms for more information.

   You might pass a property/value pair during response transform creation to display all dialogs during an installation that may not be displayed based on your system configuration (for example, to show Windows 9x-only dialogs on a Windows NT platform). You can then make appropriate responses and have them included in your transform.

7. If you want to record the response transform creation steps in a document, select the **Run Microsoft Step Recorder to document response transform creation steps to that they can be reviewed later**. For more information, see Documenting Response Transform Creation Using the Microsoft Step Recorder Tool.

8. Ensure all the information entered is correct, and click **Create**.

   - **If you are creating a Standard Transform**, the transform file is opened in the Tuner interface, displaying the Package Validation View.
   - **If you are creating a Response Transform**, a simulated installation of the selected application begins. Step through the installation, making changes as necessary. When you reach the end of the installation sequence and click Install, the installation will exit and the Tuner interface will open your transform, displaying the Package Validation View. Your transform contains all of the changes you made during the simulated installation.

Note • *You can access information about the original MSI file and associated transforms by selecting Properties from the File menu.*

Opening Existing Transforms

On the Tuner Start Page, when you click on Open an existing transform file, the pane on the right of the interface changes. You can then specify the name and location of the base Windows Installer package, any associated transforms, and the name and location of the transform file.

Note • *Generally, you will only use this option when opening existing transforms that were created by a product other than Tuner, or created by someone other than yourself. Transforms you create using Tuner are more easily accessed through using the Open a recent transform file selection.*

Opening Recently Accessed Transforms

On the Tuner Start Page, when you click on Open a recent transform file, the pane to the right changes to a list containing your most recently accessed transforms.

From this View, you can perform the following tasks:

- To open a transform file, select a transform file and click Open.
- To view information on the transform file, select a transform file and click Properties. The Properties dialog box opens, listing details about the base MSI package and associated transforms.
- To specify the view that will appear when Tuner is started, select one of the following options:
  - Reload the last project saved when restarting Tuner
  - Make this my default Tuner Start Page Screen
  - Make Welcome my default Tuner Start Page Screen

Creating Generic Transforms

Most transforms are tied to a specific product code, meaning they can only be applied to a specific version of a product. Generic transforms, however, do not have that limitation. They can be created to apply to multiple versions of a Windows Installer package (for example, Office XP and Office 2000), or to any Windows Installer package.

Task ***To create a generic transform:***

1. Create a transform in Tuner as you would do normally.
2. With the transform project open, select Transform Summary Information from the Project menu to access the **Transform Summary** dialog box.
3. When the **Transform Summary** dialog box appears, change the validation options to reflect how you want the transform applied. If you want to create a completely generic transform, deselect all validation options.

Tip • *One use for generic transforms is to enforce standard Add/Remove Programs information on every package installed in your environment. The same transform can be used to set all relevant properties.*

Using Response Transforms

There are two ways you can create a transform file in Tuner:

- The first and the most common way is to begin by creating an empty transform and then making customization by navigating to different views in Tuner.
- The second way is by running the installation and then customizing various options available in each setup panel. The installation is only simulated and no changes take place on the user system. Tuner saves all the changes that the user has made on each panel of the setup in the transform. This type of transform is called a response transform.

Response transforms, much like installation response files, allow you to run an existing Windows Installer-based installation and capture your configurations. Unlike a response file, these changes are used as a starting point for your new transform.

For example, if you use Tuner to create a response transform, you might select certain features you want installed, the location of the installation, and company information. When the Tuner interface is opened, these values will already be set for your new transform file. You can then make further customizations as necessary.

You might want to create a response transform for an installation, and then fill in your company name as the default name, and a specific directory for installation which is different from the one suggested by the manufacturer. Further, you may want to configure a specific feature, such as clip art, to not be installed. By using the familiar installation user interface, you can quickly make your basic customizations before using the Tuner environment to refine the transform.

Viewing Transform Properties

To view properties of the transform you are currently creating or editing, select **Properties** from the **File** menu. To view the properties of a project from the Open a Recent Transform View, select the transform file and click **Properties** or right-click on the transform file and select **Properties** from the shortcut menu.

The resulting Properties dialog box provides information about the transform, including the name and location of the base Windows Installer file, and any additional transforms that are associated with this transform and MSI file.

Validation

What is Validation?

Validation is the process of comparing a Windows Installer-based installation to a known set of guidelines (an evaluation file) to ensure it has been created to those guidelines. Tuner can perform two types of validations: prevalidation and postvalidation.

- **Prevalidation** compares only the base Windows Installer package to an evaluation file. This ensures that, when starting a customization project, the initial file was created using the guidelines in that evaluation file. If it does not pass prevalidation, then the installation *may* work fine, but it may not be able to use all Windows Installer features.
- **Postvalidation** compares the base Windows Installer package and the changes made in a customization project against an evaluation file. In this case, the combination of the initial file and the subsequent modifications of the transform can produce different results than a comparison of just the base Windows Installer package. If the initial file was valid and postvalidation fails, the problems exist in the customization project. In some situations, advanced users may be able to use a transform file to make an initially invalid MSI file valid in conjunction with the transform.

What Do You Validate Against?

Tuner provides two files that you can validate against: the Windows 2000 Logo Program Suite and the Full MSI Validation Suite.

- The Windows 2000 Logo Program Suite is a subset of the Full MSI Validation Suite, and is used to certify that the installation meets the Microsoft standards for the Windows logo.
- The Full MSI Validation Suite is used to ensure that the installation meets all MSI standards.

Validation Procedure

Follow this procedure when validating an installation:

- In practice, you should prevalidate the base MSI file to ensure compliance to MSI standards before you begin creating a customization project.
- After you have finished your project, postvalidate it to make sure it is still compliant.
- If the base installation was compliant and the postvalidation fails, go back to the changes you made to determine what caused the validation problems.

For full details, consult the Windows Installer Help.

Prevalidating Windows Installer Packages

Prevalidation compares only the base Windows Installer package to an evaluation file. This ensures that, when starting a customization project, the initial file was created using the guidelines in that evaluation file.

Performing a Prevalidation

Task ***To prevalidate a Windows Installer package:***

1. After creating a new transform file and specifying the base Windows Installer package, select the Prevalidation view from the checklist. The **Prevalidation View** appears, listing the name of the base Windows Installer Package.
2. Specify or browse to the Evaluation File you want to use.
3. If you want to run specific Internal Consistency Evaluators (ICEs), specify them in the ICEs to Run text box, separating them by semicolons if there are more than one (for example, ICE07;ICE13;ICE72). Otherwise, all ICEs are used.
4. Specify the result level by checking the Show “INFO” messages, Show “WARNING” messages, and/or Show “ERROR” messages check boxes. It is highly recommended that you check at least the Error check box so you are certain you are not suppressing results that occur in invalid packages.
5. Click the MSI Validation button ( ) on the toolbar, or click the Start button in the view.

Viewing the Prevalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Prevalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row’s associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the **Direct Editor** to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Directly Editing Packages.

Note • *If no errors appear in the results (providing you are displaying errors), then the package is valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Handling Invalid Windows Installer Packages

Ideally, all Windows Installer packages will pass validation. Realistically, many will fail (generating errors). When a package fails validation, it means the package was not built to Microsoft's specifications. It does not mean the installation does not work. However, there are a few things you can do when your package has validation errors:

Table 15-2 • Methods to Resolve Validation Errors

Solution	Explanation
Use Tuner to correct validation errors.	This involves opening the base package using Tuner and creating a transform file which contains your corrections.
Contact the installation vendor.	The company that created the installation (usually the same company that created the software) may be able to resolve the validation issues and provide you with a valid setup. Be sure to provide the validation report to vendors so they know where to focus.
Reconsider using the application.	Although it might be an extreme reaction to an invalid package, there may be compelling reasons not to use an installation not built to Microsoft guidelines.
Ignore the problems and install anyway.	This is probably the most likely scenario. The invalid installation may not be worth trying to fix, or even have errors that you are concerned about. You could proceed and just use the installation as it is. From a practical standpoint, this may be your best option.

Note • *Most packages will also generate Warnings during validation. These can occur in valid packages, and many cannot be removed. Although the presence of Warnings does not make a package invalid, it is generally a good practice to eliminate Warnings (if possible).*

Postvalidating Transforms

Task ***To postvalidate a Windows Installer package and the transform you are creating:***

1. Select the Postvalidation view from the checklist. The Postvalidation view appears, listing the name of the base Windows Installer Package.
2. Specify or browse to the Evaluation File you want to use.
3. If you want to run specific Internal Consistency Evaluators (ICEs), specify them in the ICEs to Run text box, separating them by semicolons if there are more than one (for example, ICE07;ICE13;ICE72). Otherwise, all ICEs are used.

4. Specify the result level by checking the Show “INFO” messages, Show “WARNING” messages, and/or Show “ERROR” messages check boxes. It is highly recommended that you check at least the Error check box so you are certain you are not suppressing results that occur in invalid packages.
5. Click the Transform Validation button ( ) on the toolbar, or click the Start button in the view.

Viewing the Postvalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row’s associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Directly Editing Packages.

Note • *If no errors appear in the results (providing you are displaying errors), then the package and transform are valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Tip • *It is possible for a package that passed the prevalidation to fail the postvalidation. Remember changes made in the Setup Properties can affect your installation. If your package fails postvalidation, check all changes made in the Setup Properties for accuracy. To identify the original Setup Properties, you can create a new transform file that can be deleted at any time. Changes made using the Direct Editor can also affect your installation’s functionality.*

Evaluation Files and Internal Consistency Evaluators

When you prevalidate the base Windows Installer package, or postvalidate the package and your transform, Tuner runs several Internal Consistency Evaluators (ICEs) contained in the specified evaluation file. If the base package or your transform and package does not pass one of these ICEs, Tuner reports the failure. If the problem is in the base package, you can contact the software vendor to report the problem.

Setup Organization

The Organization View allows you to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.

Topics in this section include the following:

- Changing a Feature's Visibility
- Setting the Initial State of a Feature

- Editing a Feature's Description
- Setting the Default Destination
- Setting the Default Organization
- Changing the Destination Variable
- Preventing Features from Displaying During Custom Installation
- Setting Feature Properties
- Using Feature Advertisement

Changing a Feature's Visibility

Task ***To change the visibility of a feature:***

1. Under Organization in the checklist, select the Features View. This project's Features are listed in the second column.
2. Select the feature that you would like to change the visibility on. The Properties for that feature are listed.
3. Click in the Visible property in the Feature Properties grid and make a selection from the drop-down menu. Your options are:

Option	Description
Not Visible	The feature will not show up in the custom setup dialog box during installation.
Visible and Expanded	The feature will be displayed with all its subfeatures visible in the custom setup dialog box during installation.
Visible and Collapsed	The feature will be displayed in a collapsed state in the custom setup dialog box during installation.

Setting the Initial State of a Feature

Task ***To set the Initial State of a feature:***

1. Under Organization in the checklist, select the Features view. This project's Features are listed in the second column.
2. Select the feature that you would like to change the initial state of. The Properties for that feature are listed.
3. Click in the Initial State property in the Feature Properties grid and choose one of the selections from the drop-down menu. Your selections are:
   - The feature is not installed: By default, the feature will not be installed during setup.
   - The feature is installed on the local drive: By default, the feature will be installed on the local drive during setup.
   - The feature is run from source, CD, or the network: By default, the feature will be run from the source, whether it be from the installation CD or from the network.

- The feature is advertised: By default, the feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever. For more information, see Using Feature Advertisement.

Note • *The initial default settings run the Setup in a quiet mode.*

Editing a Feature's Description

Task ***To edit a feature's description:***

1. Under Organization in the checklist, select the Features view. This project's Features are listed in the second column.
2. Select the feature that you would like to change the description of. The Properties for that feature are listed.
3. Click in the Description property in the Feature Properties grid.
4. Enter the new feature description in the Description value cell.

Setting the Default Destination

Task ***To specify the Default Destination Path for an installation:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties view appears.
2. Click in the Default Destination Path property in the Product Properties grid.
3. Provide the path that you want to use as the Default Destination Path.

Caution • *Consult theProduct Properties View help topic for important information about the Default Destination Variable and how it can be affected by changing this value.*

Setting the Default Organization

Task ***To specify the default organization for the installation:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties View appears.
2. Click in the Company Name property in the Product Properties grid.
3. Enter the name you want to use as the default organization name. The organization name can be a maximum of 30 characters in length.

Changing the Destination Variable

Task ***To specify the Destination Variable that holds the Default Destination Path:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties view appears.
2. Click in the Default Destination Variable property in the Product Properties grid.
3. Select the Default Destination Variable you want to use from the drop-down menu.

Caution • *Consult the Product Properties View help topic for important information about this variable.*

Preventing Features from Displaying During Custom Installation

Task ***To prevent a feature from being displayed to your end users during a custom installation:***

1. Under Organization in the checklist, select the Features View. This project's Features are listed in the second column.
2. Select the feature that you would like to hide. The Properties grid for that feature appears.
3. Click in the Visible property in the Properties grid and select Not Visible from the drop-down menu.

The feature selected will not be visible during custom installation. Depending on the feature's Initial State, the feature may or may not be installed on the end user's system.

Setting Feature Properties

Task ***To set Feature Properties:***

1. Select Features under Organization in the checklist.
2. Select the Feature that you want to edit. The Feature Properties view appears.
3. In the Description text box, enter a description that will be displayed when a feature is clicked in the Custom Setup dialog box
4. From the Visible drop down list, select an option to specify how the feature is presented to the end user in the Custom Setup dialog. The following options are available:
   - **Visible and Collapsed:** The feature will be displayed in the Custom Setup dialog with its subfeatures collapsed by default.
   - **Visible and Expanded:** The feature will be displayed in the Custom Setup dialog with its subfeatures expanded by default.
   - **Not Visible:** The feature will not be displayed to the end user in the Custom Setup dialog.

Although an end user obviously cannot select or deselect an invisible feature, this property does not have any direct bearing on whether a feature is installed. In other words, a feature is not automatically installed if it is invisible; it just cannot be deselected if it would otherwise be installed, or selected if it should not be installed.

5. From the Initial State list, select an option to determine how (or if) the feature is installed during installation:
   - **The feature is not installed (INSTALLSTATE_ABSENT):** The feature will not be installed during setup.
   - **The feature is installed on the local drive (INSTALLSTATE_LOCAL):** The feature will be installed on the local drive during setup.
   - **The feature is run from source, CD, or network (INSTALLSTATE_SOURCE):** The feature will be run from the source, whether it is from the installation CD or from the network.
   - **The feature is advertised (INSTALLSTATE_ADVERTISED):** The feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever. For more information, see Using Feature Advertisement.

Using Feature Advertisement

Windows Installer supports many features of Windows 2000 and later platforms, including feature advertisement. This convenience enables any product feature to be in one of four installation states:

- The feature is not installed
- The feature is installed on the local drive
- The feature is run from source, CD, or the network
- The feature is advertised

When features are advertised, they are not actually installed on the local system. However, they appear to be, in that the appropriate shortcuts to launch the feature are present. The first time a user attempts to use a feature that is advertised, it is installed on the computer.

Note • *For instructions on how to specify feature advertisement and the other installation states of a feature, see Setting the Initial State of a Feature.*

Configuring Package Content

Many modifications you can make in a transform file involve manipulating the original package contents. This includes the following:

- Files and Folders
- Registry Entries
- Shortcuts
- INI Files
- ODBC Resources

- NT Services

Files and Folders

From the Files and Folders view, you can perform all file operations in Tuner. This includes viewing files in the source MSI package, adding new files, preventing files from being installed, and removing added files.

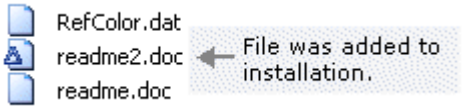
Topics in this section include the following:

- Adding Files
- Displaying Files from the Base Windows Installer Package
- Preventing Installation of Files from the MSI
- Removing Added Files
- Storing Added Files

Adding Files

Task ***To add files to an installation:***

1. Select the Files and Folders view from the checklist. The Files and Folders View appears.
2. Navigate to the location in the Source computer's directory tree that contains the file you want to add.
3. Select the file you want to add from the Source computer's files pane.
4. Drag the file to the appropriate folder in the Destination computer's folders tree. The file then appears in the Destination computer's files pane, with an icon indicating that it is an added file.



Displaying Files from the Base Windows Installer Package

Task ***To display files from the base Windows Installer package in the Files and Folders view:***

1. Select Options from the Tools menu to display the Options dialog box.
2. Select the View Settings tab. The Option Dialog's View Settings pane opens.
3. Select the Display files from the original MSI package in addition to files added in the transform check box.
4. Click OK.

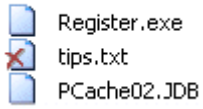
   When you return to the Files and Folders view, all files from the base Windows Installer package are displayed as well as the files added in the transform. By default, this option is enabled.

Preventing Installation of Files from the MSI

Task ***To prevent files from the base Windows Installer package from being installed:***

1. Select Files and Folders from the checklist. The Files and Folders View opens.
2. Navigate to the folder which contains the file in the Destination computer's folders tree that you want to remove from the installation.
3. In the Destination computer's files pane, right-click on the file you want to prevent from being installed and select **Remove** from the shortcut menu.

   The actual file within the Windows Installer package is not deleted—only the entry in the File table is removed. The icon for the file changes to a computer with a red "X" over it.

   If you remove a file, and later want to restore it to the installation, simply right-click the file again and select Restore.

Note • *Key files in the base Windows Installer package are denoted with a key icon, and cannot be marked for deletion. Additionally, files contained within cabinet (.CAB) files are not displayed, and therefore cannot be marked for deletion.*

Removing Added Files

Task ***To remove files you have added to an installation:***

1. Select Files and Folders from the checklist. The **Files and Folders** view opens.
2. In the Destination computer's folders tree, navigate to the folder containing the added file that you want to remove.
3. In the Destination computer's Files pane, do one of the following:
   - Select the file you want to remove and press the Delete key.
   - Right-click on the file you want to remove and select **Remove** from the shortcut menu.

Storing Added Files

When you add files to a transform, Tuner stores them in a CAB file with the same name as your transform. The added files are placed in the CAB and Tuner no longer maintains a reference to the original file location on the source computer. If you add additional files after saving the transform, the contents of the CAB file are extracted and recompressed along with the new files.

Because this mechanism relies on the presence of the CAB file, this file must be stored in the same location as the transform. If you move, modify, or delete the CAB file, Tuner will no longer be able to include the added files in the transform. You must then delete the files from the Files and Folders View and re-add them from their original locations, or locate the original CAB and place it back in the same folder as the transform.

Also, because the contents of the CAB file are uncompressed and recompressed when you add subsequent files to the installation, you must have sufficient disk space for this extraction when you save the transform.

Registry Entries

You can use the Registry view to create keys and values similar to how you use the Windows Registry Editor, or you can copy or drag and drop existing keys and values from the Source view.

Further, you can use this view to import an existing REG file using the Registry Import Wizard. You can also modify or delete registry keys that are part of the base installation. If you add new registry keys, they will always be installed.

Topics in this section include the following:

- Creating a Registry Key
- Creating a Registry Value
- Importing REG Files
- Removing Registry Information

Creating a Registry Key

Task ***To specify a registry key that will be created on the target system:***

1. Select Registry from the checklist. The Registry View opens.
2. In the Destination Computer Registry View, select the key to which you want to add a value. All existing values for that key in the Destination Computer Registry Data pane are displayed.
3. To create a subkey, right-click on a registry hive (such as HKEY_CURRENT_USER) or an existing key, point to **New** on the shortcut menu, and select **Key**.

   A new key is created with the name “NewKey*n*” (where *n* is a successive number).
4. Enter a meaningful name now to rename the key. If you want to change the name later, right-click on the key and select Rename.
5. Right-click on the new key and select whether the key is to be created on installation, deleted on uninstallation, or both.

   Choose one of the following:

Option	Description
Create Key at Install	Creates the new registry key during installation if the key does not exist on the target machine.

Option	Description
Delete Key at Uninstall	Deletes the registry key during uninstallation, regardless of whether they key existed prior to the MSI's installation. This means that the key, and all its contents and sub-keys, will be removed regardless of whether other software information that is unrelated to this MSI exists. This can have a severe impact on other programs; only select this option if you are sure that the only software affected is the base MSI.
Both Create and Delete	Both of the above scenarios will occur.

Your new key is created with an empty default string value. To modify the value name and data, see Creating a Registry Value.

6. To remove the key, right-click on it and select Delete.

Creating a Registry Value

Adding a New Value Name

Task ***To add a new value name:***

1. Select Registry from the checklist. The Registry View opens.
2. Select the key to which you want to add a value from the Destination Computer Registry view. Existing values for the key are displayed in the Destination Computer Registry Data view.
3. Right-click in the list of values and select New String Value, New Binary Value, or New DWORD Value, depending on the type of data you want to register.

   A new empty value name is created with the name "New Value #n" (where *n* is a successive number).
4. Enter a meaningful name now to rename the value. To rename the value later, right-click on the value name and select Rename.

Note • *When creating binary values, Tuner automatically converts whatever input you provide into a binary value.*

Modifying the Value Data

Each new key has an empty default string value.

Task ***To modify this or any value data:***

1. Right-click on a value name and select Modify. The Edit Data dialog box opens.
2. In the Value data text box, enter a new value or edit the existing value.
3. Click OK.

Importing REG Files

Tuner allows you to import any existing REG files that you may have created previously. To import a REG file you need to launch the Registry Import Wizard.

Task ***To import a REG file:***

1. Select Registry from the checklist. The Registry View opens.
2. Right-click on a registry hive in the Destination Computer Registry View and select Import REG File. The Welcome panel of the Import REG File Wizard appears.
3. On the Welcome panel, click Next. The Import Registry File panel appears.
4. In the Registry File text box, either type the location of the registry file or browse to it, and click Next. The Import Conflict Options panel appears.
5. Select how you would like to handle duplicate registry data during the import. You have two options:

Option	Description
Overwrite the registry data	If any conflicts exist, the old registry keys will be overwritten by the new keys.
Do not overwrite the registry data	If duplicate keys are encountered, keep the existing keys.

6. After you have selected the method, click Import to continue. The Finishing Registry Import panel appears.
7. After the registry file has been scanned, click Finish to insert all entries from the REG file into the Destination Computer Registry view. You can then modify the entries.

Removing Registry Information

Task ***To remove registry information:***

1. Select Registry from the checklist. The Registry View opens.
2. Navigate to the registry entry that you want to remove in the Destination Computer Registry View.
3. If you want to remove a value from a specific key, right-click on the value in the Destination Computer Registry Data pane and select Delete.
4. If you want to remove an entire key, right-click on the key in the Destination Computer Registry View pane and select Delete or press the Delete key.

Shortcuts

The Shortcuts view offers an integrated, visual method for adding shortcuts and program folders to the installation. Existing shortcuts can also be modified or removed.

Shortcuts can be placed in:

- folders already defined by the installation,
- standard folders that are predefined by the Windows Installer such as the Fonts folder, or
- new folders which you can create.

Each shortcut has several properties that specify the target program, hot key combination, icon, and other information necessary to launch the application. When you create a new shortcut, it will always be installed.

Note • *Shortcuts created in the transform are denoted by* *and shortcuts from the base Windows Installer package are denoted by* .

Topics in this section include the following:

- Creating Shortcuts
- Changing a Shortcut's Icon
- Change a Shortcut's Location
- Changing a Shortcut's Target
- Creating a Hot Key
- Removing Shortcuts
- Determining the Path of Changed Shortcuts

Creating Shortcuts

Task ***To create a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder in which you want to put the shortcut.
3. Right-click on the folder and select New Shortcut.
4. Provide a name for the shortcut.
5. Enter properties for the shortcut in the Properties Grid.

Changing a Shortcut's Icon

Task ***To change the icon used for a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the shortcut you want to edit.
3. Select the Icon property from the Properties Grid.

4. Click the Change Icon button in the pane below the grid. The Change Icon dialog box opens.
5. Select one of the displayed icons or browse to the file that contains the icon you want to use for the shortcut.
6. After you have selected the appropriate icon, click OK. The new icon is now displayed to the left of the Change Icon button.

Change a Shortcut's Location

Task ***To change a shortcut's location:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the shortcut you want to move.
3. Select the shortcut and drag it to another folder in the Shortcuts tree.

Changing a Shortcut's Target

Task ***To change a shortcut's target:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the appropriate shortcut.
3. Select the Target property from the Properties Grid.
4. Select the appropriate Target from the list. The Target Type that is selected affects what you should enter in the Target property field:
   - **File from MSI Package & File from File System:** Provide the full path to the application or batch file.
   - **Destination Folder:** Select a folder name from the drop-down list. The list includes available folders on the target system, from the MSI package, and from the transform.
   - **Advertised Shortcut:** Enter the feature name. You can determine the name of the feature by going to the Direct Editor and selecting the Feature table. The list of features that you can target is listed in the Features column of the table.

Creating a Hot Key

A Hot Key is a combination of keys used to launch a shortcut instead of using the mouse.

Task ***To create a hot key:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. Select the shortcut to which you want to add the Hot Key.

3. Click on the Hot Key field in the properties grid. The Hot Key dialog box opens.
4. Press the keys on the keyboard that you want to use for the shortcut. The shortcut appears in the dialog.
5. If the shortcut is correct, click OK. The dialog box closes and the shortcut's converted ASCII value appears as the value for the Hot Key.

Note • *These four fields in the Shortcuts Property Box are required for creating a Hot Key: Icon, Target, Run, and Hot Key. When you are creating the Hot Key Combination, DO NOT use a keyboard combination already adopted by Microsoft (such as Ctrl+V, which is used for Paste). Otherwise, the shortcut will not work.*

Removing Shortcuts

Task ***To remove a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. Use the Shortcuts Tree to navigate to the shortcut you want to delete.
3. Right-click on the shortcut and select Delete to remove it.

Determining the Path of Changed Shortcuts

Task ***To determine the actual path of a changed shortcut:***

1. Go to the Direct Editor and select the Directory table.
2. Find the shortcut Target directory, such as INSTALLDIR.
3. In that Directory row, find the value in the Directory_Parent column. In this example, the value is DIR26.
4. Look for a row in the Directory column that does not have a Directory_Parent entry. The directory in the row that has no value in the Directory_Parent column is the root directory. In this example, the root directory is TARGETDIR.

INI Files

Initialization (INI) files serve as a repository in which you can store and retrieve information between uses of your application. Typically INI Files contain key name-value pairs representing run-time options for applications. Some .ini files, such as Boot.ini and Wininit.ini, are used by the operating system.

INI files are divided into sections, each section containing keywords. Sections are divided by the square brackets surrounding them—[SectionName], for example. INI file keywords are the lowest level of organization in an .ini file. These keywords store data that must persist between uses of an application.

The INI Files View provides a graphical way for users to add, modify, or delete the contents of the IniFile Table. It displays the contents of the IniFile table from the source Windows Installer package and the transform.

Topics in this section include the following:

- Adding INI Files
- Importing Existing INI Files
- Adding Sections to INI Files
- Adding New Keys to INI File Sections
- Modifying INI File Keys, Values, and Actions
- Removing INI Files
- Removing Sections from INI Files
- Removing INI File Section Keys

Adding INI Files

Task ***To add an INI file to your transform file:***

1. Select INI Files from the checklist. The INI Files View opens.
2. Right-click on the appropriate destination folder in the INI File tree and select New IniFile.
3. To rename the new INI file, right click on IniFile#.ini and select **Rename** from the shortcut menu.
4. To rename the new INI section, right click on NewSection#.ini and select **Rename** from the shortcut menu.
5. With the new INI section selected, enter a Key name, Value, and Action for the default INI key value. See Modifying INI File Keys, Values, and Actions.
6. Add additional Sections and Keys, as described in Adding Sections to INI Files and Adding New Keys to INI File Sections.

Importing Existing INI Files

Task ***To import an existing INI file:***

1. Select INI Files from the checklist. The INI Files View opens.
2. Right-click on the appropriate destination folder in the INI File tree and select Import INI File from the shortcut menu. The Welcome Panel of the Import INI File Wizard opens.
3. Click Next. The Import INI File Panel opens.
4. Enter or browse to the INI file you want to import. Click Next. The Import Conflict Options Panel opens.
5. Select how you want to handle duplicate keys and values. Click Import.
6. Once the INI file has been imported, click Finish.

The imported INI file appears under the selected destination folder. You can then make further adjustments to it as needed.

Adding Sections to INI Files

Task ***To add a section to an INI file:***

1. Select the INI Files View from the checklist.
2. Right-click on the appropriate INI File in the INI File tree and select New Section from the shortcut menu. A new INI section, named NewSection1, is created under the selected INI file, complete with a default Key name, Value, and Action.
3. To rename the new section, right-click on NewSection1 and select Rename from the shortcut menu.
4. With the new INI section selected, enter a Key name, Value, and Action for the default INI key value. See Modifying INI File Keys, Values, and Actions.
5. Add additional Keys to this new INI file section as necessary. See Adding New Keys to INI File Sections.

Adding New Keys to INI File Sections

Task ***To add a new INI file key:***

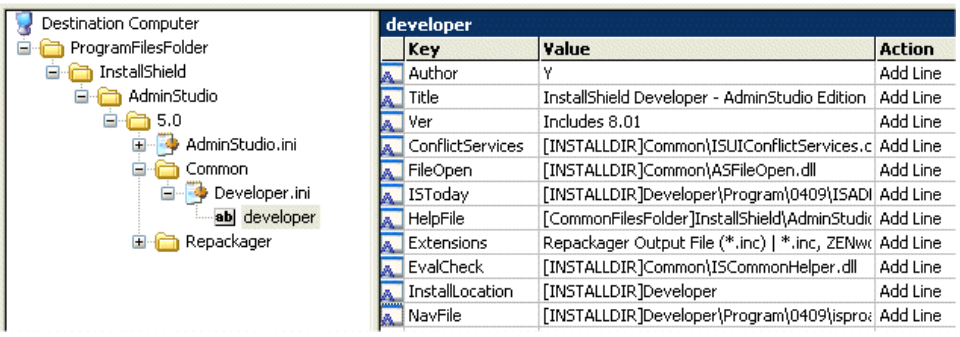
1. Select the INI Files View from the checklist.
2. Expand the listing of the INI file that you would like to edit so that all of its sections are displayed.
3. Select the INI file section that you would like to edit. That section's defined keys are listed on the right.
4. Right-click in the key listing and select Add from the menu. A new key is added to the key listing with a default Key name, Value and Action.
5. Edit the Key name, Value, and Action for this new Key. See Modifying INI File Keys, Values, and Actions.

Modifying INI File Keys, Values, and Actions

INI files contain key name-value pairs representing run-time options for applications. To define and modify the key names, values, and /or actions, perform the following steps.

Task ***To modify keys, values, and/or actions in INI files:***

1. Select the INI Files View from the checklist.
2. In the INI File tree, select the INI file that you want to edit, and expand the listing so that you can select the appropriate section. That section's keys, values, and actions are listed:



3. Modify the Key name, Value, and Action for each key, as necessary.

Option	Description
Key	The name of the key. This should be entered in the exact way you want it to appear in the target INI file.
Value	The key's value. Windows Installer properties can be used in your keyword's value. To do this, surround the property with square brackets—[INSTALLDIR], for example. For a comprehensive list of Windows Installer properties, refer to the Property Reference topic in the Windows Installer help.
Action	The action the key performs. Select this from the list in the property sheet. The available options are: • **Add Tag:** Creates a new entry or appends a new comma-separated value to an existing entry. • **Create Line:** Creates a .ini entry only if the entry does not already exist. • **Add Line:** Creates or updates a .ini entry.

Removing INI Files

Task ***To remove an INI file:***

1. Select the INI File view from the checklist.
2. From the INI File tree, right-click on the INI file you want to delete and select Remove.

Removing Sections from INI Files

Task ***To remove a section from INI file:***

1. Select the INI File view from the checklist.
2. From the INI File tree, right-click on the section you want to delete and select Remove.

Removing INI File Section Keys

Task ***To remove an INI file section key:***

1. Select the INI Files view from the checklist.
2. Select the INI File section that contains the key you want to delete from the INI Files tree.
3. Right-click the key you want to remove and select Delete.

ODBC Resources

Open Database Connectivity (ODBC) Resources are ones that involve interaction with databases. Tuner allows you to view existing ODBC Data Sources, ODBC Drivers, and ODBC Translators.

Topics in this section include the following:

- Adding New Data Sources
- Adding New ODBC Data Source Attributes
- Adding New ODBC Driver Attributes
- Editing ODBC Data Source Attributes
- Editing ODBC Driver Attributes
- Removing Existing ODBC Data Sources
- Removing ODBC Driver Attributes
- Removing ODBC Data Source Attributes

Adding New Data Sources

Task ***To add a new ODBC Data Source:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Right-click either ODBC Data Sources group or one of its children groups from the ODBC Resources tree and select New Data Source from the shortcut menu. The ODBC Data Source dialog box opens.
3. Select the required data source and click OK.

Caution • *If you are adding an ODBC Data Source that does not exist on your computer, type the name of the Data Source into the ODBC Data Source dialog. Keep in mind that adding a data source to a Windows Installer package that does not contain the corresponding driver may render the package useless.*

Adding New ODBC Data Source Attributes

Task ***To add a new ODBC data source attribute:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Select the ODBC Data Source to which you want to add a new attribute from the ODBC Resources tree. The property grid for the selected ODBC Data source opens.
3. Right-click in the property grid and select Add. A new attribute is listed, with the default values of ATTRIBUTE and NULL_VALUE.
4. Enter information for the new attribute.

Adding New ODBC Driver Attributes

Task ***To add a new ODBC driver attribute:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Select the ODBC driver to which you want to add a new attribute from the ODBC Resources tree. The property grid for the selected ODBC driver appears.
3. Right-click in the property grid and select Add from the shortcut menu. A new attribute is listed, with the default values of ATTRIBUTE and NULL_VALUE.
4. Enter information for the new attribute.

Editing ODBC Data Source Attributes

Task ***To edit an ODBC data source attribute:***

1. On the ODBC Resources View, select the ODBC data source that contains the attribute you want to modify from the ODBC Resources tree. The property grid for that data source appears.
2. In the properties grid, edit the appropriate attribute.

Editing ODBC Driver Attributes

Task ***To edit an ODBC driver attribute:***

1. On the ODBC Resources View, select the ODBC driver that contains the attribute you want to modify from the ODBC Resources tree. The property grid for that ODBC driver appears.
2. In the properties sheet, edit the appropriate attribute.

Removing Existing ODBC Data Sources

Task ***To remove an existing ODBC Data Source:***

1. Select the ODBC Resources view from the checklist.
2. Right-click on the ODBC data source you want to remove from the ODBC Resources tree and select Delete.

Removing ODBC Driver Attributes

Task ***To remove an ODBC driver attribute:***

1. Select the ODBC Resources view from the checklist.
2. Select the ODBC driver that contains the attribute you want to delete from the ODBC Resources tree.
3. In the properties sheet, right-click on the attribute you want to remove and select Delete.

Removing ODBC Data Source Attributes

Task ***To remove an ODBC data source attribute:***

1. Select the ODBC Resources view from the checklist.
2. Select the ODBC data source that contains the attribute you want to delete from the ODBC Resources tree.
3. In the properties sheet, right-click on the attribute you want to remove and select Delete.

NT Services

The NT Services view provides a way to change parameters for NT Services included in the base Windows Installer package. Topics in this section include the following:

- Setting NT Service Arguments
- Setting NT Service Dependencies
- Setting the NT Service Description
- Setting the NT Service Display Name
- Setting the NT Service Error Control Level
- Setting the NT Service Load Order Group
- Setting the NT Service Overall Install Result
- Setting the NT Service Start Type
- Setting NT Service Start Name and Password

- Setting the NT Service Type

Setting NT Service Arguments

Task ***To set NT service arguments:***

1. Select the NT Services from the checklist. The NT Services View opens.
2. Double-click the current value in the properties grid and modify it as needed.

Setting NT Service Dependencies

Task ***To set NT service dependencies:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Dependencies value in the properties grid and add the names of services or load ordering groups that must be started prior to this service.

Note • *If the dependency is on a load ordering group, the service can start if at least one member of the load ordering group is running after an attempt is made to start all load ordering group members.*

Setting the NT Service Description

Task ***To set the NT service description:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Description value in the properties grid and modify it as needed.

Setting the NT Service Display Name

Task ***To set the NT service display name:***

1. Select the NT Services view from the checklist.
2. Double-click the current Display Name value and modify it. The display name can be up to 256 characters in length.

Setting the NT Service Error Control Level

Task ***To set the NT service error control level:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Error Control value in the properties grid to access the pop-up menu. The possible values are as follows:

Value	Description
Ignore Error	Logs the error and continues with service startup.
Normal Error	Logs the error, displays an error message, and continues with service startup.
Critical Error	Logs the error (if possible) and restarts the system with the last configuration known to be good. If the last-known-good configuration is the one that caused the error, fail the startup.

Setting the NT Service Load Order Group

Task ***To set the NT service load order group:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Load Order Group value in the properties grid and modify it as needed. If this service does not belong to a group, leave this value blank.

Setting the NT Service Overall Install Result

Task ***To set the NT service overall install result:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Overall Install value in the properties grid to access the drop-down menu.
3. Select either Continue overall install if service fails to install or Fail overall install if service fails to install as this property's value.

Setting the NT Service Start Type

Task ***To set the NT service start type:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Start Type value in the properties grid to access the drop-down menu.
3. Select the desired Start Type from the following possible values:

Option	Description
Automatic	The service starts during system startup.
Manual	The service is only started when the service control manager calls the StartService function.
Disabled	The service is not started.
Start at Boot Time	The driver is started by the operating system loader. (Device driver only)
Started by the System	The driver is started by calling the IoInitSystem function. (Device driver only)

Setting NT Service Start Name and Password

Task ***To set the NT service start name and password:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Start Name value in the properties grid and provide the name under which this service will run.
3. Click the current Password value in the properties grid and provide the password associated with the Start Name.

Setting the NT Service Type

Task ***To set the NT service type:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Service Type value in the properties grid to access the drop-down menu.
3. Select the desired Start Type from the following possible values:
   - Service that Runs in its Own Process
   - Service that Shares a Process with Others
4. Optional: click the current Interact with Desktop value in the properties grid to access the drop-down menu and specify whether the service needs to interact with the desktop.

Working with Dialogs

When customizing the Windows Installer package, you may want to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences. You can do so from the Dialogs view. This view contains a list of each of the four installation modes (installation, administrative, maintenance, and patch), with the associated dialogs that appear as part of the UI sequence during the selected mode. You can enable or disable these dialogs by either the check box to the left of the dialog name, or by using the Show and Hide buttons.

Topics in this section include the following:

- Hiding Dialogs During UI Sequences
- Restoring Dialog Sequences
- Suppressing the License Agreement Dialog Box
- Disabling Custom Setups
- Editing Dialog Properties
- Dialogs View vs. Command-Line Options
- Dialog Suppression Issues

Hiding Dialogs During UI Sequences

Task ***To hide a dialog box during a UI sequence:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog you want to hide.
3. From the Dialogs list, clear the check mark next to the dialog you want to hide. Alternatively, select the dialog from the list and either click Hide or press the Space Bar.
4. If necessary set the properties for the dialog you are removing to preserve the UI sequence integrity. The Dialog Properties dialog box automatically appears if it is necessary to edit properties.

Restoring Dialog Sequences

Task ***To restore dialogs in a UI sequence:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog you want to restore.
3. From the Dialogs list, check the box next to the dialog you want to restore. Alternatively, select the dialog from the list and either click Show or press the Space Bar.

Suppressing the License Agreement Dialog Box

Using the Dialogs view, it is possible to suppress the license acceptance dialog. This involves both turning off its display, and providing the value that the setup will interpret as acceptance of the agreement.

Note • *This procedure assumes the original Windows Installer setup was created using InstallShield Editor. If another setup authoring application was used, the names of dialogs and properties may not be the same. The same general procedure still applies.*

Task ***To use a transform to suppress display of the license acceptance dialog:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the License Agreement dialog. Typically, this is only in the Installation Sequence.
3. Select the **LicenseAgreement** dialog from the Dialogs list and click Hide.
4. When the Dialog Properties dialog box appears, change the AgreeToLicense value to Yes.
5. Click OK to dismiss the dialog.

When an end user runs the installation using this transform, the License Agreement dialog will not appear.

Disabling Custom Setups

You can use the Dialogs view to prevent users from performing custom setups. However, this requires not only the elimination of the Custom Setup panel during installation, but also the Setup Types panel. Additionally, you must ensure you have configured the features you want installed, as your end user will have no way to override them.

Note • *This procedure assumes the original Windows Installer setup was created using InstallShield Editor. If another setup authoring application was used, the names of dialogs and properties may not be the same. The same general procedure still applies.*

Task ***To use a transform to disable a custom setup:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the SetupType dialog. Typically, this is only in the Installation Sequence.
3. Select the SetupType dialog from the Dialogs list and click Hide.
4. When the Dialog Properties dialog box appears, change the ADDLOCAL value to ALL.
5. Click OK to dismiss the dialog.
6. Select the CustomSetup dialog from the Dialogs list and click Hide.

7. When the Dialog Properties dialog box opens, change the _BrowseProperty value to INSTALLDIR.
8. Click OK to dismiss the dialog.

When an end user runs the installation using this transform, the user will not have the option to perform a custom setup.

Editing Dialog Properties

Task ***To edit properties for a UI sequence dialog:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog box containing properties you want to edit.
3. From the Dialogs list, select the appropriate dialog.
4. Click Properties. The Dialog Properties dialog box opens.

Note • *The Properties button is only enabled when you have selected a dialog box containing editable properties.*

5. From the Dialog Properties dialog box, double-click the value cell for the property you want to edit.
6. Change the property value as necessary, and click OK.

Dialogs View vs. Command-Line Options

Generally, you should use the Dialogs View when you are still planning on displaying some panels during UI sequences. Typically, you may want to remove the License Agreement panel or the ability for a user to perform a custom setup, and these can both be accomplished easily from the Dialogs View.

However, consider using the Windows Installer command-line options (particularly `/qn`) when you want to eliminate the user interface entirely.

Dialog Suppression Issues

When suppressing dialog box display, it is important to consider some implications of your actions. Particularly, when removing a dialog from a user interface sequence, there may be properties normally set by that dialog. For example, the LicenseAgreement dialog has a radio button which can set a property to Yes or No, depending on whether you agree to the terms in the license. The value of this property also determines whether the installation should continue. Therefore, if you remove the LicenseAgreement dialog from a sequence, you need to use the Dialog Properties dialog box to set the value of this property so the installation can continue.

Beyond setting necessary properties, you also should consider how features are displayed. For example, you may want to disable custom setups via the transform file. However, you must ensure each feature you want installed is configured to be installed; your end users will have no way to override the choices by performing a custom setup.

Configuring Additional Server Locations

If you install from a network server, and if you install features to run from the server or that will be advertised for installation on their first use, the applications may need access to the server sometime after the initial installation. The applications may also require access to the server if a file is deleted or becomes corrupt, as the application can copy the problematic file(s) automatically from the server.

Topics in this section include the following:

- Adding Additional Server Locations
- Modifying Server Locations
- Removing Server Locations
- Reordering Server Locations

Adding Additional Server Locations

Task ***To add an additional server location:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Click the New button ( ) in the Addition Server Location Paths window.
3. Enter or click the Browse (...) button and browse to the server location.

Note • *The validity of the server location is determined when the installation needs to access the server remotely. In other words, if a server is not available, or if you added an invalid server, the entry will be ignored if the resource is needed.*

Modifying Server Locations

Task ***To modify an additional server location entry:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select the server entry from the Addition Server Location Paths window and either edit the entry or use the Browse (...) button to browse to desired location.

Removing Server Locations

Task ***To remove an additional server location:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select the server entry from the Addition Server Location Paths window.
3. Click the Remove button ( ).

Reordering Server Locations

Task ***To change the order in which additional server locations are accessed:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select a server entry from the Addition Server Location Paths window.
3. Depending on whether you want to promote the server location or demote it, click the up and down arrow buttons at the top right of the view ( ). You can also use the Alt+Up Arrow and Alt+Down Arrow keyboard shortcuts.
4. Repeat with other server location entries as necessary.

Changing Add/Remove Program Settings

Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button.

Topics in this section include the following:

- Changing Add/Remove Programs Properties
- Disabling the Modify, Remove, or Repair Buttons

Changing Add/Remove Programs Properties

Task ***To change properties in Add/Remove Programs:***

1. Select Add/Remove Programs from the checklist. The Add/Remove Programs View opens.
2. Double-click the value for the property you want to change.
3. Either enter the information into the properties grid, or use the drop-down menu to select a value.

Disabling the Modify, Remove, or Repair Buttons

Task ***To disable the Modify, Remove, or Repair buttons in Add/Remove Programs in Control Panel:***

1. Select Add/Remove Programs from the checklist. The Add/Remove Programs View opens.
2. Double-click the appropriate Disable Modify/Remove/Repair Button property from the properties grid.
3. Use the drop-down menu to change the value to No.

Customizing Setup Properties

Even though Tuner provides you views to customize many areas of the Windows Installer package, it may be necessary to edit property values that are not available elsewhere. The Setup Properties view exposes the entries in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties here.

Topics in this section include the following:

- Adding Custom Setup Properties
- Adding and Editing Comments
- Removing Custom Setup Properties
- Modifying Setup Properties

Adding Custom Setup Properties

Task ***To add a new setup property:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Right-click in the properties grid and select Add. A New property is added to the bottom of the list with the a Property Name of NEW_PROPERTY and a Value of NULL_VALUE.
3. Provide a new name for your property. If you want to change it later, click on the property name to edit it.
4. Enter the property's value.

Adding and Editing Comments

Tuner supports the ability to add comments to each property available in the Setup Properties view. This provides a way to clarify what specific properties do, and to enter any important information that you may need later. The original software vendor may have used InstallShield Editor to include comments in the original Windows Installer package.

Task ***To add or edit a comment for a property:***

1. Select the Setup Properties view from the checklist.
2. Double-click in the comment column for the property to which you want to add or edit the comment.
3. Add or edit the comment as appropriate.

Removing Custom Setup Properties

Task ***To remove a custom setup property:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Right-click on the property you want to remove and select Delete.
3. Confirm the deletion.

Modifying Setup Properties

Task ***To modify the property value of a setup:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Select the property that you want to modify.
3. Double-click the property's value and edit it in the grid.

Make sure you are entering valid values when you modify properties, otherwise validation or installation errors may result.

Preparing Packages for Distribution

The final step in creating a customization involves two parts. First, you should postvalidate your transform and base Windows Installer package. This ensures that you have not introduced any errors into the installation, and may help you verify that you have corrected errors that existed in the base package. Secondly, you need to actually package the transform and base package for distribution. These tasks are accomplished using the Postvalidation view and Package view, respectively.

Topics in this section include the following:

- Copying the Installation to a Network Location
- Copying the Installation to an FTP Server
- Creating a Package Definition File (PDF)
- Creating an SMS File
- Instructing SMS to Create a Management Information Format File at Deployment Time

- Deploying Windows Installer Setup Packages with Systems Management Server 2.0
- Creating a Setup.exe File for the Package and Transform
- Additional Setup.ini Parameters

Copying the Installation to a Network Location

Task ***To copy your installation to a Network location during packaging:***

1. From the checklist, select Package, and then select Location from the second column. The Location View of the Package View opens.
2. Select the Network Location option button.
3. Specify the network location, or click Browse to locate it.

   When you create your package, the appropriate files will be copied to the network location you specified.

Note • *If the transforms are copied to the same location as the original MSI, only the transform, setup.exe, setup.ini, and Windows Installer engines are copied.*

Copying the Installation to an FTP Server

Task ***To copy your installation to an FTP server during packaging:***

1. From the checklist, select Package, and then select Location from the second column. The Location View of the Package View opens.
2. Select the FTP Server option button.
3. Specify the FTP server name (FTP Location), the UserName, and Password for the FTP server.

   When you create your package, the appropriate files will be copied to the FTP server you specified.

Creating a Package Definition File (PDF)

Task ***To create a Package Definition File (PDF):***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create Package Definition File check box.

   When you create your package, the resulting file has a .PDF extension. Here is a sample ORCA.PDF file:

   [PDF]

```
Version=1.0

[Package Definition]
Product=Orca
Version=1.20.1827.1
Comment=Microsoft
SetupVariations=Typical, Automated

[Typical Setup]
CommandName = Typical Installation
CommandLine = msiexec /i Orca.msi
UserInputRequired = TRUE
SynchronousSystemExitRequired = TRUE
SupportedPlatforms = Win 9x, Win NT (i386)

[Automated Setup]
CommandName = Automated Installation
CommandLine = msiexec /i /q Orca.msi
UserInputRequired = FALSE
SynchronousSystemExitRequired = TRUE
SupportedPlatforms = Win 9x, Win NT (i386)

[Setup Package for Inventory]
InventoryThisPackage=FALSE
```

Creating an SMS File

Task ***To create a SMS file for SMS 2.0 or later:***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create SMS file check box.

   When you create your package, the resulting file has an .SMS extension. Here is a sample ORCA.SMS file:

```
[PDF]
Version=2.0

[Package Definition]
MIFFilename=Sample.MIF
Name=Orca
Publisher=Microsoft
Version=1.20.1827.1
Language=English
Programs=Typical, Automated, Test

[Typical]
Name = Typical
CommandLine = msiexec /i Orca.msi
UserInputRequired = TRUE
UninstallKey={8FC71000-88A0-4B41-82B8-8905D4AA904C}
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)
```

```
[Automated]
Name = Automated
CommandLine = msiexec /i /q Orca.msi
UserInputRequired = FALSE
UninstallKey={8FC71000-88A0-4B41-82B8-8905D4AA904C}
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)

[Test]
Name = Test
CommandLine = msiexec /i Orca.msi EXECUTEMODE=None
UserInputRequired = FALSE
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)
```

Instructing SMS to Create a Management Information Format File at Deployment Time

Task ***To instruct SMS to create a Management Information Format (MIF) file at deployment time:***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create SMS file check box.
3. Provide the name of the application you are installing in the Install MIF Filename field.
4. Provide the name of the application to be uninstalled in the Uninstall MIF Filename field.
5. Enter the serial number for the product in the Serial Number field.

The resulting file has a .MIF extension.

Deploying Windows Installer Setup Packages with Systems Management Server 2.0

Task ***To deploy Windows Installer setup packages with SMS 2.0:***

Perform the steps detailed in the Microsoft White Paper: Deploying Windows Installer Setup Packages with Systems Management Server 2.0.

Creating a Setup.exe File for the Package and Transform

Task ***To create a Setup.exe file to begin the installation of your base package and transform:***

1. From the checklist, select Package, and then select Setup from the second column. The Setup View of the Package View opens.
2. Select the Create Installation Launcher (Setup.exe) check box.
3. Specify whether you want to include the Windows 95/98 or Windows NT MSI engines.
4. Provide any command-line arguments for your installation.
5. Save your transform. The Setup.exe file is stored in the directory specified in the Location panel of the Packaging Wizard, or in the Location view within the Package view.

Additional Setup.ini Parameters

You can modify the Setup.ini file generated by Tuner for added functionality. The following two parameters can be included in the [Startup] section of the Setup.ini:

Table 15-3 • Setup.ini File [Startup] Section Parameters

Parameter	Description
SuppressWin2k	If you add this line and set its value to “Y” or “y” (SuppressWin2k=Y or SuppressWin2k=y), Setup.exe will not display a message on the target system stating that an older version of Windows Installer is installed.
SuppressReboot	If you add this line and set its value to “Y” or “y” (SuppressReboot=Y or SuppressReboot=y), Setup.exe will delay the reboot typically required by the installation of a newer version of Windows Installer on the target machine. If the application’s setup requires a reboot, this will occur normally.

Directly Editing Packages

Windows Installer packages are relational databases consisting of dozens of interrelated tables. These tables reflect the application’s features, components, relationship between features and components, registry information, and user interface.

The Direct Editor allows you to edit values in the MSI tables of the base Windows Installer package and store them in your transform. As you change values elsewhere in your transform, those changes are reflected in the Direct Editor, and vice versa. The complete list of MSI tables contained in the installation package is displayed in the left pane. When you select a table, the contents are displayed in the right pane.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table’s columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Adding a New Record Using the Direct Editor

Task ***To add a new record in the Direct Editor:***

1. Select Direct Editor from the checklist. The Direct Editor opens.
2. Select the table you want to add a row to from the table tree.
3. Click on the last row in the table's grid, or click anywhere in the table and press the Insert key.

   A new record is added to the grid using a unique key for the record (use `ColumnName<n>` as the template). You can then modify the record.

Caution • *To ensure files are deployed on the target system, add files in the Files and Folders View, rather than to the file table. Files added to the File table are not physically added to the transform.*

Finding and Replacing Using the Direct Editor

The Direct Editor supports find and replace throughout all tables. The following commands are available:

Table 15-4 • Direct Editor Commands

Command	Description
Find (Ctrl+F)	Displays a standard Find dialog box and allows you to search for a string in all tables.
Find Next (F3)	Allows you to search for the next occurrence of a given string in all tables.
Replace (Ctrl+H)	Displays a standard Search and Replace dialog box and allows you to search for data in all the MSI tables and gives you the option to replace the data.

Launching the Direct Editor from the Validation Tab

When performing a Prevalidation or Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red. This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table's columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Documenting Response Transform Creation Using the Microsoft Step Recorder Tool

You can use the Microsoft Steps Recorder documentation tool with Tuner to automatically record the step-by-step actions that occur during response transform creation. This information, which is saved in a web archive (`.mht`) file, includes a text description of where you clicked on each screen, along with a screen capture for each click.

To enable this option, select the **Run Microsoft Step Recorder to document transform creation steps so they can be reviewed later** option on the Tuner **Create a New Transform** view.

Task ***To use the Microsoft Steps Recorder during transform creation:***

1. Launch Tuner.
2. Open the **Create a New Transform** view.
3. Next to the **Select an MSI** file field, click **Browse** and select the Windows Installer package that you want to create a response transform for.
4. Select the **Response Transform** option.
5. Select the **Run Microsoft Step Recorder to document transform creation steps so they can be reviewed later** option.
6. Click **Create**. The installer is launched.
7. Click through the installer, making the selections that you want to record in the response transform, as described in Creating New Transform Files.
8. When the response transform has been created, open the directory containing the transform file, and locate the following web archive (`.mht`) file:

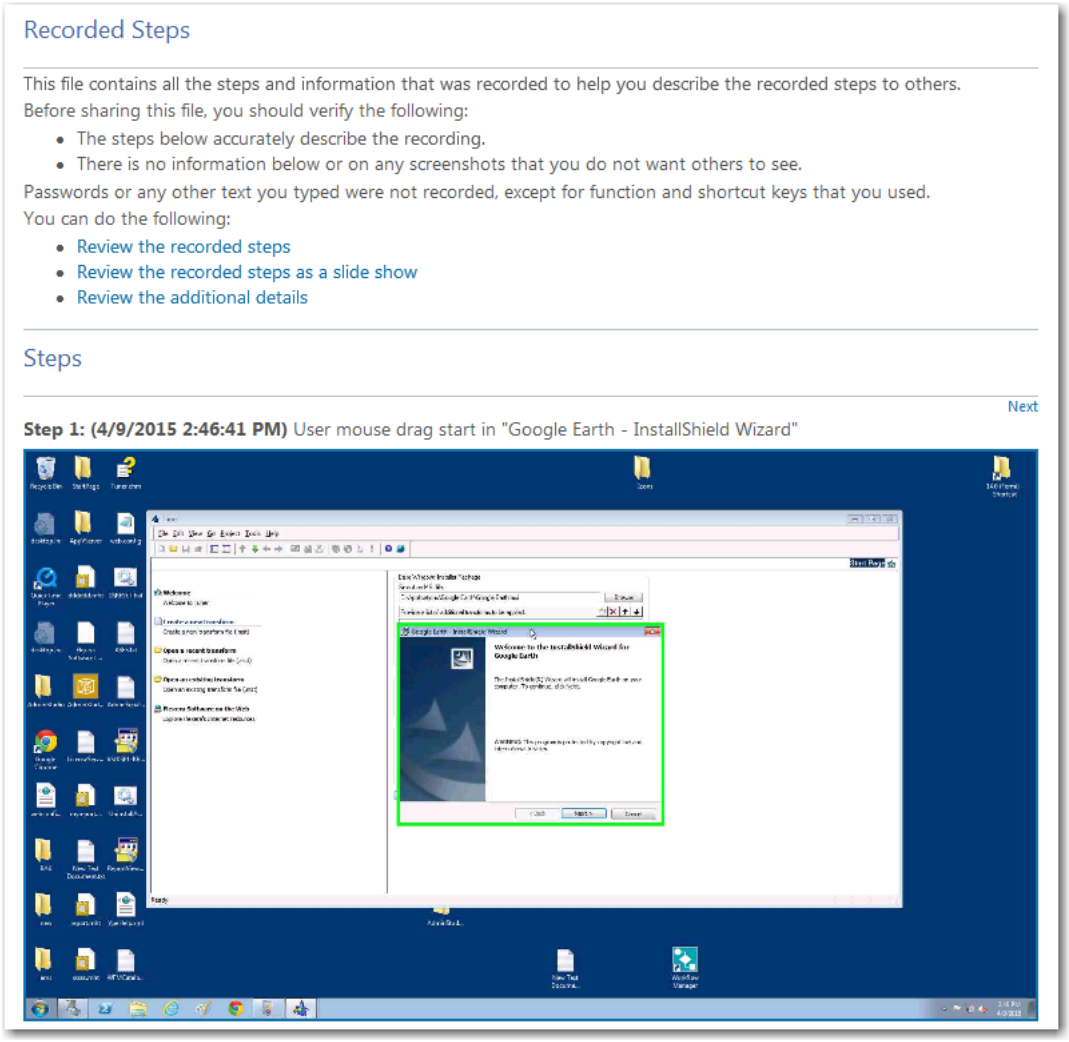
   *InstallerName*_Recording_*YYYYMMDD_TIME*.mht

For example:

```
QuickTime_Recording_20150409_1015.mht
```

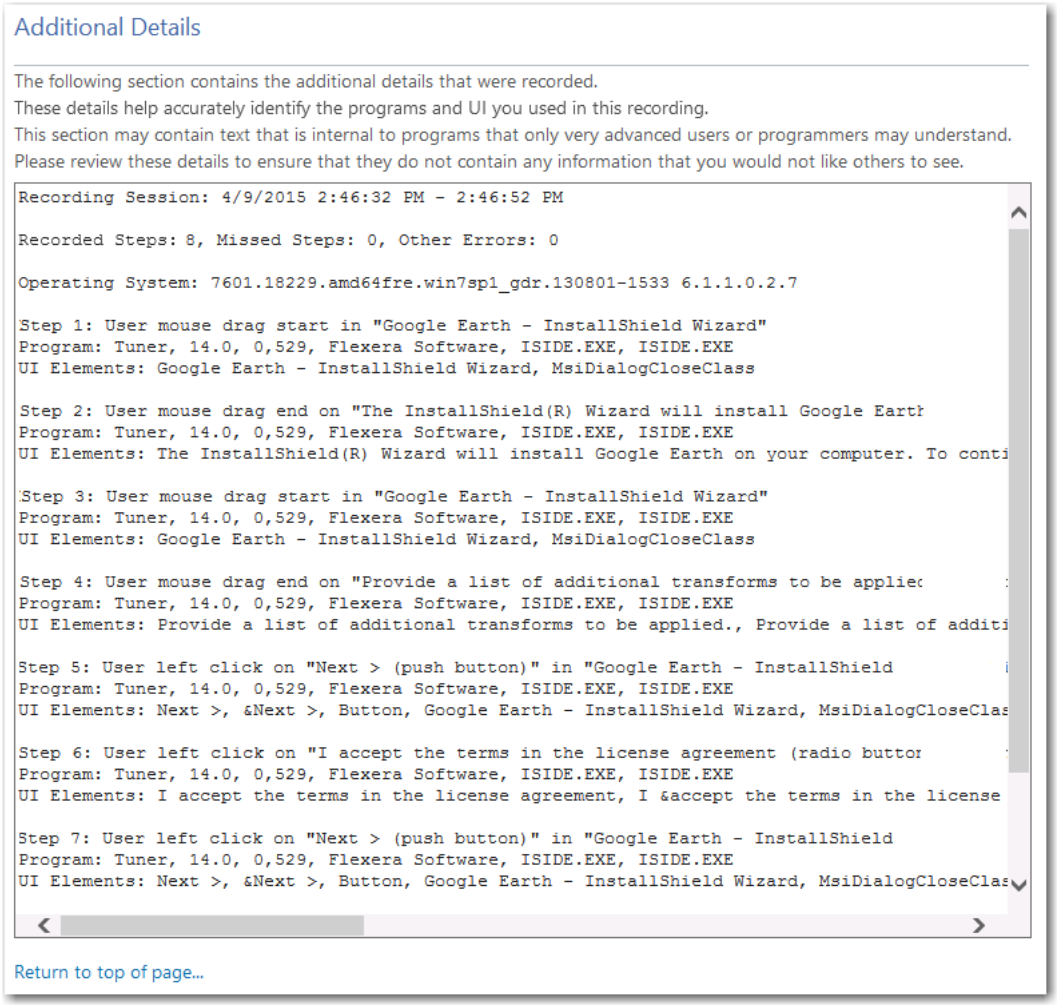
9. Double-click the file to open it. The file opens in a browser window.

10. In the **Steps** section, scroll down to view all of the steps that you performed during transform creation along with screen captures of each step.



Tip • *If you want to view all of the screens as a slide show instead of scrolling through them, click* ***Review the recorded steps as a slide show****.*

11. Review the information in the **Additional Details** area, which contains a text description of the steps that were taken, along with information that is internal to the application for which a transform was created.



Tuner Reference

Reference information for Tuner is divided into the following sections:

Table 15-5 • Tuner Reference Sections

Section	Description
User Interface Reference	Contains general information about the Tuner user interface, including menus, toolbars, keyboard shortcuts, windows, and dialog boxes.
Tuner Views	Contains detailed reference information about each Tuner view.
Import INI File Wizard	Provides reference information for the Import INI File Wizard. This is the same information accessible by clicking Help in the Wizard.
Import REG File Wizard	Provides reference information for the Import REG File Wizard. This is the same information accessible by clicking Help in the Wizard.

Table 15-5 • Tuner Reference Sections (cont.)

Section	Description
Packaging Wizard	Contains panel-by-panel reference information for the Tuner Packaging Wizard. Also includes information about additional Setup.ini options.

User Interface Reference

This book describes the user interface components, such as menu items, the toolbar, views, and dialog boxes you will encounter throughout Tuner.

- Menus and Toolbar
- View Bar
- Checklist
- Output Window
- Customize Dialog Box
- Properties Dialog Box

These topics are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog.

Menus and Toolbar

The Tuner user interface has several menus, each of which contain different commands. The functionality of each command is described below. Additionally, the Toolbar provides quick access to some of the frequently used commands; the corresponding toolbar buttons are listed with the appropriate commands.

Table 15-6 • Tuner Menus and Toolbar

Menu	Command	Shortcut	Toolbar Button	Description
File	**New**	Ctrl+N		Takes you to the Create a New Transform File area of the Tuner Start Page view.
File	**Open**	Ctrl+O		Takes you to the Open a Recent Transform File area of the Tuner Start Page view.
File	**Close**			Closes the currently open transform file.
File	**Save**	Ctrl+S		Saves the current transform file.
File	**Save As**			Prompts you to name the transform file you are saving.

Table 15-6 • Tuner Menus and Toolbar (cont.)

Menu	Command	Shortcut	Toolbar Button	Description
File	**Properties**			Displays properties for the current transform, including the name and location of the base Windows Installer package.
File	**[1], [2], [3], or [4]**			Allows you to select one of the four most recently accessed transforms.
File	**Exit**			Exits Tuner.
Edit	**Undo**	Ctrl+Z		Undoes the last action.
Edit	**Cut**	Ctrl+X		Removes the selected text to the clipboard.
Edit	**Copy**	Ctrl+C		Copies the selected text to the clipboard.
Edit	**Paste**	Ctrl+V		Pastes the contents of the clipboard to the current cursor location.
View	**Output Window**			Toggles the Output window.
View	**Check List**			Toggles the checklist.
View	**View Bar**			Toggles the View Bar.
View	**Header Bar**			Toggles the Header Bar.
View	**Toolbar**			Toggles the Toolbar.
View	**Status Bar**			Toggles the Status Bar.
Go	**Previous View**			Takes you to the previous view in the checklist.
Go	**Next View**			Takes you to the next view in the checklist.
Go	**Back**			Moves you to the last view.
Go	**Forward**			Moves you to the next view.
Go	**Start Page**			Takes you to the Tuner Start Page.
Go	**Help**			Takes you to the Help view.

Table 15-6 • Tuner Menus and Toolbar (cont.)

Menu	Command	Shortcut	Toolbar Button	Description
Go	**Package Validation**			Takes you to the Prevalidation view.
Go	**Organization**			Takes you to the Product Properties and Features views.
Go	**System Configuration**			Takes you to the Files, Registry, Shortcuts, INI Files, NT Services, and ODBC Resources views.
Go	**Application Configuration**			Takes you to the Server Locations, Setup Properties, Dialogs, and Add/Remove Programs views.
Go	**Package Preparation**			Takes you to the Postvalidation and Package views.
Go	**Additional Tools**			Takes you to the Direct Editor view.
Project	**MSI Validation**			Runs prevalidation on the MSI file.
Project	**Transform Validation**			Runs postvalidation on the MST file.
Project	**Transform Summary Information**			Launches the Transform Summary dialog.
Project	**Test**	Ctrl+T		Allows you to test your custom installation without actually installing.
Project	**Run**	Ctrl+F5		Performs the actual installation of the Windows Installer package and your transform.
Project	**Package**			Packages the transform and Windows Installer package based on the current packaging settings.
Project	**Packaging Wizard**	Ctrl+F7		Launches the Packaging Wizard.
Project	**Stop**			Halts in-progress validation or packaging.
Tools	**Customize**			Allows you to customize toolbars and menus.
Tools	**Options**			Allows you to specify the default locations for MSI and MST files.

Table 15-6 • Tuner Menus and Toolbar (cont.)

Menu	Command	Shortcut	Toolbar Button	Description
Help	**Help Library**			Launches the online help (which you are currently viewing).
Help	**MSI Help**			Brings up the MSI online help.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.
Help	**Support Central**			Connects to the AdminStudio Support website.
Help	**About Tuner**			Displays information about Tuner, including the version and copyright notice.

View Bar

The View Bar, when visible, is located at the far left of the user interface. It provides quick shortcuts to important areas of Tuner, and can be toggled on and off using the View Bar command under the View menu, or from the corresponding toolbar button.

There are three different View Bars available:

InstallShield

This View Bar gives you quick access to the Tuner Start Page and the Help view.

Checklist Steps

This View Bar provides access to each of the checklist steps, which include Package Validation, Organization, System Configuration, Application Configuration, Package Preparation, and Additional Tools views.

Views

This View Bar gives you quick access to each view in Tuner.

Checklist

The checklist is a graphical tree that shows you all of the views available in Tuner, as well as their association with other views. When you select a view, it appears in the pane to the right of the checklist; you can then customize the part of the Windows Installer package pertaining to that view. The Customization Steps Checklist is a subset of the entire checklist.

The checklist can be toggled from the View menu and from the Toolbar.

Customization Steps Checklist

To assist you with your customization, Tuner provides you with a set of steps that cover all parts of the MSI file that can be customized in your transform project. You do not have to follow these steps in sequential order, or even complete all the steps. Below is a brief description of each step, and how they fit into the customization workflow.

Table 15-7 • Customization Steps

Step	Description
Prevalidation View	This step allows you to pre-check the base Windows Installer package to ensure it is valid according to MSI standards before you take the time to create the transform file. If it is invalid, there may be unexpected results with your installation. Tuner allows you to copy the results of the prevalidation to the clipboard, where you can paste them into a message to the application's vendor. This is one of the most important steps in the Tuner workflow. Without it, all of your work might be wasted.
Organization View	This allows you to specify both the default destination that the installation will suggest, as well as the suggested default company name. You also can modify some properties of individual features in the product.
System Configuration View	This is where Tuner shows you its true customization power. You can add files and registry entries to the installation, allowing you to add company-specific files, templates, etc. to the installation. For example, if you have a set of word processing templates your company uses, you can include them you can modify or remove existing shortcuts, or add your own as necessary.
Application Configuration View	This step allows you to change functionality for Add/Remove Programs in Control Panel for Windows 2000 and later. You can also see the full spectrum of MSI properties as they exist in the base Windows Installer package, and add, modify, or remove them as necessary. Additionally, you can configure source resiliency through the Server Locations view, and customize user interface sequences through the Dialogs view.
Package Preparation View	This step allows you to postvalidate your project, verifying the original Windows Installer package and transform file, meets MSI-validation standards. You also can select the distribution options, including ones for network, FTP, single-executable, and SMS distributions.

Output Window

When you prevalidate the base Windows Installer package, postvalidate the package and your transform, or package the transform and base package, the Output Window appears at the bottom of the interface. It consists of three tabs:

- **Output**—Lists the Errors ( ), Warnings ( ), and Informational Messages ( ) that are generated during prevalidation, postvalidation, and packaging.
- **Validation**—Upon completion of the pre- or postvalidation, this tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red. You can then use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.

- **Packaging**—Displays packaging/distribution information, and displays a summary of the files copied. You can copy these results to the clipboard by right-clicking and selecting Copy.

The Output Window can be toggled from the View menu.

Customize Dialog Box

The Customize Dialog box allows you to customize which toolbars are available in the Tuner user interface, as well as the buttons that are available on the toolbars. The dialog box consists of two tab panels:

Toolbars

From the Toolbars panel, you can select viewing properties for all toolbars, such as whether tooltips are displayed, the style of the toolbar, and the size of the buttons. You can also create your own custom toolbar, onto which you can place buttons found in the Command tab panel.

Command

The Command panel allows you to customize toolbars and the menu bar. Simply drag the command or menu you want to add to the existing toolbar; it appears where you place it. To remove a command or menu, select it and drag it off the toolbar.

Properties Dialog Box

This dialog box displays properties of the transform you are currently creating or editing—including the name and location of the base Windows Installer file, and any additional transforms that are associated with this transform and MSI file.

Options Dialog Box

The Options dialog box has two tabs: File Locations and View Settings.

File Locations

Within the File Locations tab, you can specify the default location of your source MSI files. This is reflected in the New Transform and Open Existing Transform Views of the Tuner Start Page.

View Settings

From the View Settings tab, you can select whether you want to display files from the base Windows Installer package (MSI) in the Files and Folders view in addition to files added in the transform.

Transform Summary Dialog Box

The Transform Summary dialog box, available from the Project menu, allows you to configure how to handle specific errors when the transform is applied. Additionally, you can configure how the Windows Installer Service verifies whether the transform can be applied against a given package.

Suppression Options

Options in this section allow you to configure whether installations with this transform applied will continue or fail if certain errors are encountered. You can configure the following options:

Table 15-8 • Suppression Options

Option	Description
Add Existing Row	Suppresses errors resulting from adding rows that already exist.
Delete Missing Row	Suppresses errors resulting from deleting rows that do not exist.
Add Existing Table	Suppresses errors resulting from adding existing tables.
Delete Missing Table	Suppresses errors resulting from deleting tables that do not exist.
Modify Missing Row	Suppresses errors resulting from updating rows that do not exist.
Change Code Page	Suppresses errors resulting from mismatched code pages.

Validation Options

Options in this section allow you to specify how the Windows Installer Service verifies the transform can be applied to a Windows Installer package. You can configure the following options:

Table 15-9 • Validation Options

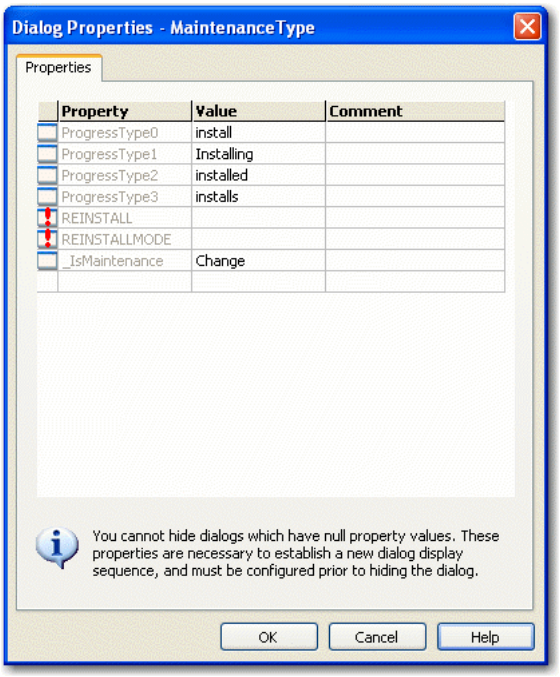
Option	Description
Same Language	If selected, the package against which the transform is applied must be the same language as the package used to create the transform.
Same Product Code	If selected, the product code for the package against which the transform is to be applied must be the same as the package product code of the package used to create the transform. If not selected, you can create a generic transform that can be applied against multiple Windows Installer packages.
Same Upgrade Code	If selected, the upgrade code of the package against which the transform is applied must be the same as the upgrade code of the package used to create the transform.
Product Version is Lower	If selected, the product version must be less than the version of the package used to create the transform. This can be combined with the Product Version is Equal option to create a “less than or equal to” comparison.

Table 15-9 • Validation Options (cont.)

Option	Description
Product Version is Equal	If selected, the product version must be equal to the version of the package used to create the transform. This can also be combined with either the Product Version is Lower option or the Product Version is Higher option, creating a “less than or equal to” or “greater than or equal to” comparison.
Product Version is Higher	If selected, the product version must be greater than the version of the package used to create the transform. This can be combined with the Product Version is Equal option to create a “greater than or equal to” comparison.
Version Checking	When using product version comparisons, you must indicate to what degree you want the comparison made. You can compare only the major versions, the major and minor versions, or the major, minor, and update versions. You can also select None to clear version checking.

Dialog Properties Dialog Box

You can use the Dialog Properties dialog box to view or change properties associated with UI sequence dialogs. The Dialog Properties dialog box is accessible by either selecting the appropriate dialog in the sequence and clicking Properties, or if you attempt to hide a dialog which has properties that must be configured.



Note *• You must ascertain the purpose of each property from the Windows Installer package, as these properties are usually custom in nature. This is especially true for properties that must be set prior to hiding a dialog from the UI sequence.*

Null Properties

You cannot hide UI sequence dialogs until you provide values for all currently null properties. These values are necessary to establish a new dialog box display sequence. Each property which must be configured prior to hiding the dialog in the sequence is marked with . Either configure the null properties, or click Cancel to return to the Dialogs view. Typically, configuring a null property involves clicking in the property's value field and selecting the value from the drop-down list.

Once you have provided a value for a null property, or if the property does not require configuration, it is denoted with .

Tuner Views

The following views are available in Tuner:

Table 15-10 • Tuner Views

Views	Subviews
Tuner Start Page	• Welcome to Tuner View • Create a New Transform View • Open a Recent Transform View • Open an Existing Transform View
Help View	
Package Validation View	• Prevalidation View
Organization View	• Product Properties View • Features View
System Configuration View	• Files and Folders View • Registry View • Shortcuts View • INI Files View • ODBC Resources View • NT Services View
Application Configuration View	• Server Locations View • Setup Properties View • Dialogs View • Add/Remove Programs View
Package Preparation View	• Postvalidation View • Package View

Table 15-10 • Tuner Views (cont.)

Views	Subviews
Additional Tools View	• Direct Editor

Tuner Start Page

From the Tuner Start Page, you can create a new transform project or open an existing one.

Select one of the following links for more information:

Table 15-11 • Tuner Start Page Subviews

View	Description
Welcome to Tuner View	General information about Tuner.
Create a New Transform View	Create a new customization, either starting with a blank transform, or by using a Response Transform based on selections from a custom installation.
Open a Recent Transform View	Select a transform previously created with Tuner.
Open an Existing Transform View	Select a transform created with a tool other than Tuner. You need to provide additional information, such as the base Windows Installer package for this transform.

Welcome to Tuner View

The Welcome view provides you with links to information on the Windows Installer service, Microsoft's integrated method of handling installations, as well as information about Tuner in the form of checklist steps. These steps cover the customization capabilities of Tuner.

Create a New Transform View

This view is displayed when you click **Create a new transform** or select **New** from the **File** menu.

This view contains the following options:

Base Windows Installer Package Subview

Table 15-12 • Base Windows Installer Package Subview

Field	Description
Select an MSI file	Enter the name and location of the Windows Installer package that you are customizing, or click Browse to locate it.
Provide a list of additional transforms to be applied	If there are transforms already associated with the Windows Installer package, (for example, previous customizations or transforms containing language-specific information), click the New button ( ). When an entry appears in the list, click the Browse button (...) to the right of it and locate the transform. If multiple transforms are associated with this package, use the Move Up ( ) and Move Down ( ) buttons to specify the order in which the transforms are applied. ***Caution*** *• When using multiple transforms, keep in mind that the order in which they are applied is critical. For example, if you create a transform for a Windows Installer package that creates a new value for a property, and then create a second transform that changes the value created in the first transform, everything works correctly. However, if you apply the second transform first, that transform is attempting to modify the property's value, instead of creating it. That will result in an error.* *One simple example of where this may be a problem is with the default company name. If the value is not set by default, and you set it in using the first transform, a new value for the property is created. If you create a second transform that modifies the combined original package and first transform, and the second transform changes the default company name, it is only changing the property. However, if you try to apply the second transform without the first one, Windows Installer interprets this as trying to change a null value to another value, which will result in an error.*

Windows Installer Transforms Subview

Table 15-13 • Windows Installer Transforms Subview

Field	Description
Provide a new project name and location (or accept the default) and click Create to create a new Customization project	Provide a new project name and location. By default, the transform will be created in the same directory as the Windows Installer package, and named the same as the base package with an .mst extension. If you want to change the name and/or location of the transform, click Browse to open the Save Customization File dialog. Navigate to the directory in which you want to store the transform file you are creating. Provide the name of the transform with an .mst extension (for example, MyTransform.mst) and click Save. The dialog box closes and the path and file name appear in the edit field.

Table 15-13 • Windows Installer Transforms Subview

Field	Description
Response Transform	If you want to create a Response Transform, select this check box. If creating a Response Transform, step through the installation, making changes as necessary. When you reach the end of the installation sequence and click Install, the installation will exit and the Tuner interface will open your transform, which contains all of the changes you made during the simulated installation.
Command line properties	If you are using a response transform, you can specify additional command-line properties (in property name/value pairs separated by semicolons) to pass to the response transform. These must be PUBLIC properties, and only control how the dialogs are displayed during creation of the response transform. They are not persisted outside of the UI sequence during creation. For example, you can pass the property/value pair ARPHELPTELEPHONE=1-111-111-1111 to set the value of the Help Telephone field of Add/Remove Programs. You might pass a property/value pair during response transform creation to display all dialogs during an installation that may not be displayed based on your system configuration (for example, to show Windows 9x-only dialogs on a Windows NT platform). You can then make appropriate responses and have them included in your transform.
Run Microsoft Step Recorder to document response transform creation steps so that they can be reviewed later	Select this option to record the response transform creation steps in a document using the Microsoft Step Recorder. For more information, see Documenting Response Transform Creation Using the Microsoft Step Recorder Tool.

Note • *You can access information about the original MSI file and associated transforms by selecting Properties from the File menu.*

Open a Recent Transform View

This view, a list containing your most recently accessed transforms, is displayed when you click on Open a recent transform. Select a transform and click Open to open it, or select Properties from the shortcut menu to view information about it (including details about the base MSI package and associated transforms).

You can also select one of the options at the bottom of the view to determine the view that is opened when you start up Tuner:

- load the last accessed transform when opening Tuner,
- make this recent list the default Tuner Start Page screen, or
- make the Welcome screen the default Tuner Start Page screen

Open an Existing Transform View

This view is displayed when you click on Open an existing transform or select File | Open. On this view, you can specify the name and location of the base Windows Installer package, any associated transforms, and the name and location of the transform file.

Generally, you will only use this option when opening existing transforms that were created by a product other than Tuner, or created by someone other than yourself. Transforms you create using Tuner are more easily accessed through using the Open a recent transform selection.

Help View

The Help view provides you instant access to this online help library. You can also access Microsoft's comprehensive Windows Installer reference library.

Package Validation View

The first recommended step in creating a transform is to perform validation on the base Windows Installer package. This helps you identify potential problems that you may or may not want to correct using Tuner.

Package validation is performed in the Prevalidation View. To continue, click Prevalidation under Package Validation in the checklist.

Prevalidation View

The Prevalidation view provides you a way to ensure the base Windows Installer package for your transform is valid. If it fails the validation test, then unexpected (and unwanted) results can occur during installation.

To begin the prevalidation process, select the evaluation file that you want to use for package validation (or click Browse to locate it), and click Start. By default, the file is evaluated using the full logo-compliant validation file, and all internal consistency evaluators (ICEs) are checked. If you just want to test specific ICEs, after you select the evaluation file, specify the ICE names, and separate them by semicolons if there is more than one.

You can toggle the information level of the displayed results by checking the Show Info Messages, Show Error Messages, and Show Warning Messages check boxes. If any errors are present, the Windows Installer Package is invalid. Warning messages highlight potential problems, but will not cause validation to fail. Informational messages display ongoing information during the validation process.

Viewing the Prevalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Prevalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Direct Editor.

Note • *If no errors appear in the results (providing you are displaying errors), then the package is valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Organization View

The Organization view allows you to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.

Each subview of the Organization View is described below:

Table 15-14 • Organization View Subviews

Views	Description
Product Properties View	When a user runs a custom installation of a Windows Installer package, the Custom Setup dialog box provides a default installation path and a default organization name. The Product Properties view provides a mechanism for changing these defaults.
Features View	Features are the building blocks of the installation. They represent distinct pieces of functionality to end users, such as program files, help files, or clip art. You can modify which features and subfeatures are installed, and how they are installed, in the Features view.

Product Properties View

This view gives you a way to specify the default path on the user's computer into which the application will be installed. You can also specify the default organization name (i.e., your company's name).

The following properties are associated with this view:

Default Destination Variable

This is the name of the variable that stores the Default Destination Path. If you change this variable, you could create errors during postvalidation. Click on the variable's value to display a combo box that allows you to select a variable.

Generally, the variable used will be INSTALLDIR or INSTALLLOCATION (both author-created variables). However, another variable can appear as the value: TARGETDIR (a Windows Installer variable). If TARGETDIR is suggested, it is strongly recommended you contact the vendor who created the original MSI and ask what was used for the Default Destination Variable. While it is possible that it was TARGETDIR, it is also possible another variable was used and Tuner cannot identify this non-standard variable.

If the incorrect variable is set here, and/or if the Default Destination Path is changed, the installation may not function properly. If that happens, you can reset the information in this view by clicking on the Reset button when the Default Destination Variable is selected.

More information can be found in the Windows Installer Help.

Default Destination Path

This location, stored within the Default Destination Variable, is the path where the application will be installed on the target machine, unless overridden during installation from the Custom Setup dialog. Click on the path's value to display a combo box of possible paths, or edit the path in the value field. It can be a hard-coded path, or it can be a Windows Installer folder property . Further levels can be separated with a backslash—for example, `ProgramFilesFolder\MyApp\Bin`.

To comply with Windows logo requirements, the application must default to a subfolder of ProgramFilesFolder, which can vary depending on the system's locale and user settings. If ProgramFilesFolder\ProductName, is specified as the default value for the Destination Folder property, then this product's files will always be installed to the logo-compliant location.

Company Name

This is the name the installation suggests for your organization during setup. If it is not set, the installer will automatically set it during installation using values from the registry. Once the value has been entered for the name of the organization, the COMPANYNAME property can be seen in the Setup Properties View.

Features View

This view allows you to change Feature properties to best suit your situation.

Features are the building blocks of an installation from an end user's perspective. They represent a specific capability of the product, such as the help files or a part of a product suite that can be installed or uninstalled based on the end user's selections. Features can be composed of subfeatures, which in turn can be composed of further subfeatures. Depending on the visibility of the "parent" feature, end users can select which portions of a feature to install in the Custom Setup dialog.

Each feature and subfeature has properties that can be modified within Tuner. These include a description of each feature (as it appears in the Custom Setup dialog box), its visibility, and its initial state . Tuner allows you to change these feature properties to best suit your situation. For example, you may want to prevent a specific application within a suite from being installed in a particular transform file. By changing its initial state and visibility, you can prevent your end user from ever seeing this feature during installation.

The Features View contains the following options:

Table 15-15 • Features View Options

Options	Description
Description	This description will be displayed when this feature is clicked in the Custom Setup dialog box during installation.

Table 15-15 • Features View Options (cont.)

Options	Description
Visible	Specifies how the feature will be presented to the end user during installation in the Custom Setup dialog. The following options are available: • **Visible and Collapsed**—This feature will be displayed in the Custom Setup dialog with its subfeatures collapsed by default. • **Visible and Expanded**—This feature will be displayed in the Custom Setup dialog with its subfeatures expanded by default. • **Not Visible**—This feature will not be displayed to the end user in the Custom Setup dialog. Although an end user obviously cannot select or deselect an invisible feature, this property does not affect whether a feature is installed. In other words, a feature is not automatically installed if it is invisible; it just cannot be deselected if it would otherwise be installed, or selected if it should not be installed.
Initial State	Provide the initial state for the feature. The end user can override this from the Custom Setup dialog. Your options are: • **The feature is not installed**—By default, the feature will not be installed during setup. • **The feature is installed on the local drive**—By default, the feature will be installed on the local drive during setup. • The feature is run from source, CD, or network—By default, the feature will be run from the source, whether it be from the installation CD or from the network. • **The feature is advertised**—By default, the feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever.

System Configuration View

The System Configuration view provides you with the ability to add additional files to a Windows Installer installation package, as well as add, remove, or modify shortcuts and registry information. Ultimately, this allows you to customize the installation to your needs, such as including company-specific templates in the correct folder during installation.

Files and Folders View

The Files and Folders View consists of four panes, representing the Source and Destination views.

Source Computer View

The Source view, located at the top, displays the folder and file structure on the user's computer.

Destination Computer View

The Destination computer's folders pane represents the folders on the target machine for the installation. The folders initially displayed for the target machine are ones used commonly in installations.

The Destination computer's files pane displays the files that are part of the installation. Initially, only files contained in the base Windows Installer package are displayed. When you add files to the package (into the transform), these also appear in this pane.

Note • *Tuner cannot display files contained within compressed files.*

A key file is a file that the Windows Installer uses to detect a component's presence. If the key file is in its proper location, the installer assumes that the entire component is installed correctly. Each component can have a key file, represented in the Files and Folders View by a key icon ( ). The key files were set by the setup author, and cannot be modified using Tuner.

Destination Computer View Tasks

The following tasks are performed in the Destination Computer View:

- **Defining a New Folder**—To define a new folder, right-click on either the Destination Computer or a predefined folder from the Destination Computer's Folders pane and select Add from the shortcut menu.
- **Adding Files to an Installation**—To add files to an installation, simply locate them within the Source view, and either drag and drop them to the appropriate destination folder, or use the copy and paste commands.
- **Removing Folders or Files**—To remove folders and files you have already added to the installation, right-click on the file or folder from the Destination view and select Delete from the shortcut menu. Predefined folders are required for installation and cannot be removed. If you add files to an installation, they are always installed.
- **Removing Files from the Base Windows Installer Package**—Files from the base Windows Installer package can be removed during installation, except for key files. To remove a non-key file from the MSI, right-click on the file and select Remove from the shortcut menu. The file is marked with an icon indicating it is not to be installed during installation.

Registry View

Similar to the Files and Folders View, the Registry view consists of four panes representing the Source and Destination views.

Source Computer View

The Source view, located at the top, displays the registry entries on the administrator's computer.

Destination Computer View

The Destination view represents the registry entries on the target machine for the installation. By default, the Destination view contains the following read-only registry hives: HKEY_CLASSES_ROOT, HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, HKEY_USERS, and HKEY_USER_SELECTABLE.

You can use the Registry view to:

- Create registry keys and values. You can do this using the same procedure you use to create registry keys and values in the Windows Registry Editor, or you can copy or drag and drop existing keys and values from the Source view to the Destination view.
- Import an existing REG file using the Import REG File Wizard.
- Modify or delete registry keys that are part of the base installation. If you add new registry keys, they will always be installed.

Setting Registry Key Options

You must be cautious when creating new registry keys so adverse results do not occur. You must specify whether the key is to be created at install, removed at uninstall, or both. These mutually exclusive options are available by right-clicking on a key.

Table 15-16 • Registry Key Options

Option	Description
Create Key at Install	Creates the new registry key during installation if the key does not exist on the target machine.
Delete Key at Uninstall	Deletes the registry key during uninstallation, regardless of whether the key existed prior to the MSI's installation. This means that the key, and all its contents and sub-keys, will be removed regardless of whether other software information that is unrelated to this MSI exists. This can have a severe impact on other programs; only select this option if you are sure that the only software affected is the base MSI.
Both Create and Delete	Both of the above scenarios will occur.

Shortcuts View

The Shortcuts view offers an integrated, visual method for adding shortcuts and program folders to the installation. Existing shortcuts can also be modified or removed. For information on how to use the Shortcuts view to create, edit, or remove shortcuts, refer to one of the following topics:

- **Shortcut Properties**—Explains the Properties that you set when creating a shortcut.
- **Shortcut Locations**—Explains how to specify the location of a shortcut, either in a predefined folder or a folder that you create.
- **Shortcut Targets**—Explains how to specify the path to the shortcut's target application, batch file, folder, or feature.

Note • *Shortcuts created in the transform are denoted by* *and shortcuts from the base Windows Installer package are denoted by* .

Shortcuts View/Shortcut Properties

When creating or editing a shortcut, you specify properties on the Shortcut Properties View.

The Shortcut Properties are described in the following table:

Table 15-17 • Shortcut Properties

Option	Description
Icon	Displays the name of the icon for this shortcut. • **To change the icon**, click the Change Icon button. On the Change Icon dialog box, click Browse to select an EXE file or DLL. All of the icons contained in that EXE or DLL file are displayed. Select the appropriate icon and click OK. On the Shortcut Properties view, the name of the icon now appears in the icon value cell and a picture of the icon appears next to the Change Icon button. • **To remove the icon**, click Clear Icon.
Description	A brief description of the shortcut. The text in this field will appear when users hover the mouse pointer over the shortcut in Windows 2000 or later.
Arguments	Any command-line arguments for the shortcut.
Target Type	The destination folder, or a file from the MSI or transform. This property can be set to one of the following values: • **File from MSI Package**—Used if the shortcut is to a file that is part of the base Windows Installer package. • **File from System**—Used if the shortcut is to a file that already exists on the target system. It may also be from the transform file. • **Destination Folder**—Used if the shortcut points to a folder rather than a file. The folder can be on the target system, from the MSI package, or from the transform. • **Advertised shortcut**—Used if you want this to be a shortcut to an "advertised" feature. Advertised features are not installed immediately during the setup process. Instead, they are installed when requested. The shortcut makes it appear that the feature is already installed, although it is not installed until the end user requests it.
Target	Path and file name for this shortcut's target. There are many potential types of targets, including applications, files, folders, printers, and computers on a network. Instead of hard-coding a path, you can use a Windows Installer folder property in square brackets—for example, *[INSTALLDIR]\MyApp.exe*. You can also target Windows Installer features, which you can use for feature advertisement. To specify a feature as a shortcut target, enter the name of the feature in the Target field. You can determine the name of the feature by going to the MSI Table Editor and selecting the Feature table. The list of features you can target with your shortcut is listed in the Features column of the table.

Table 15-17 • Shortcut Properties (cont.)

Option	Description
Run	Specifies how the item is displayed when the shortcut is double-clicked. You can select from the following options: • **Normal Window**—Launches the program in a normal sized window. • **Maximized Window**—Launches the program in full-screen view. • **Minimized Window**—Launches the program in a minimized window, visible only on the taskbar.
Working Directory	Default directory for the Save As and Open dialogs. If you are modifying an existing shortcut, or creating a new one, you can select a Windows Installer folder property from the list instead of hard-coding a path. Separate further levels with a backslash—for example, `<ProgramFilesFolders>\MyApp\Bin`.
Hot Key	The decimal value of the hot key combination for this shortcut. The Hot Key feature allows end users to launch a shortcut by pressing a combination of keys, rather than using the mouse. When you click in the Hot Key field to create or modify a shortcut, the Hot Key dialog box opens. While the Hot Key dialog box is open, press the desired hot key combination; those keystrokes are recorded. When you click OK on the Hot Key dialog box, Tuner automatically converts the keystrokes into a decimal value and enters that value in the Hot Key field. ***Caution*** • *Microsoft recommends that you do not set this value, as it may conflict with existing hot key combinations on the target machine.*

Shortcuts View/Shortcut Target

There are four shortcut Target Types you can add to a transform:

Table 15-18 • Shortcut Target Types

Target Type	Description
File from MSI Package	Used if the shortcut's target is a file that is part of the base Windows Installer package.
File from File System	Used if the shortcut's target is a file already existing on the target system.
Advertised Shortcut	Used if you want this to be a shortcut to an "advertised" feature. Advertised features are not installed immediately during the setup process. Instead, they are installed when requested. The shortcut makes it appear that the feature is already installed, although it is not installed until the end user requests it.
Destination Folder	Used if the shortcut points to a folder rather than a file.

The Target Type that is selected affects what you should enter in the Target property field:

File from MSI Package & File from File System

Under most circumstances, shortcut targets are applications or batch files. You simply provide the full path to the application or batch file in the Target property. However, after entering the target, if you leave the Shortcuts view and then return to it, you find that the target has changed. For example, you might have entered `C:\Temp\MyFolder\mytarget.exe` as the target, but it now reads `[MyFolder]mytarget.exe`. What has happened is that the path has been replaced based on entries made to the Directory table. For more information, see Determining the Path of Changed Shortcuts.

By stringing together the directories you just located in the Directory table, you can determine the path represented by `[MyFolder]` in the shortcut target. If you use the drop-down list in the Target property, you can determine the absolute values of these other directories in the same fashion.

Destination Folder

To point the shortcut to a folder rather than a file, select Destination Folder in the Target Type property, and then select a folder name from the Target property drop-down list. The Target list includes available folders on the target system, from the MSI package, and from the transform.

Advertised Shortcut

You can also target Windows Installer features, which you can use for feature advertisement. To specify a feature as a shortcut target, simply enter the feature name in the Target field. You can determine the name of the feature by going to the Direct Editor and selecting the Feature table. The list of features that you can target is listed in the Features column of the table.

Shortcuts View/Shortcut Locations

When you first navigate to the Shortcuts view, you see a set of predefined folders with existing shortcuts (if the base MSI package had shortcuts defined). You can modify or remove these shortcuts according to your needs.

If you need to create your own shortcut, you can place it in a new folder you define or in a predefined folder. Additional predefined folders that are not displayed can be accessed by right-clicking on the uppermost item in the Shortcuts explorer and selecting Show Folder.

Under the Show Folder submenu is a list of the additional predefined folders supported in the Shortcuts view. Select the folder where you want your shortcut created to have it displayed in the Shortcuts explorer. The predefined shortcut destinations are described below.

Alternately, you can create your own folder in which to place shortcuts by right-clicking either a folder or the top level Shortcuts item and selecting New Folder. To remove a folder that you have added, right-click on it and select Delete.

The following predefined folders are available for shortcuts:

Table 15-19 • Predefined Shortcut Folders

Predefined Folder Name	Description
AppDataFolder	The current user's Application Data folder.
CommonFilesFolder	The Common Files folder for the current user.
DesktopFolder	The user's desktop. Although placing a shortcut on the desktop makes it easily visible, it can also be distracting to users, so it should be used sparingly.

Table 15-19 • Predefined Shortcut Folders (cont.)

Predefined Folder Name	Description
FavoritesFolder	he Favorites folder for the current user.
FontsFolder	References the target machine's Fonts folder.
INSTALLDIR	The installation's default destination folder.
ProgramFilesFolder	References the target machine's Program Files folder.
ProgramMenuFolder	The Program menu for the current user.
SendToFolder	The user's Send To folder, which is accessible when you right-click on files. Shortcuts are placed here so users can have quick access to the target program from many file types.
StartMenuFolder	The Start menu folder for the current user.
StartupFolder	The current user's Startup folder. Shortcuts placed here automatically launch their targets whenever Windows is started.
SystemFolder	The target machine's System folder.
TempFolder	References the target machine's Temp folder (usually C:\Temp).
TemplateFolder	The current user's Template folder.
WindowsFolder	The target machine's Windows folder.

INI Files View

Initialization (INI) files serve as a database in which you can store and retrieve information between uses of your application. Typically, INI Files contain key name-value pairs representing run-time options for applications. Some INI files, such as Boot.ini and Wininit.ini, are used by the operating system.

The INI Files view provides a graphical way for users to add, modify, or delete the contents of the IniFile Table.

- The INI Files view displays the contents of the IniFile table from the source Windows Installer package and the transform .
- The view itself consists of three panes: the leftmost a tree of predefined folders from the Windows Installer package and user-defined folders from the transform.
- The top-right section displays the keys and values in the selected IniFile section. Windows Installer and transform values are distinguished by different icons in this pane.
- The lower-right pane provides information about the selected key.

Editing INI Files in Tuner

To edit an INI File in Tuner, simply expand the appropriate IniFile node in the tree. Then select the appropriate section, which appears in the upper right pane. You can then edit the keys and values appropriately. You can also insert new keys and values by right-clicking in the key and value pane and selecting Add. If you want to add a new section to an INI File, right-click on the INI File in the tree and select New Section. You can also delete an INI File, a section of an INI File, or a key by right-clicking the appropriate node or property sheet entry and selecting Remove.

For detailed instructions on performing these tasks, click on one of the following topics:

- Adding INI Files
- Adding New Keys to INI File Sections
- Adding Sections to INI Files
- Importing Existing INI Files
- Modifying INI File Keys, Values, and Actions
- Removing INI Files
- Removing INI File Section Keys
- Removing Sections from INI Files
- System Configuration View

ODBC Resources View

Open Database Connectivity (ODBC) Resources are ones that involve interaction with databases. Tuner allows you to view existing ODBC Data Sources, ODBC Drivers, and ODBC Translators.

The left pane of the ODBC Resources view contains a tree with the three root nodes: ODBC Data Sources, ODBC Drivers, and ODBC Translators. When any of these are expanded, individual Data Sources, Drivers, and Translators contained in the Windows Installer package are displayed. When selected, each of these individual nodes displays information in a property grid displayed in the upper right pane.

There are three different types of ODBC Resources available for viewing and/or modification through the ODBC Resources view:

Table 15-20 • ODBC Resource Types

Resource Type	Description
ODBC Data Sources	The source of the data (database type) and information on how to connect to that database. Common database types include Microsoft SQL Server, Microsoft Access, and Visual FoxPro. Connection information may include the name of the database, where the server that hosts it is located, and logon/password information. You can add new ODBC Data Sources from the ones existing on your computer, or delete ones you add or existing ones from the MSI. You can also add, edit, and delete ODBC Data Source attributes. If your machine does not have the ODBC Data Source that is needed by the package, you can type it into the ODBC Data Source dialog. See ODBC Resources and Adding New Data Sources for more information.

Table 15-20 • ODBC Resource Types (cont.)

Resource Type	Description
ODBC Drivers	These are libraries that implement functions involving ODBC. Each database type has its own ODBC driver. You can add only those Data Sources for which ODBC Drivers exist in the MSI package. You can add, edit, or delete new attributes for ODBC Drivers, and you can edit or delete all attributes except for File, Setup File, and Feature. See Adding New ODBC Driver Attributes and Editing ODBC Driver Attributes for more information.
ODBC Translators	These translate one form of raw data into another form that can be used with a specific database type. For example, an ODBC translator may convert from one code package to another. You can only view the contents of an ODBC Translator and cannot add, delete or modify them.

Note *• Only ODBC Data Source attributes are editable; ODBC Drivers and ODBC Translators are provided in read-only form.*

NT Services View

The NT Services view provides a way to change parameters for NT Services included in the base Windows Installer package.

Note *• NT services cannot be added to a setup using Tuner. You can only make modifications to services in the base Windows Installer package.*

The following options can be modified:

Table 15-21 • NT Services View Options

Option	Description
Name	This property contains the name of the service to install. This property may have the same value as the Display Name, but is used by the installer in a different way.
Display Name	The name of the service as it appears in user interfaces (such as the name used under the NT Services control panel). This string can be a maximum of 256 characters in length. It may be the same as the Name property.
Service Type	There are two service types available: • Service that runs in its own process • Service that shares a process with others
Interact with Desktop	Although uncommon, some services need to interact with the desktop to display message or dialog boxes for the user. If this service requires this functionality, this property's value is set to Yes.

Table 15-21 • NT Services View Options (cont.)

Option	Description
Start Type	The value in this property dictates when the service is started. The possible values are: • **Automatic**—The service starts during system startup. • **Manual**—The service starts when the service control manager calls the StartService function. • **Disabled**—The service cannot be started. There are two additional values, available only for driver services: • **Start at boot time**—The device driver is started by the operating system loader. • **Started by the system**—The device driver is started by calling the IoInitSystem function.
Error Control	This property specifies what action is taken by the startup program should the service fail to start properly during startup. The available values are: • **Ignore Error**—Logs the error and continues startup. • **Normal Error**—Logs the error, displays a message box informing the user of the problem, and continues startup. • **Critical Error**—Logs the error, if possible, and restarts the system with the last-known-good configuration. If the last-known-good configuration caused the failure, the startup operation fails.
Overall Install	This property's value specifies how the installation handles a situation when this service cannot be installed for some reason. There are two possible resolutions: • Continue overall install if service fails to install • Fail overall install if service fails to install
Load Order Group	The value of this property is a string that names the load ordering group of which this service is a member. If the service does not belong to a load order group, this value should be either an empty string or NULL.
Dependencies	A list of names of services or load ordering groups that the system must start prior to starting this service.
Start Name	The name under which the service is logged on. Leaving this field blank causes the service to be installed for the LocalSystem account. See the Windows Installer help topic ServiceInstall Table for information on the format for the StartName value.
Password	The password associated with the start name. Most services will have a blank value for this property.

Table 15-21 • NT Services View Options (cont.)

Option	Description
Arguments	This property contains any command-line arguments or properties required to run the service.
Description	This property contains a localizable description of the service. It is typically set by the setup author.
Feature	This read-only property contains the name of the feature with which this service is associated.

Application Configuration View

Application Configuration in Tuner involves adding or modifying properties that affect your setup as well as specifying properties for Add/Remove Programs in Control Panel for Windows 2000 and XP. You can also configure source resiliency using the Server Locations view, and customize user interface sequences from the Dialogs view.

The Windows 2000 and Add/Remove Programs in Control Panel differs from the previous Windows operating systems in many ways. Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button. Windows 2000 users are also be able to access additional information in Add/Remove Programs not available on previous platforms. With this information, it is easier for your end users to find technical support links, phone numbers, product update information, and information about your company.

Add/Remove Programs functionality can also be disabled to limit the number of end users who have access this feature.

Server Locations View

If you install from a network server, and if you install features to run from the server or that will be advertised for installation on their first use, the applications may need access to the server sometime after the initial installation. The applications may also require access to the server if a file is deleted or becomes corrupt, as the application can copy the problematic file(s) automatically from the server.

To ensure that users always have access to an available network server for these circumstances, you can copy the administrative installation to one or more additional servers, and then specify those servers from within this view. If the primary server should become unavailable, the Windows Installer will attempt to connect to the other servers specified here, in the order they are specified. If no server is found, the Windows Installer will prompt the user to specify the location of the server.

Setup Properties View

Even though Tuner provides you views to customize many areas of the Windows Installer package, it may be necessary to edit property values that are not available elsewhere. The Setup Properties view exposes the entries in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties here.

Properties exist in two formats: Private and Public. Private properties are set by the software vendor or by the Windows Installer during installation and cannot be altered. Private properties are always lowercase, and appear in Tuner in grayed out text. Public properties, which are always in capital letters, can also be set by the software vendor, but can be edited. Tuner also allows the addition of Public properties to the transform. The Public properties that you create can be edited or removed as necessary, whereas preexisting Public properties can only be edited.

Caution • *Before you begin changing properties in the Setup Properties view, ensure you know exactly what you are doing. The changes you make may cause validation errors, installation errors, or other unforeseen problems.*

Dialogs View

When customizing the Windows Installer package, you may want to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences. You can do so from the Dialogs view.

This view contains a list of each of the four installation modes (installation, administrative, maintenance, and patch), with the associated dialogs that appear as part of the UI sequence during the selected mode. You can enable or disable these dialogs by either the check box to the left of the dialog name, or by using the Show and Hide buttons.

If you hide a dialog that appears in more than one sequence, the dialog is hidden during all sequences.

Note • *During each installation mode, Windows Installer displays a Wizard containing a sequence of panels. However, the underlying Windows Installer technology actually uses a series of dialogs displayed in sequence. During runtime, they are referred to as panels (as with other Wizards); at design time, they are individual dialogs that can be enabled or disabled as necessary.*

Add/Remove Programs View

Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button.

You can set the following options from the Add/Remove Programs view:

Table 15-22 • Add/Remove Programs View Options

Option	Description
Publisher URL	Contains a URL for the publisher's home page. Corresponds to the ARPURLINFOABOUT property in the Setup Properties view.
Product Info and Update URL	Contains a URL that links to update information for the application. Corresponds to the ARPURLUPDATEINFO property in the Setup Properties view.
Help URL	Contains the Internet address for technical support. Product maintenance applets display this value. Corresponds to the ARPHELPLINK property in the Setup Properties view.

Table 15-22 • Add/Remove Programs View Options (cont.)

Option	Description
Help Telephone	Contains the telephone number that users can call for assistance with the product. Corresponds to the ARPHELPTELEPHONE property in the Setup Properties view.
Contact Person	Contains a the name of the person to contact for help or information about the product. Corresponds to the ARPCONTACT property in the Setup Properties view.
Comments	Contains additional information that is provided for the user. Corresponds to the ARPCOMMENTS property in the Setup Properties view.
Disable Modify Button	Provides a way to prevent users from running the application setup to modify the product's installation. Corresponds to the ARPNOMODIFY property in the Setup Properties view.
Disable Remove Button	Provides a way to prevent users from running the application setup to remove (uninstall) the product from the user's computer. Corresponds to the ARPNOREMOVE property in the Setup Properties view.
Disable Repair Button	Provides a way to prevent users from running the application setup to repair missing or corrupt product files. Corresponds to the ARPNOREPAIR property in the Setup Properties view.

Package Preparation View

The final step in creating a customization involves two parts. First, you should postvalidate your transform and base Windows Installer package. This ensures that you have not introduced any errors into the installation, and may help you verify that you have corrected errors that existed in the base package. Secondly, you need to actually package the transform and base package for distribution.

These steps are carried out in the following Package Preparation View subviews:

Table 15-23 • Package Preparation View Subviews

Views	Description
Postvalidation View	Allows you to ensure that your Windows Installer package is valid. The difference is that it also checks your newly created transform to make sure it is valid in relation to the base package.
Package View	When you have finished with your transform and postvalidation, the last step is to prepare the overall package so you can distribute it to your users (using a third-party tool such as Microsoft SMS).

Postvalidation View

Much like the Prevalidation view, the Postvalidation View allows you to ensure that your Windows Installer package is valid. The difference is that it also checks your newly created transform to ensure it is valid in relation to the base package. You can run the same internal consistency evaluators in the evaluation file, and receive the report back on the overall package and transform validity. By default, all ICEs are checked for the specified evaluation file.

You can select the information level of the displayed results by checking the Show Info Messages, Show Error Messages, and Show Warning Messages check boxes. If any errors are present, the Windows Installer package is invalid. Warning messages highlight potential problems, but will not cause validation to fail. Informational messages display ongoing information during the validation process.

If you started off with a valid Windows Installer package, yet postvalidation fails, it is likely your problems relate to changes you made in the transform. Make sure you look at the Evaluation Files and Internal Consistency Evaluators topic to see what each ICE message means. You can also consult the online MSI Help reference, available from the Help Menu, for more details.

It is also possible to start off with an invalid base package, but the postvalidation does not have any errors. If this happens, the properties you changed in your transform can bring the overall package and transform up to a valid package.

Output of the postvalidation appears in the Output and Validation tabs of the Output Window.

Viewing the Postvalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Direct Editor.

Note • *If no errors appear in the results (providing you are displaying errors), then the package and transform are valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Tip • *It is possible for a package that passed the prevalidation to fail the postvalidation. Remember changes made in the Setup Properties can affect your installation. If your package fails postvalidation, check all changes made in the Setup Properties for accuracy. To identify the original Setup Properties, you can create a new transform file that can be deleted at any time. Changes made using the Direct Editor can also affect your installation's functionality.*

Package View

When you have finished with your transform and postvalidation, the last step is to prepare the overall package so you can distribute it to your users (using a third-party tool such as Microsoft SMS). Tuner provides several different packaging options that can be used individually or in conjunction with one another. Note that Tuner does not actually perform the distribution; rather, it gathers the necessary files together, and copies them to a location you specify. You can then use some of the standard software distribution tools to roll out your customized package.

The three subviews contained within the Package view are:

Table 15-24 • Package View Subviews

View	Description
Package View/Location View	Place the transform and base Windows Installer file on either an FTP server or network location.
Package View/Setup View	Package the transform and base Windows Installer file with an executable launcher (Setup.exe) to begin the installation. You have the option of including the MSI Engine for the appropriate platform to ensure Windows Installer functionality. Setup.exe uses information contained in setup.ini to determine the package, associated transform, and any command-line parameters.
Package View/SMS View	Package and prepare the transform and Windows Installer file for distribution using Microsoft SMS. Tuner can create a PDF file and/or an SMS file for your package and transform.

Note *• These three subviews are also available through the Packaging Wizard, which can be accessed from the Project menu by selecting Package.*

Package View/Location View

It is from the Location View that you specify information regarding where to place the transform and initial Windows Installer file. You have the option to copy files to a network location, including performing an administrative installation, or copy files to an FTP server.

Package View/Setup View

In the Setup view, you can set options that are used to create a Setup.exe file to launch both your transform and Windows Installer package. You have the option to include the MSI engine for Windows 95/98 and NT, as well as include any command-line arguments for the installation.

Package View/SMS View

In many cases, you may want to perform distribution of the transform and Windows Installer package using Microsoft SMS. Tuner provides a way to create both a Package Definition File (PDF) or an SMS file in the SMS View.

Select the file type(s) you want to create. If you create an SMS file, Tuner can instruct SMS to create a Management Information Format (MIF) file when SMS deploys the package and transform. If you want to do this, provide the Install MIF Filename, Uninstall MIF Filename, and serial number.

For more information about Microsoft SMS, consult the SMS documentation.

Additional Tools View

Tuner includes an extremely flexible additional tool: the Direct Editor. Using this tool, you can directly edit the Windows Installer tables that make up the Windows Installer package. This provides you with extremely granular control over the transform you are creating.

Direct Editor

Windows Installer packages are relational databases consisting of dozens of interrelated tables. These tables reflect the application's features, components, relationship between features and components, registry information, and user interface.

The Direct Editor allows you to edit values in the MSI tables of the base Windows Installer package and store them in your transform. As you change values elsewhere in your transform, those changes are reflected in the Direct Editor, and vice versa. The complete list of MSI tables contained in the installation package is displayed in the left pane. When you select a table, the contents are displayed in the right pane.

Working Directly with MSI Tables

The Tuner Direct Editor provides the ability to work directly with MSI tables. This includes the ability to edit the contents, as well as find and replace values.

Tip • *When viewing or editing specific tables, pressing F1 launches the Microsoft Windows Installer help system to the appropriate table, if it is a standard Windows Installer table. When F1 is pressed while viewing a non-standard table, the Windows Installer help system launches to its default topic. Consult the software vendor for information about custom tables.*

Table Functionality

The following functionality is available for tables:

Table 15-25 • Direct Editor Table Functionality

Function	Keyboard Shortcut	Description
Add Records	Insert	Adds a new record to the table.
Delete Records	Del	Deletes the selected record after user confirmation. Referential integrity is not maintained.
Cut Row(s)	Ctrl+X	Enables users to cut single or multiple rows or cells in the grid to the clipboard.
Copy Row(s)	Ctrl+C	Copies the selected cell or row in the grid to the clipboard.

Table 15-25 • Direct Editor Table Functionality (cont.)

Function	Keyboard Shortcut	Description
Paste Row(s)	Ctrl+V	Pastes the contents of the clipboard into a given cell or row(s).

Editing Tables by Launching the Direct Editor from the Validation Tab

Upon completion of a Pre- or Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table's columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Import INI File Wizard

Tuner allows you to import any existing INI files that you may have created previously. To import a INI file, you need to launch the Import INI File Wizard.

The Wizard consists of the following panels:

- Welcome Panel
- Import INI File Panel
- Import Conflict Options Panel
- Finishing INI File Import Panel

Within the INI Files View, right-click on a folder under the Destination Computer node (or right-click on the Destination Computer node to add a folder first), and then select Import INI File. The Wizard that appears prompts you for the location of the INI file, as well as what to do when there are conflicts arising from duplicate values. When import occurs, Tuner merges the contents of the INI file with existing INI file data.

Welcome Panel

The Import INI File Wizard allows you to import data contained in an INI file into your transform.

Import INI File Panel

From this panel, you need to specify the name of the INI file (.ini) you want to import into your transform. Alternatively, click Browse and navigate to it.

Import Conflict Options Panel

The Import Conflict Options panel allows you to specify how you want to handle duplicate INI file data.

Select one of the following options for the Wizard to use to determine how to handle these conflicts:

Table 15-26 • Import Conflict Options

Option	Description
Overwrite the data in the IniFile table	If conflicts exist, the Wizard will overwrite the INI file keys and values with any duplicate keys from the registry file (.reg).
Do not overwrite the data in the IniFile table	If duplicate keys and values are found, the Wizard will retain the existing INI file data and not overwrite it.

Click Import to import the .ini file. When the file has been imported, the Finishing INI File Import Panel is displayed.

Finishing INI File Import Panel

This panel appears following import of the .ini file. Click Finish to exit the Wizard and return to the INI Files View.

Import REG File Wizard

Tuner allows you to import any existing REG files that you may have created previously. To import a REG file, you need to launch the Import REG File Wizard.

The Wizard consists of the following panels:

- Welcome Panel
- Import Registry File Panel
- Import Conflict Options Panel
- Finishing Registry Import Panel

To launch the Import REG File Wizard, go to the Registry View, right-click on one of the registry hives or on a registry key you have added, and select Import REG File from the shortcut menu.

The Wizard that appears prompting you for the location of the registry file, as well as what to do when there are conflicts arising from duplicate keys. When import occurs, Tuner merges the contents of the REG file with existing registry data.

Welcome Panel

The Import REG File Wizard allows you to add registry data contained in a registry file (.reg) into your transform.

Import Registry File Panel

From this panel, you need to specify the name of the registry file (.reg) you want to import into your transform. Alternately, click Browse and navigate to it.

Import Conflict Options Panel

The Import Conflict Options panel allows you to specify how you want to handle duplicate registry keys and values.

Select one of the following options for the Wizard to use to determine how to handle these conflicts:

Table 15-27 • Import Conflict Options

Option	Description
Overwrite the registry data	If conflicts exist, the Wizard will overwrite the registry keys and values with any duplicate keys from the registry file (.reg).
Do not overwrite the registry data	If duplicate keys and values are found, the Wizard will retain the existing registry data and not overwrite it.

Click Import to import the .reg file. When the file has been imported, the Finishing Registry Import Panel opens.

Finishing Registry Import Panel

This panel appears following import of the .reg file. Click Finish to exit the Wizard and return to the Registry View.

Packaging Wizard

The Packaging Wizard provides a way to step through the packaging process for the transform and Windows Installer package. Please note that this packaging merely places the installation on a network location or FTP server, creates a Setup.exe file, and/or creates files for SMS distribution. It does not actually distribute the installation to client machines. To invoke the Packaging Wizard, select Packaging Wizard from the Project menu, or select the Packaging Wizard button ( ) from the toolbar.

The Packaging Wizard consists of the following four panels:

- Location Panel
- Setup.exe Panel
- SMS Panel
- Packaging Summary Panel

Location Panel

The first panel of the Packaging Wizard allows you to specify the location to store the installation files (including transforms). If you select Network Location, you can specify or browse to the directory location.

Alternately, you can copy the installation files to an FTP server. If you select this option, you must specify the FTP location (URL), the user name under which to log in, and the password.

Setup.exe Panel

This Packaging Wizard panel allows you to include a setup.exe launcher for your package and transform, and include the appropriate MSI engine for Windows 9x or NT. You can also specify command-line arguments for the Windows Installer.

SMS Panel

In many cases, you may want to perform distribution of the transform and Windows Installer package using Microsoft SMS. Tuner provides a way to create both a Package Definition File (PDF) or an SMS file in the SMS View. Select the file type(s) you want to create. If you create an SMS file, Tuner can instruct SMS to create a Management Information Format (MIF) file when SMS deploys the package and transform. If you want to do this, provide the Install MIF Filename, Uninstall MIF Filename, and serial number.

Packaging Summary Panel

The Package Summary panel informs you of the packaging options selected in the three previous panels. If you need to make changes, use the Back button to return to the previous panel. The Cancel button aborts the packaging operation. If you are satisfied with the selected options, click Finish to copy the installation files to the specified location and/or create Setup.exe and SMS files.

16

Using Analyze to Perform Package Testing

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

Application Manager's Analyze view is a unified testing, reporting, and issue management interface that simplifies and streamlines all phases of application compatibility testing. Using Analyze, you can execute a broad range of compatibility, validation and conflict tests; manage and re-mediate issues; and monitor overall status in a single location.

Information about using Analyze is organized into the following sections:

Table 16-1 • Using the Analyze Tab

Section	Description
Analyze Overview	Provides an overview of the tasks you can perform using Analyze, the benefits of using Analyze, the groups of tests that are available to run, and explains how functionality found in previous releases of AdminStudio is now performed using Analyze.
Configuring Testing Options on the Analyze Tab	Explains how to select the tests to execute, and how to set resolution options.
Performing Compatibility, Best Practices, and Risk Assessment Testing	Describes how to perform testing for operating system compatibility, best practices, risk assessment, and application virtualization compatibility.
Performing MSIX Conversion Compatibility Testing	Describes how to perform MSIX conversion compatibility on msi packages in the Application Catalog.
Performing Application Conflict Testing	Describes how to perform conflict testing between source packages and target packages/operating systems.
Integrating Analyze With Other Applications	Explains how to connect your Application Catalog with your Microsoft ACT database, enabling you to view ACT test results in Application Manager.

Table 16-1 • Using the Analyze Tab

Section	Description
Viewing and Filtering Test Results	Describes how to view Analyze test results, and how to suppress errors/warnings generated by specific tests.
Resolving Issues	Describes how to perform automatic resolution of errors/warnings that Analyze detected. Also provides guidelines for performing manual resolution.
Viewing Test Summary Reports on Reports Tab	Describes the reports that are available on the Reports tab.
Analyze Reference	Describes the views, wizards, and dialog boxes used when performing package testing using Application Catalog Analyze tab.

Analyze Overview

Edition • *Application Manager, including Analyze, is included with AdminStudio Professional and Enterprise Editions.*

Application Manager's Analyze view is a unified testing, reporting, and issue management interface that simplifies and streamlines all phases of application compatibility testing. Using Analyze, you can execute a broad range of compatibility, validation and conflict tests; manage and remediate issues; and monitor overall status in a single location.

Tasks You Perform Using Analyze

The main tasks that you perform using Analyze involve clicking one of the following buttons on the Analyze tab ribbon:

Table 16-2 • Tasks You Perform Using Analyze

Button		Description
	Select Tests to Execute	Click to open the **Select Tests to Execute** dialog box, where you can select the tests that you want to execute when the **Execute Tests** button is clicked. On this dialog box, you can also set automatic fix preferences for the Operating System Compatibility test groups. See Configuring Testing Options on the Analyze Tab.

Table 16-2 • Tasks You Perform Using Analyze

Button		Description
	Execute Tests	Click to execute the following groups of tests against the selected Windows Installer, App-V, Apple iOS, or Google Android package or against all of the packages in the selected application or group: • Operating System Compatibility • MSIX Conversion Compatibility • Application Virtualization Compatibility • Best Practices • Risk Assessment • Remote Application Publishing Compatibility See Performing Compatibility, Best Practices, and Risk Assessment Testing.
	Launch Conflict Wizard	Click to open the Conflict Wizard, which you can use to run the Application Conflicts group of tests to determine conflicts between the source and target packages. See Performing Application Conflict Testing.
	Resolve Issues	Click to apply automatic fixes to those errors and warnings for which fixes are available. See Resolving Issues.

Viewing Test Results in Analyze

Analyze offers both summary and detailed test result views. Summary views display icons to quickly indicate the overall tests status of the package, application, or group of applications per test category:

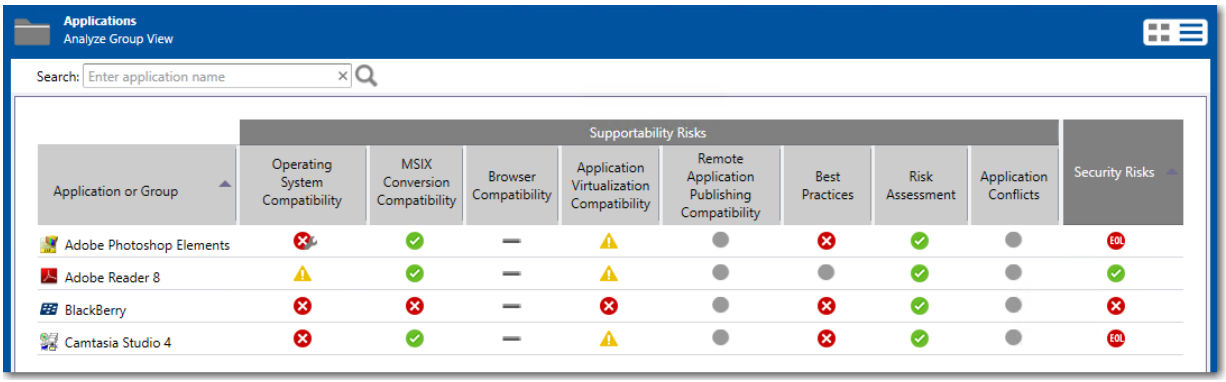


Figure 16-1: Analyze Group View

More Information

For more information about Analyze, see the following topics:

- Benefits of Using Analyze
- Test Run Optimization
- About Mobile Application Testing
- About Microsoft Windows Application Compatibility Infrastructure Testing

Benefits of Using Analyze

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

Using Analyze, you can execute a broad range of compatibility, validation and conflict tests; manage and remediate issues; and monitor overall status in a single location.

Using Analyze provides the following benefits:

- **Full suite of tests**—When using Analyze, with a single click you can simultaneously test groups of Windows Installer, App-V, Apple iOS, and Google Android packages for operating system compatibility, MSIX Conversion Compatibility, best practices, risk assessment, and application virtualization compatibility. You can also quickly perform conflict analysis between source and target packages/operating systems using a streamlined Conflict Wizard.
- **Efficient issue resolution**—In Analyze, you can manage the resolution of issues efficiently. You can filter results by test group. With one click, you can automatically resolve a single set of issues, groups of issues, or all issues. You can also suppress issues that the packaging manager feels should not be resolved.
- **Test run optimization**—When you initiate testing, Analyze checks to see which of the selected tests have already been run on the selected applications. If an application's packages or associated transform files have not changed, Analyze will run only those tests which have not yet been run. This enables you to halt testing at any time, and restart it later without unnecessarily repeating the run of any of the selected tests. For more information, see Test Run Optimization.
- **Integrated into the Application Catalog**—Analyze is a fully integrated component of the AdminStudio Application Manager and Application Catalog, which serves as the central repository for applications in all formats. Using the catalog structure, you can execute tests at a package level, an application level, a group level or even across an entire catalog. You can also manage and resolve issues in the same manner, from the same interface, giving you a single place to manage multiple steps of the packaging process.
- **Integration with Microsoft ACT database**—You can integrate Application Manager Analyze with your Microsoft ACT (Application Compatibility Toolkit) database and display ACT test results. ACT is used to create an inventory of an organization's installed applications, computers, and devices, and enables you to collect compatibility data. For more information, see Integrating with Microsoft Application Compatibility Toolkit (ACT).

Test Run Optimization

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

To make testing more efficient, Application Manager has a test run optimization option that can speed up the testing of large groups of packages, and enable you to perform testing incrementally without rerunning tests unnecessarily.

If the **Optimize each test run** option on the **Analyze** tab of the Application Catalog **Options** dialog box is selected, when you click **Execute Tests**, Application Manager will only execute tests on a package that were not previously run. If this option is selected, before beginning testing, Application Manager:

- Checks each selected package to see which tests have been run on it and which have not been run.
- Checks each selected Windows Installer file (and its transform files) and App-V package to see if it has changed since the last time that testing was performed.

If the packages have not changed, Application Manager will then only execute those selected tests which have not yet been run.

If the **Optimize each test run** option is not selected, Application Manager will execute all selected tests on all selected packages each time testing is initiated, even if the test has already been run on a package and neither the package nor its transform file has changed.

If you have a large number of applications in your Application Catalog, selecting this option enables you to start testing, then click **Stop** to pause testing when you want to access Application Manager to perform other tasks. When you click **Stop**, Application Manager would finish executing the current test. When you were ready to resume testing, you could then click **Execute Tests** and Application Manager would immediately begin testing where it left off the last time testing was performed.

About Mobile Application Testing

You can import and test both local mobile app files and links to mobile apps in public stores. The supported import types along with the available testing categories is summarized in the following table.

Table 16-3 • Mobile App Testing by Type

Type	Platform	OS Compatibility Testing	Best Practices Testing	Risk Assessment Testing
Local File	**iOS (.ipa)**	✓	✓	✓
	Google Android (.apk)	✓		✓
	Windows Store (.appx)	✓		✓

Table 16-3 • Mobile App Testing by Type

Type	Platform	OS Compatibility Testing	Best Practices Testing	Risk Assessment Testing
Link to Mobile App in Public Store	**Apple App Store**	✓	✓ ***Important*** • *Only supported if mobile app file is available locally in directory identified on the* ***Plugin Options*** *tab of* ***Options*** *dialog box.*	✓
	Google Play Store	✓		✓
	Windows Store			✓

About Microsoft Windows Application Compatibility Infrastructure Testing

The Microsoft Windows Application Compatibility Infrastructure (Shim Infrastructure) is a technical solution provided by Microsoft to ensure compatibility of existing software with new releases of their operating systems.

As the Windows operating system evolves from version to version—changing to support new technology or incorporate bug fixes—changes may affect existing applications. It is often not possible to modify the application to address these operating system changes. To make sure that these applications will continue to work in the updated operating systems, Microsoft uses the Shim Infrastructure to provide fixes (such as a transform or custom action) for a particular application version that may encounter problems in the updated operating system.

When Microsoft identifies an installer/application/driver with an incompatibility with a specific operating system, Microsoft will either provide a “shim” to enable it to run (such as a transform or custom action) or blocks it from running.

Analyze includes tests to scan installers, applications, and drivers for known runtime compatibility issues with various operating systems that have been documented in the Microsoft Windows Application Compatibility Infrastructure. The following table lists the test numbers of these tests.

Table 16-4 • Microsoft Windows Application Compatibility Infrastructure Test Numbers

Operating System	Compatibility Issues with Installers	Compatibility Issues with Drivers	Compatibility Issues with Applications
Windows 8 (32-bit)	3058	3059	3060
Windows 8 (64-bit)	3158	3159	3160

Table 16-4 • Microsoft Windows Application Compatibility Infrastructure Test Numbers

Operating System	Compatibility Issues with Installers	Compatibility Issues with Drivers	Compatibility Issues with Applications
Windows 10 1809 (and 2019 LTSC) (32-bit)	2858	2859	2860
Windows 10 1909 (32-bit)	5258	5259	5260
Windows 10 2004 (32-bit)	5458	5459	5460
Windows 10 20H2 (32-bit)	5658	5659	5660
Windows 10 21H1 (32-bit)	5858	5859	5860
Windows 10 21H2 (32-bit)	7058	7059	7060
Windows 10 1809 (and 2019 LTSC) (64-bit)	2958	2959	2960
Windows 10 1909 (64-bit)	5358	5359	5360
Windows 10 2004 (64-bit)	5558	5559	5560
Windows 10 20H2 (64-bit)	5758	5759	5760
Windows 10 21H1 (64-bit)	5958	5959	5960
Windows 10 21H2 (64-bit)	7158	7159	7160
Windows 11 21H2 (64-bit)	6758	6759	6760
Windows Server 2012	0558	0559	-
Windows Server 2016	0658	0659	-
Windows Server 2019	1058	1059	-

Analyze can identify these compatibility issues during testing and alert you to potential issues. If the installer/application/driver will not run in a particular operating system, an error will be generated. If a 'shim' exists to enable it to run at, perhaps, reduced functionality, a warning will be generated.

If a warning or error is generated by one of these tests, it is recommended that you find out if a newer version of the installer/application/driver is available.

Note *• For more information, see Understanding Shims in the Microsoft TechNet Library.*

Configuring Testing Options on the Analyze Tab

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

Before you begin testing, you need to specify which tests are going to be run, and specify auto-fix options for Operating System Compatibility, and MSIX Conversion Compatibility tests. Both of these tasks are performed on the **Select Tests to Execute** dialog box. This dialog box also provides detailed information on each test, including information on how to resolve issues that are found.

For information on configuring testing, see the following topics:

- About Analyze Tests
- Selecting Tests to Execute
- Setting the Compliance Level for Operating System Compatibility Tests
- Setting Automatic Fix Preferences for Operating System Compatibility Tests
- Updating the Location of the Custom ACE Rule File
- Changing the ICE Validation File
- Creating Custom Mobile Tests Using the Mobile Test Wizard

About Analyze Tests

Analyze offers the following groups of tests:

Table 16-5 • Analyze Test Group

Test Group	Description
Best Practices Tests	This category of tests checks the structure of Windows Installer packages, App-V packages, and mobile apps, and determines if they violate best-practice guidelines. This category includes the following areas: • Windows Installer internal consistency evaluators (ICEs) • Windows best practices • Microsoft App-V best practices For of a detailed description of all of the tests in this category, see Best Practices Tests.

Table 16-5 • Analyze Test Group

Test Group	Description

Risk Assessment Tests	This category of tests perform risk assessment checks for both mobile and desktop apps. This category includes the following areas:
	• Mobile risk assessment
	  • Android Mobile
	  • Apple Mobile
	  • Windows Mobile
	• Desktop risk assessment
	For of a detailed description of all of the tests in this category, see Risk Assessment Tests.

Application Conflicts Tests	This category of tests identifies conflicts between packages in the Application Catalog, as well as between packages and OS Snapshots. This category includes tests for Windows Installer packages and Microsoft App-V packages. For of a detailed description of all of the tests in this category, see Application Conflicts Tests.
Operating System Compatibility Tests	This category of tests checks packages for application readiness on the following operating systems: • Windows 8.1 (32 bit and 64 bit) • Windows 10 1809 (and 2019 LTSC)(32 bit and 64 bit) • Windows 10 1909 (32 bit and 64 bit) • Windows 10 2004 (32 bit and 64 bit) • Windows 10 20H2 (32 bit and 64 bit) • Windows 10 21H1 (32 bit and 64 bit) • Windows 10 21H2 (32 bit and 64 bit) • Windows 11 21H2 (64 bit) • Windows Server 2012 • Windows Server 2016 • Windows Server 2019 For of a detailed description of all of the tests in this category, see Operating System Compatibility Tests.
MSIX Conversion Compatibility	This category of tests analyzes Windows Installer packages to determine if they are suitable candidates for conversion to msix: • Desktop For of a detailed description of all of the tests in this category, see MSIX Conversion Compatibility Tests.

Table 16-5 • Analyze Test Group

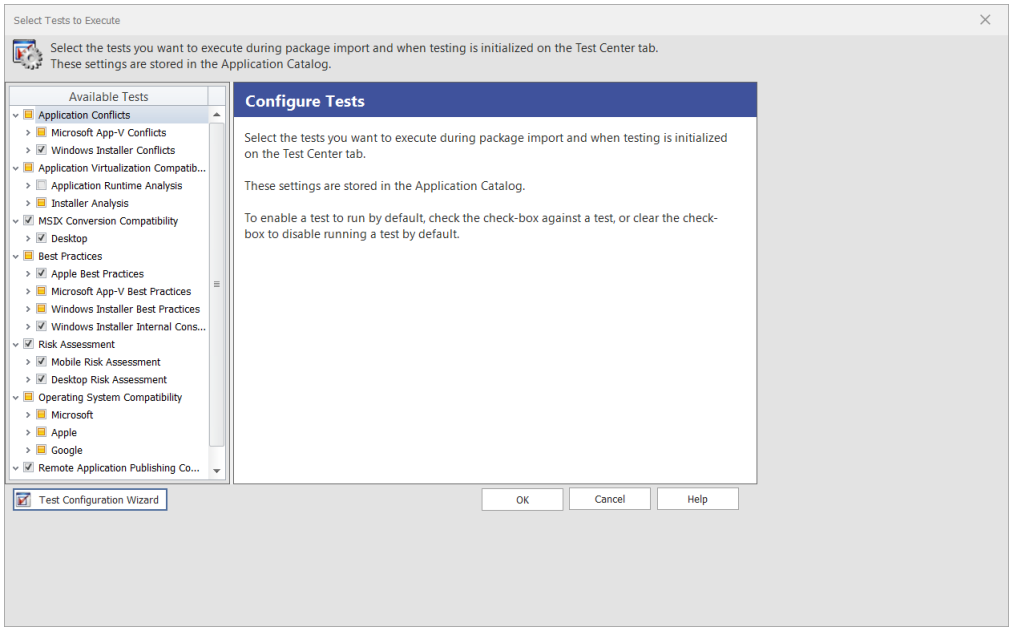
Test Group	Description
Application Virtualization Compatibility Tests	This category of tests analyzes Windows Installer packages to determine if they are suitable candidates for virtualization to the following formats: • Microsoft App-V • VMware ThinApp • Citrix XenApp For of a detailed description of all of the tests in this category, see Application Virtualization Compatibility Tests. ***Note*** *• The Application Virtualization Compatibility tests are always run any time that you run tests in Analyze. However, the selections you make on the* ***Select Tests to Execute*** *dialog box determine which virtual formats to display in test results.*
Remote Application Publishing Compatibility	This category of tests checks Windows Installer packages for compatibility to be run via Windows Remote Desktop. • Windows Remote Desktop Service tests For of a detailed description of all of the tests in this category, see Remote Application Publishing Compatibility Tests.

Selecting Tests to Execute

To select which tests are run each time **Execute Tests** or **Launch Web Test** is clicked, or when the Conflict Wizard is run, perform the following steps:

Task ***To select the tests to execute:***

1. Open the Application Manager **Analyze** tab.
2. Click **Select Tests to Execute** in the ribbon. The **Select Tests to Execute** dialog box opens.



3. Expand the tree listing and, in each of these groups, select the tests that you want to execute each time the **Execute Tests** button or the **Launch Web Test** button is clicked:

   - Operating System Compatibility Tests
   - MSIX Conversion Compatibility Tests
   - Application Virtualization Compatibility Tests
   - Best Practices Tests
   - Risk Assessment Tests
   - Application Conflicts Tests
   - Remote Application Publishing Compatibility Tests

   When you select a test in the tree, information about that test is displayed in the right pane. Reviewing this information may assist you in making your selections.

4. In the Application Conflicts Tests test group, select the tests that you want to run each time you perform conflict analysis using the Conflict Wizard.

5. Click **OK** close the dialog box.

Note • *To add custom ACE tests to the set of tests that are executed, see Updating the Location of the Custom ACE Rule File. You may want to do this if you have written some of your own custom ACE tests (as described in Creating Your Own Custom ACE Tests).*

Note • *To change the set of ICE tests that are run, see Changing the ICE Validation File.*

Setting the Compliance Level for Operating System Compatibility Tests

Instead of selecting individual Operating System Compatibility tests to run on the **Select Tests to Execute** dialog box, you have the option of using the Test Configuration Wizard to identify the tests to run by selecting one of three compliance levels, which are based on industry standard compliance rule sets:

- **Complete Analysis**—Test applications for all potential Operating System Compatibility issues.
- **Industry Standard Analysis**—Test for the Operating System Compatibility issues that would cause an application to fail.
- **Industry Standard Analysis With Auto-Fixes**—Only test applications for potential Operating System Compatibility issues for which an automatic fix is available.

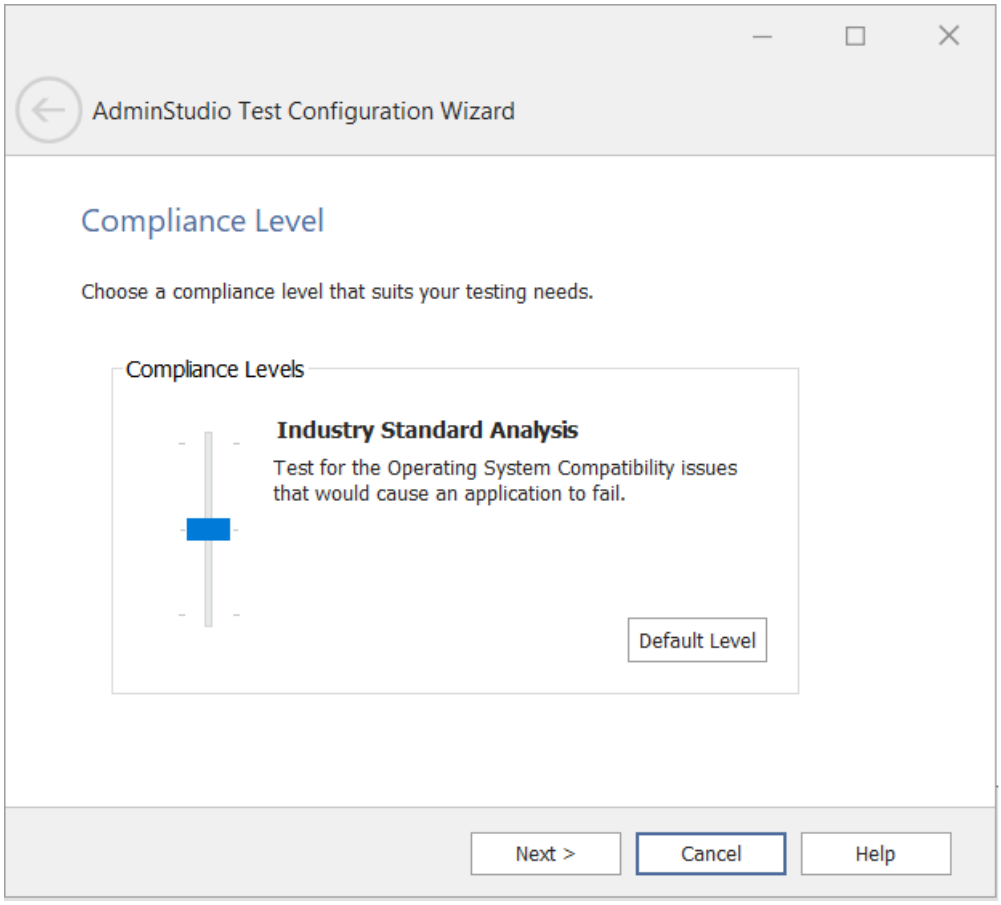
You can also further refine the tests that are run by specifying an OS Snapshot to test against. When you select an OS Snapshot to use to filter the test selection, the following items are considered:

- Operating system version
- Operating system patches applied
- Internet Explorer version installed
- .NET framework version installed

To set the compliance level for Operating System Compatibility tests using the **AdminStudio Test Configuration Wizard**, perform the following steps:

Task ***To use the AdminStudio Test Configuration Wizard:***

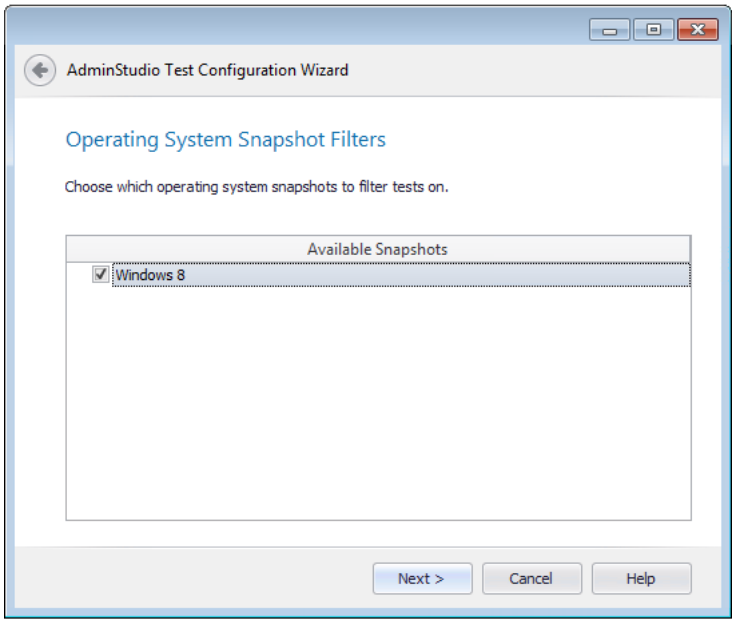
1. Open the Application Manager **Analyze** tab.
2. Click **Select Tests to Execute** in the ribbon. The **Select Tests to Execute** dialog box opens.
3. Click the **Test Configuration Wizard** button in the lower left. The **Compliance Level** panel of the **AdminStudio Test Configuration Wizard** opens.



4. Use the slider to select one of the following options:
   - **Complete Analysis**—Test applications for all potential Operating System Compatibility issues.
   - **Industry Standard Analysis**—Test for the Operating System Compatibility issues that would cause an application to fail.
   - **Industry Standard Analysis With Auto-Fixes**—Only test applications for potential Operating System Compatibility issues for which an automatic fix is available.

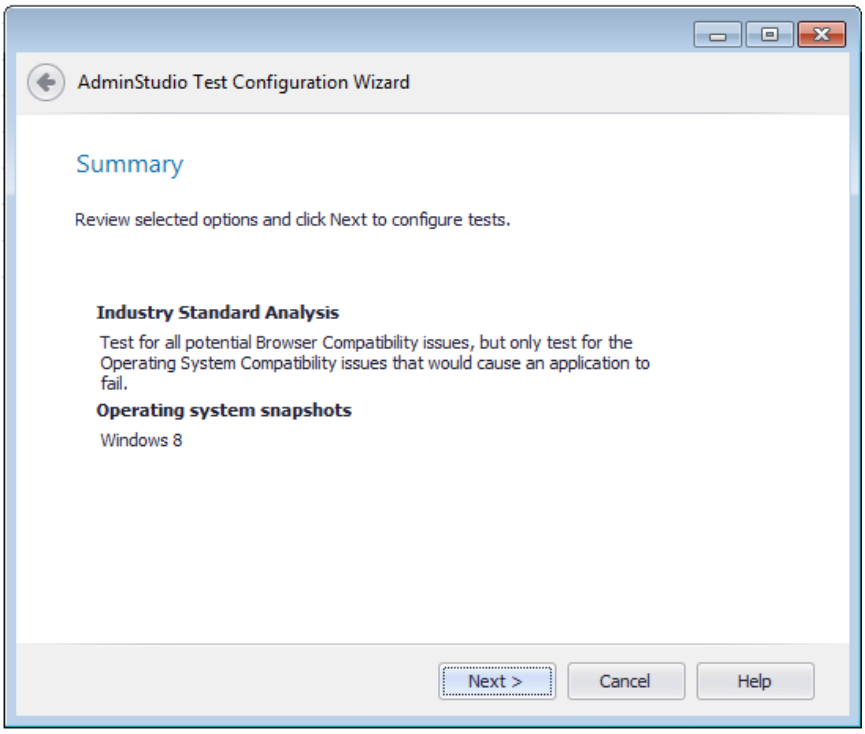
Important • *The **Compliance Level** selection you make on this panel does not affect the selection of tests in the Application Conflicts, Application Virtualization Compatibility, Best Practices, Risk Assessment, or Remote Application Publishing Compatibility test categories.*

5. Click **Next**. The **OS Snapshot(s)** panel opens, listing all OS Snapshots you have imported into your Application Catalog.



Note • *For more information, see Taking OS Snapshots and Importing OS Snapshots.*

6. If desired, select an OS Snapshot to test against. When you select an OS Snapshot to use to filter the test selection, the following items are considered:
   - Operating system version
   - Operating system patches applied
   - Internet Explorer version installed
   - .NET framework version installed
7. Click **Next**. The **Summary** panel opens, listing your selections.



8. Click **Next**. A message appears stating that the test configuration has been updated.
9. Click **Finish** to close the wizard.
10. Notice the following changes that have been made in the **Available Tests** list:

Test Category	Change
Operating System Compatibility	The Operating System Compatibility tests that are selected depend upon the level you chose on the **Choose a Compliance Level** panel. The selection of Operating System Compatibility tests will be further filtered if you selected an OS Snapshot on the **OS Snapshot(s)** panel. The only Operating System test categories that will have any selected tests will be the categories of the selected operating systems. For example, if you choose a Windows 8 64-bit OS Snapshot, tests will be selected only in the **Windows 8 64-bit** test category.
Application Conflicts **Application Virtualization Compatibility** **Best Practices** **Risk Assessment** **Remote Application Publishing Compatibility**	Test selection in these test categories are not affected by any selections made in the **AdminStudio Test Configuration Wizard**.

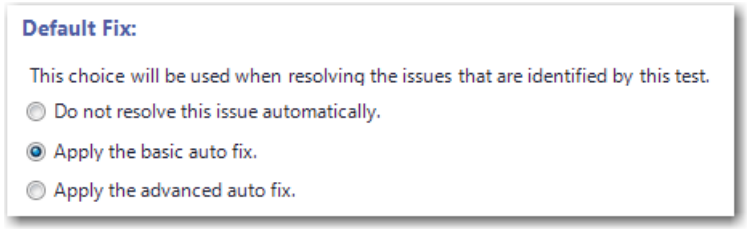
Setting Automatic Fix Preferences for Operating System Compatibility Tests

For some of the tests in the **Operating System Compatibility** test group, you have the option of specifying how you want Application Catalog to resolve automatically resolvable issues. You can instruct Application Catalog to perform the *basic* auto fix, the *advanced* auto fix, or not to fix issues generated by the test.

To set automatic fix preferences for operating system compatibility tests, perform the following steps.

Task ***To set automatic fix preferences for operating system compatibility tests:***

1. On the **Analyze** tab of Application Manager, click the **Select Tests to Execute** button in the ribbon. The **Select Tests to Execute** dialog box opens.
2. In the **Operating System Compatibility** test group, select the test that you want to modify. A description of that test appears in the right pane.
3. After reviewing the information, scroll down until you locate the **Default Fix** section:



Note • *For some tests, one of these options is disabled. For others, all options are disabled.*

4. Select one of the following options to specify the action Application Catalog should take when you click the **Resolve Issues** button in the ribbon:

Fix Type	Description
Do not resolve this issue automatically	Select this option if you do not want Application Catalog to automatically resolve any issues generated by this test.
Apply the basic auto fix	Select this option if you want Application Catalog to resolve issues generated by this test by applying the basic auto fix. Applying the **basic auto fix** is relatively safe. It results in minimal changes to an MSI package via a Windows Installer transform. It does not change the target system's security or a system policy.
Apply the advanced auto fix	Select this option if you want Application Catalog to resolve issues generated by this test by applying the advanced auto fix. Applying the **advanced auto fix** may result in a loss of functionality, and it may not resolve all types of issues. This type of fix may change the target system's security or a system policy. One example of an advanced auto fix is the removal of a registry key that is protected by Windows Resource Protection.

5. Click **OK**.

Note • *The test description pane often also includes information on how to perform a manual fix. For more information, see Performing Manual Issue Resolution.*

Updating the Location of the Custom ACE Rule File

You can optionally use user-defined custom ACEs to extend the functionality of existing tests with company-specific functionality. By selecting different user-defined ACE files, you can organize rules appropriate for individual users in your organization.

To specify that you want custom ACE tests to be executed when the **Execute Tests** button is clicked or when the Conflict Wizard is run, perform the following steps:

Task ***Updating the location of the Custom ACE rule file:***

1. Create custom ACE tests, as described in Creating Your Own Custom ACE Tests.

   By default, an empty user-defined ACE file is installed in the following location on the machine where AdminStudio is installed.

   AdminStudio Shared Directory\ConflictSolver\CustomConflictFile.ace

2. Open the **ACE Tests** tab of the Application Catalog **Options** dialog box.
3. In the **Custom ACE Rule File** field, select the custom ACE file containing the ACEs that you want to run.

   Note • *Only one user-defined ACE file can be active at one time.*

Changing the ICE Validation File

Validation involves comparing a Windows Installer package against a known criteria to identify deviations from those rules. By default, Application Catalog compares packages against the full Windows Installer validation suite. This suite contains a comprehensive set of Internal Consistency Evaluators (ICEs)—guidelines created by Microsoft to ensure an installation package works correctly with the Windows Installer engine.

The CUB files containing these ICE validation tests are specified on the **Virtualization and Windows Installer** tab of the Application Catalog **Options** dialog box. By default, the following CUB files are specified:

```
C:\Program Files (x86)\AdminStudio\2022 R2\Common\Support\darice.cub
C:\Program Files (x86)\AdminStudio\2022 R2\Common\Support\MergeMod.cub
```

In the overwhelming majority of cases, these are the files you will want Application Catalog to use. However, there may be times you want to compare your packages against a different validation file, depending on your needs.

To specify a different ICE validation file, perform the following steps.

Task ***To specify a different validation file to use:***

1. In Application Catalog, select **Options** on the **Application Catalog** menu. The **Options** dialog box opens.
2. Open the **Virtualization and Windows Installer** tab.
3. Next to the **Windows Installer CUB validation file** and **Merge Module CUB validation file** fields, click the browse button (...) and select the validation file (.cub) that you would like to use.

   The files specified in these fields contain the Internal Consistency Evaluators (ICEs) that will be used for validation of Windows Installer packages and merge modules.

Creating Custom Mobile Tests Using the Mobile Test Wizard

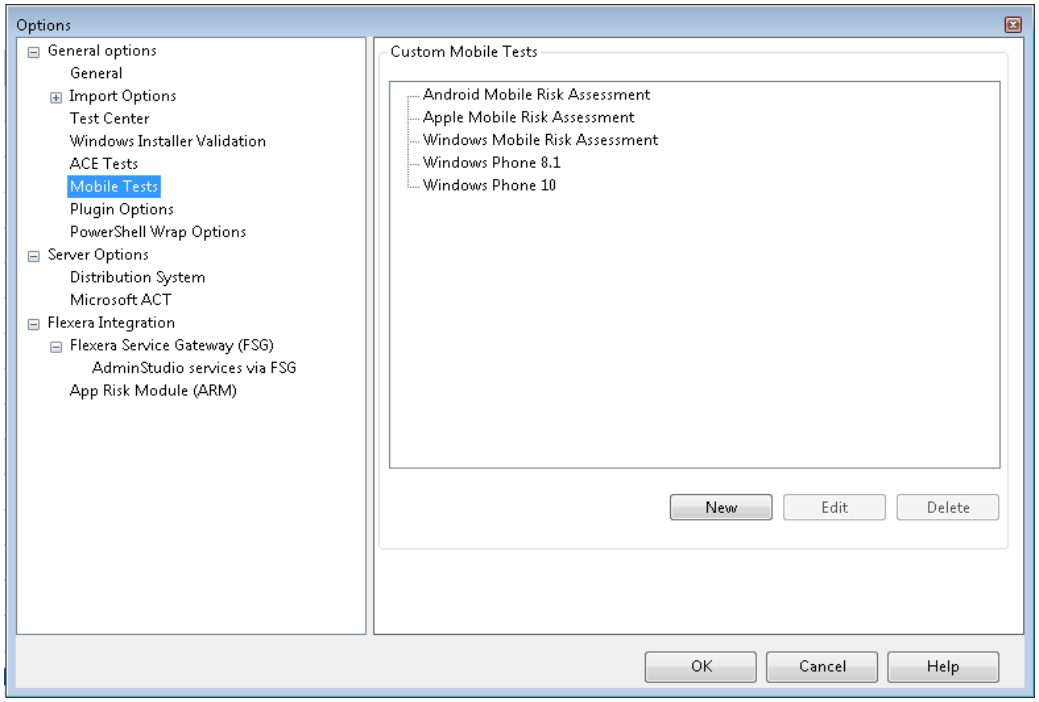
AdminStudio's mobile risk assessment tests enable you to find out which features a specific mobile app uses, such as telephone, location services, camera, microphone, etc. You can enhance this testing by using the **Mobile Test Wizard** to create custom tests that combine risk assessment checks with AND or OR operators.

For example, you could create a custom test to see if a mobile application uses a gyroscope OR accelerometer. Or you could create a test that determines whether a mobile application uses location services AND allows location tracking.

To create a custom test, perform the following steps.

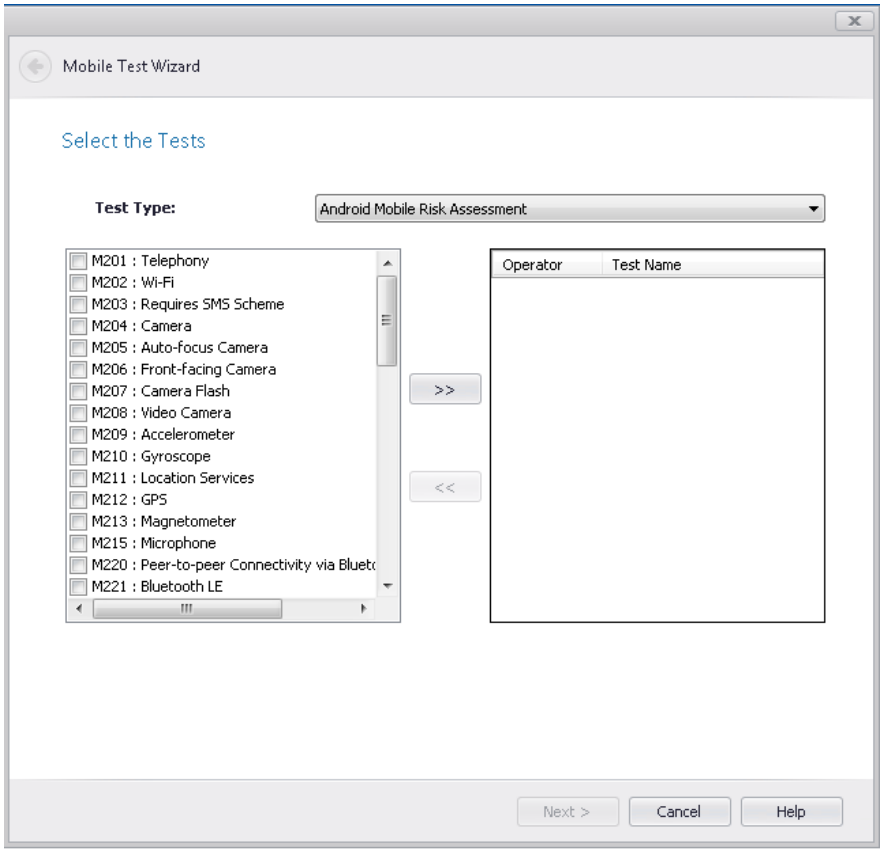
Task ***To create a custom mobile test:***

1. In Application Catalog, select **Options** on the **Application Catalog** menu. The **Options** dialog box opens.
2. Open the **General Options > Mobile Tests** tab.

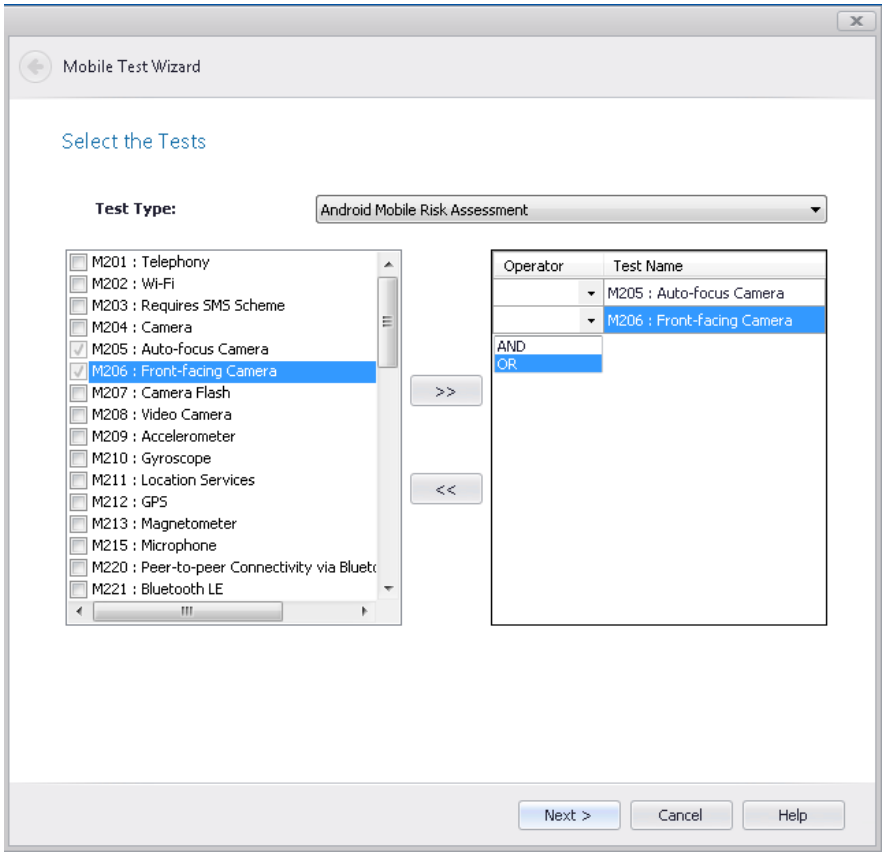


Existing mobile tests, if any, are listed in the pane on the right, under one of the listed categories.

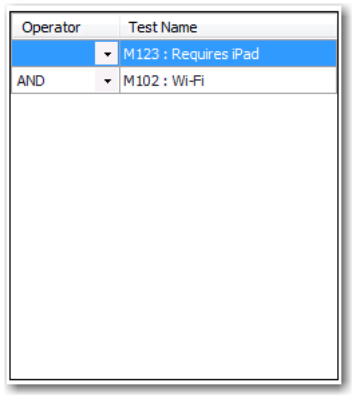
3. Click **New**. The **Select the Tests** panel opens.
4. From the **Test Type** list, select a category. The available tests in that category are listed in the box on the left.



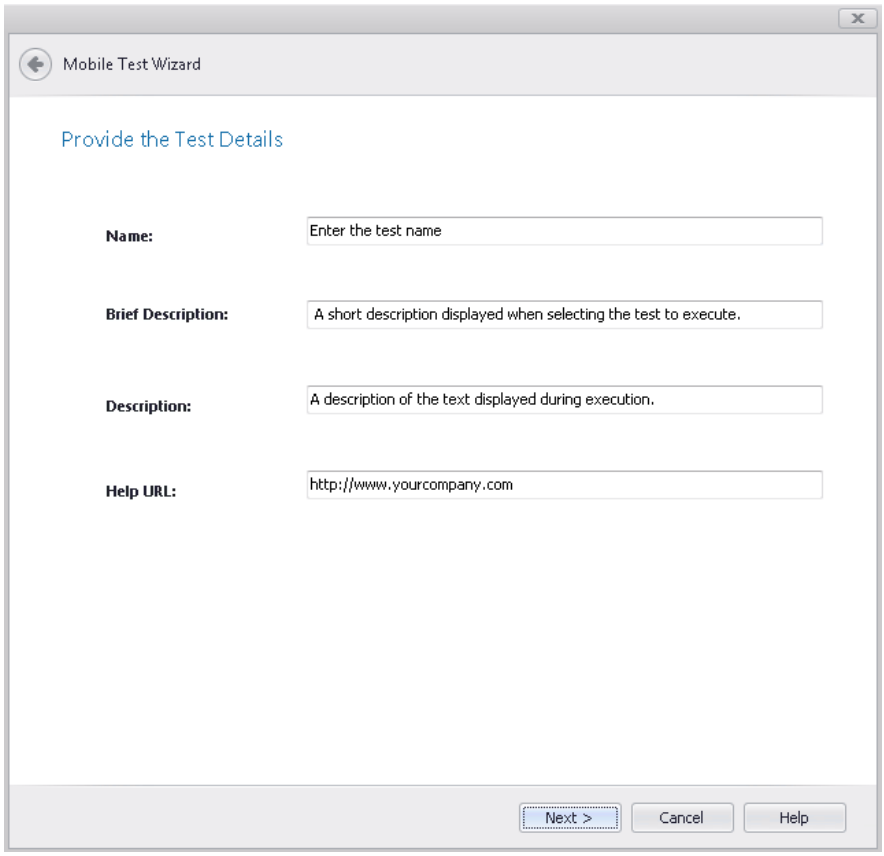
5. Use the arrow buttons to move the tests you want to include in the custom test to the list on the right. As you add the tests, join them using AND or OR operators by making selections from the **Operator** drop down list.



For example, the following test would test a iOS mobile app to see if it requires iPad and WiFi.



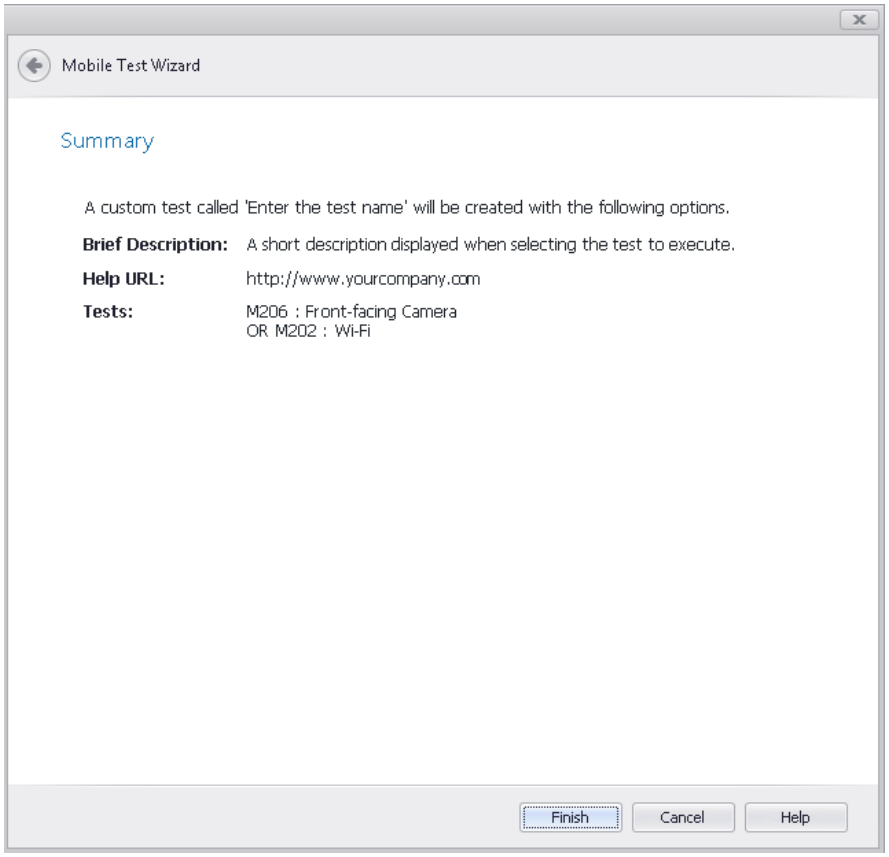
6. After you have selected the tests and joined them with the AND or OR operator, you click **Next**. The **Provide the Test Details** panel opens.



7. Enter the following information:

Property	Description
Name	Enter a name to identify this custom test. This name will be displayed on the **Mobile Tests** panel of the **Options** dialog box. This name will also be displayed on the **Select Tests to Execute** dialog box, and in test results on the Analyze tabs.
Brief Description	Enter a short description of the purpose of this test. This text will be displayed under **Brief Description** on the **Select Tests to Execute** dialog box when this test is selected in the tree.
Description	Enter a thorough description of how this test works and why it was created. This text will be displayed under **Background** on the **Select Tests to Execute** dialog box when this test is selected in the tree.
More Information	Enter a link to a web page that provides additional information on this custom mobile test. This hypertext link will be listed under **More Information** on the **Select Tests to Execute** dialog box when this test is selected in the tree.

8. Click **Next**. The **Summary** panel opens.



9. Click **Finish**. The custom mobile test is created.

Viewing Custom Mobile Tests

After you create a custom mobile test, it is listed on the **Mobile Tests** tab of the **Options** dialog box

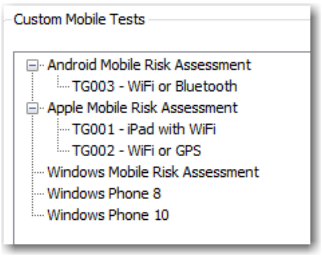


Figure 16-2: Custom Mobile Test on the Mobile Tests Tab of Options Dialog Box

The custom mobile test is also listed in the tree on the **Select Tests to Execute** dialog box at the bottom of the list of tests in that category:

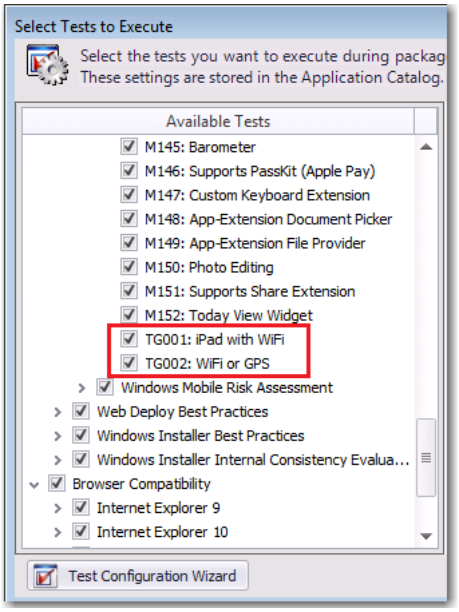


Figure 16-3: Select Tests to Execute Dialog Box

The information that was entered in the **Brief Description**, **Description**, and **Help URL** fields of the **Provide the Test Details** panel of the Mobile Test Wizard is displayed in the pane on the right of the **Select Tests to Execute** dialog box when the custom test is selected in the tree.

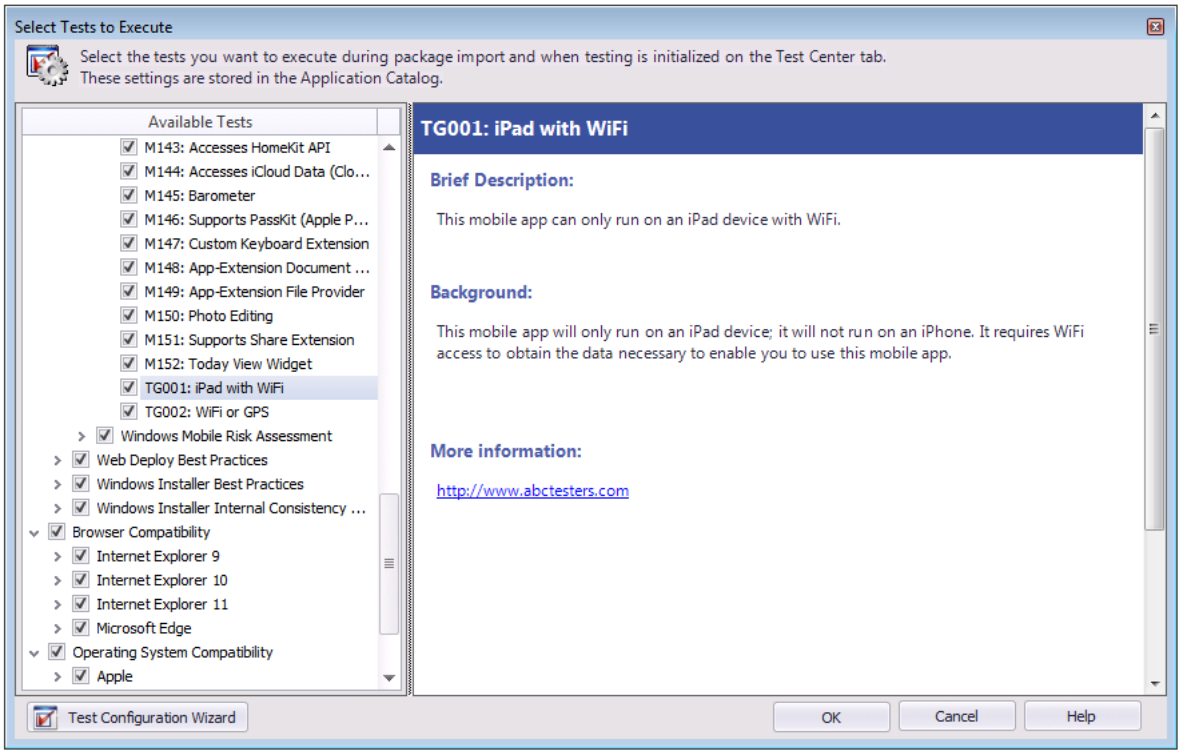


Figure 16-4: Select Tests to Execute Dialog Box

When this issue is detected during the testing of a mobile app, the error message is listed on the **Best Practices** tab in Analyze.

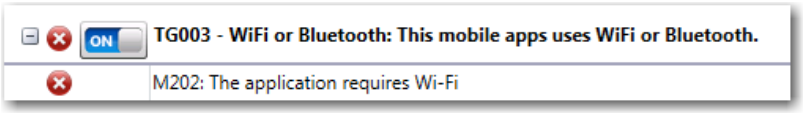


Figure 16-5: Custom Test Displayed on Best Practices Tab of Analyze

Performing Compatibility, Best Practices, and Risk Assessment Testing

On the Application Catalog **Analyze** tab, you perform compatibility, best practices, and risk assessment testing of selected packages simultaneously using the **Execute Tests** button in the ribbon.

You can test a group of packages (and all of its subgroups), a single application, or a single package. Tests are run on packages of all deployment types simultaneously, including Windows Installer packages, Apple iOS and Google Android mobile apps, and App-V packages.

When you click the **Execute Tests** button, Application Catalog runs all of the selected tests in the following test categories:

- Application Virtualization Compatibility Tests
- Best Practices Tests
- Risk Assessment Tests
- Operating System Compatibility Tests
- Remote Application Publishing Compatibility Tests

Note • *The **Application Virtualization Compatibility** tests are always run any time that you run tests in Analyze. However, you can choose which virtual formats to display in test results. For more information, see Choosing the Virtual Formats to Display in Test Results.*

Note • *By default, all of the selected tests on packages are run immediately after import. To disable this feature, you need to clear the selection of the **Automatically Execute Tests After Import** option on the **Import Options** tab of the Application Catalog **Options** dialog box.*

To test a package or group of packages for compatibility, best practices, and risk assessment, perform the following steps.

Task ***To perform compatibility, best practices, and risk assessment testing:***

1. Configure the tests that you want to run by performing the steps in Configuring Testing Options on the Analyze Tab.
2. Open the **Analyze** tab of Application Manager.
3. Specify the packages you want to test by selecting a group, application, or package in the tree.

Note • *If you select the root group (which is, by default, named **Applications**), all of the Windows Installer packages, App-V packages, Apple iOS mobile apps, and Google Android mobile apps in the Application Catalog will be tested.*

Note • *The Application Virtualization Compatibility Tests are not executed on App-V packages, Apple iOS mobile apps, or Google Android mobile apps even if they are selected; these tests are not applicable to App-V packages or mobile apps.*

4. Click **Execute Tests** in the ribbon. Testing is initiated and messages are displayed in the Output window. When testing is finished, `Completed testing package(s)` is displayed in the Output window.
5. View the test results, as described in:
   - Viewing Operating System Compatibility Test Results
   - Viewing Application Virtualization Compatibility Test Results
   - Viewing Best Practices Test Results
   - Viewing Risk Assessment Test Results
6. Resolve any issues that were detected, as described in Resolving Issues.

Performing MSIX Conversion Compatibility Testing

You can test the Windows Installer (.msi) packages in your Application Catalog for compatibility for conversion to MSIX package format.

When a Windows Installer package is tested during import or by clicking the **Execute Tests** button on the **Analyze** tab, the MSIX conversion compatibility tests are run. The results of these tests are listed in the **MSIX Conversion Compatibility** column under **Supportability Risks** on the Analyze Group View and Analyze Application View.

To test the Windows Installer packages in your Application Catalog for compatibility for conversion to MSIX package format, perform the following steps.

Testing Windows Installer Packages for MSIX Conversion Compatibility

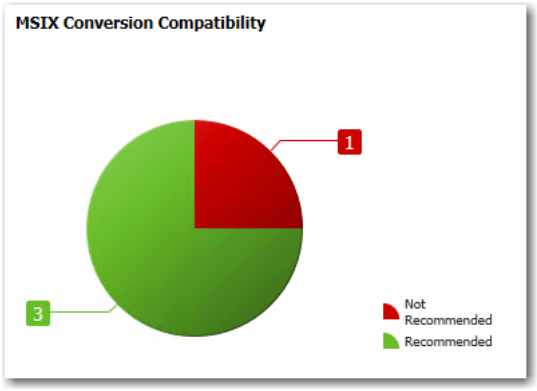
To test Windows installer packages for MSIX conversion compatibility, perform the following steps.

Task ***To test Windows Installer packages for MSIX conversion compatibility:***

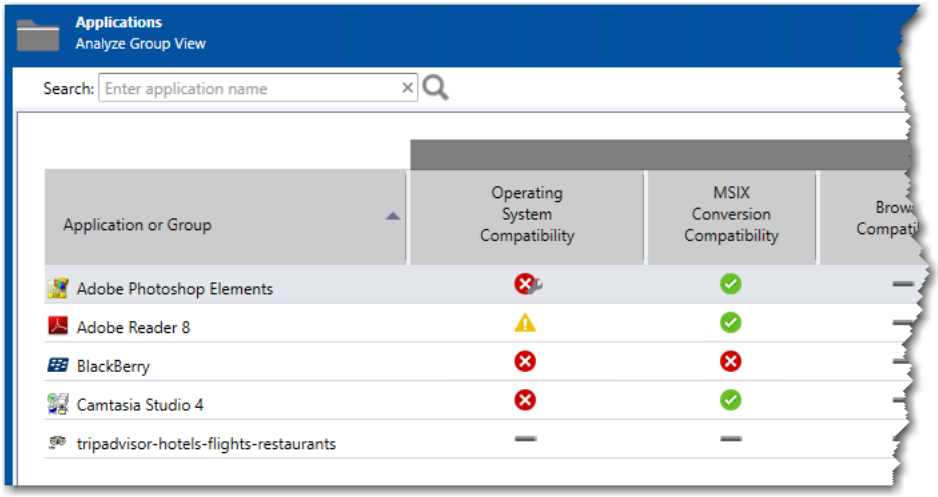
1. Open the Application Manager **Analyze** tab.
2. Make sure that the **MSIX Conversion Compatibility** tests are selected by performing the steps in Selecting Tests to Execute.
3. In the Application Manager tree, select the group or application containing the Windows Installer packages you want to test, and then click **Execute Tests**.

4. Testing is initiated and messages are displayed in the Output window. When testing is finished, `Completed testing package(s)` is displayed in the Output window.

5. Open the chart format of the **Analyze Group View** and view the **MSIX Conversion Compatibility** pie chart to see a summary of all packages that were tested for MSIX conversion compatibility.



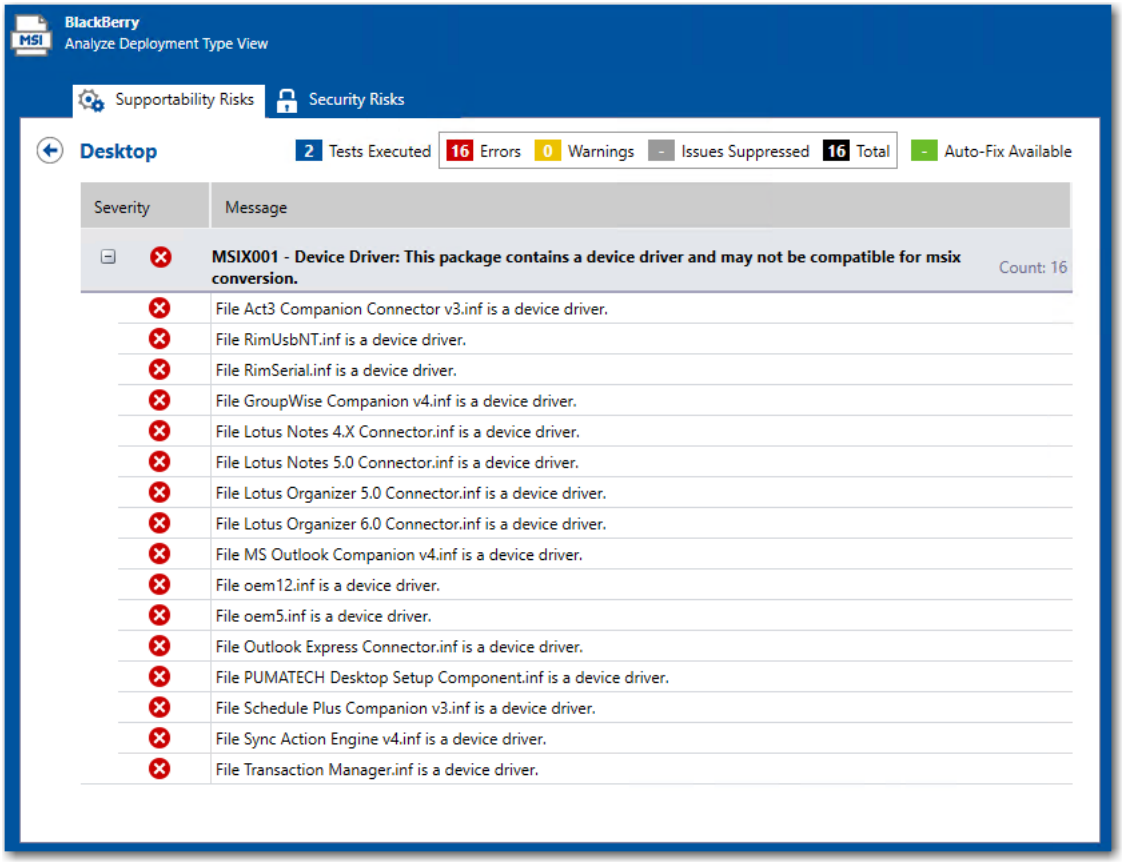
6. Open the list format of the **Analyze Group View** or the **Analyze Application View**. Status icons are listed in the **MSIX Conversion Conmpatibility** column under **Supportability Risks.**



Each package is identified as with one of the following icons:

Test Result	Icon	Description
Recommended		Recommended for conversion to MSIX format.
Not Recommended		Not recommended for conversion to MSIX format.

7. To view the detailed errors and warnings for those packages with the status of **Not Recommended**, click on the **Not Recommended** icon until you drill down to the **Supportability Risks Detail** view, where you can expand the test name to view all generated errors:



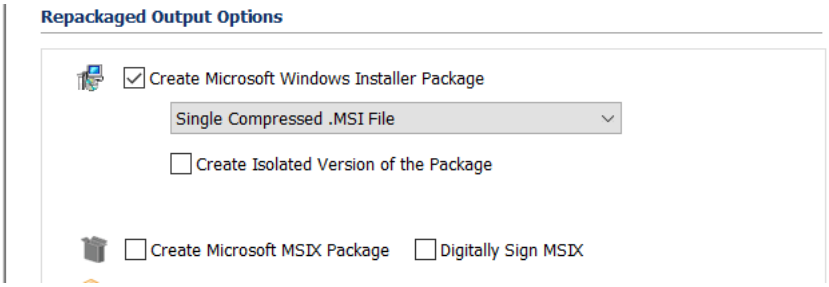
Note • *The results of MSIX conversion compatibility tests are also listed in the **MSIX Conversion Compatibility** report on the **Reports** tab. This report identifies each application in the Application Catalog as either **Recommended** for conversion to the MSIX package format or **Not Recommended**.*

Converting your Existing Packages to MSIX using Repackager Tool and Digital Signing

To convert your existing packages to MSIX using the Repackager tool and digital signing, perform the following steps.

Task ***To convert existing packages to MSIX using Repackager and digital signing:***

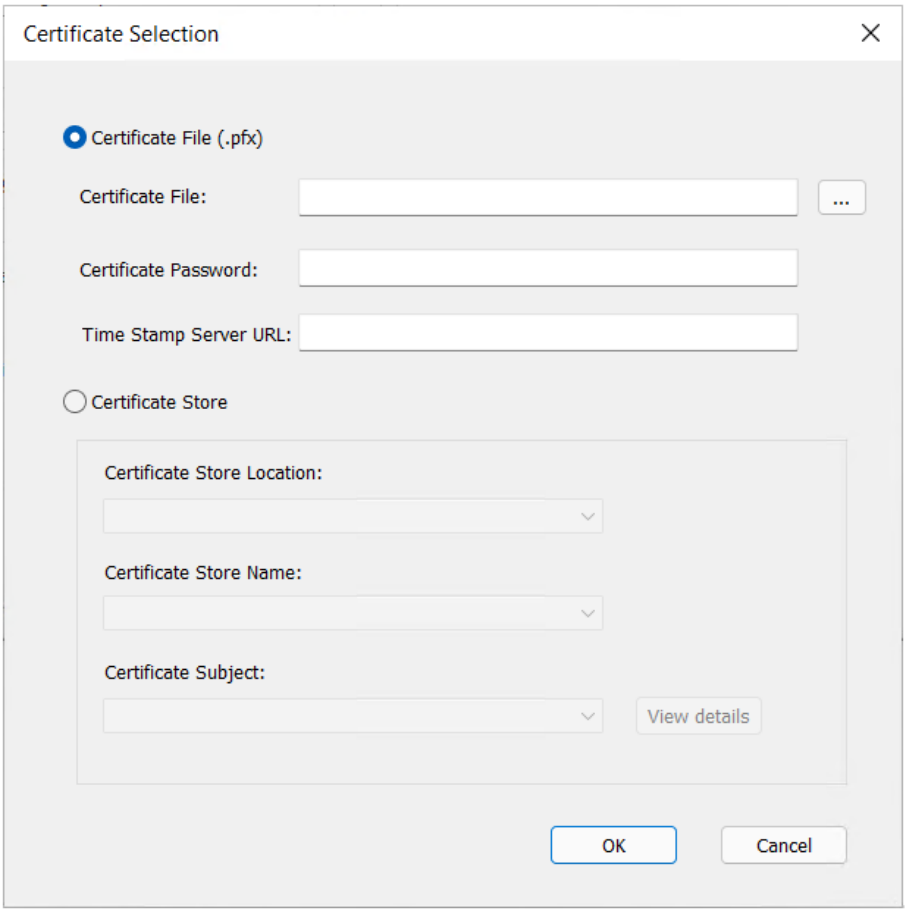
1. Launch Repackager.
2. Repackage an `.msi` or `.exe` to convert to `.msix` format.
3. On the **Repackaged Output Options** dialog box, select a package.



4. You can select any of the following options to convert the package:

   - **Create Microsoft MSIX Package**—You can convert the selected package to MSIX package but will not be signed digitally.
   - **Digitally Sign MSIX**—You will need a valid Pfx file certified by CA, Pfx file password, Certificate Store Location, Store Name, and Store Subject where to find the certificate.

   If you select the **Digitally Sign MSIX** option, the **Certificate Selection** dialog box is displayed.



5. To convert to digital signing, select either the **Certificate File (.Pfx)** or **Certificate Store** option.
6. If you select **Certificate (.pfx)**, enter **Certificate File** path or browse and select the certificate path, enter the password, and then enter the valid **Time Stamp Server URL**.
7. If you select **Certificate Store**, then provide the certificate store details as mentioned in the below steps:

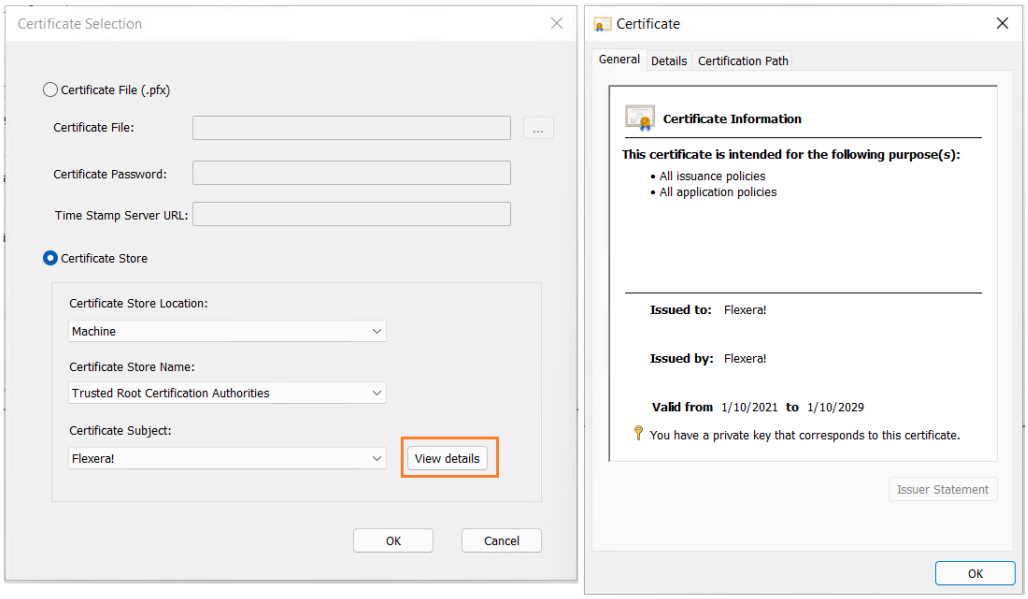
Note • *If you select the Certificate Store option, make sure that the Certificate has been imported. For more details on importing the certificate, see Import Certificate.*

a. From the **Certificate Store Location** drop down, select either **User** or **Machine**.

b. From the **Certificate Store Name** drop down, select desired store name from the list.

c. From the **Certificate Subject** drop down, select certificate subject from the list.

d. After selecting the valid details, **View details** button enables. Click the button and view the certificate details.



8. Click **OK**.

9. Click **Build**. The package is created in the `MSI_Package` folder.

Note • *Repackager now support Windows Services for MSIX Packages. A Windows Service installed while converting a legacy package format (MSI/EXE) will be captured and packaged into the MSIX package upon conversion.*

Signing the MSIX Package in the Application Catalog

To sign a selected MSIX app package using the **Sign Package** option, you can either use the `certificate` file or the certificate store.

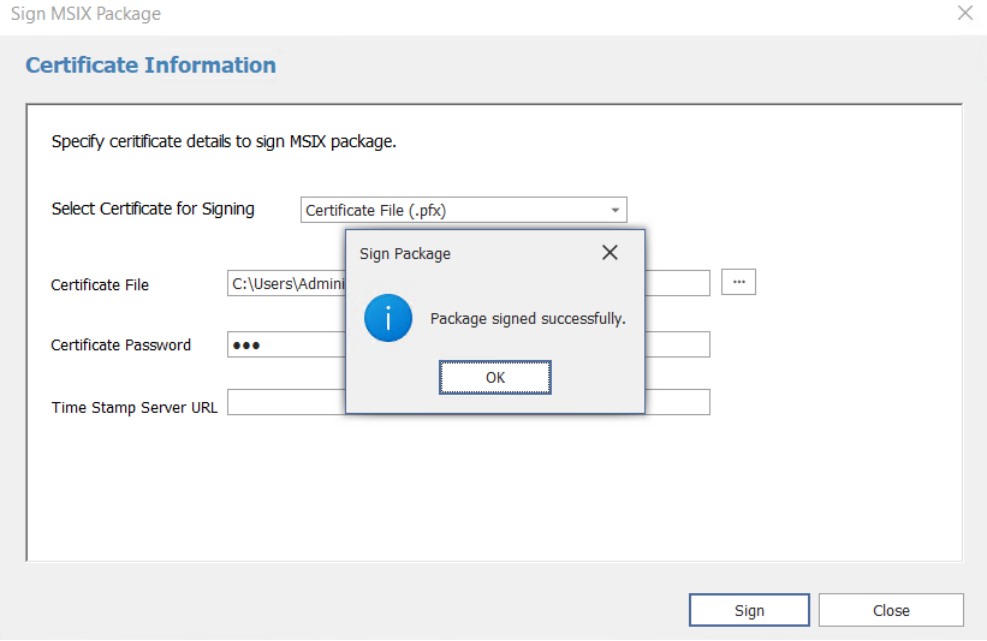
- Signing an MSIX App Package Using a Certificate File (.pfx)
- Signing an MSIX App Package Using the Certificate Store

Signing an MSIX App Package Using a Certificate File (.pfx)

An MSIX package can be signed using a certificate using the Certificate File (.pfx file).

Task ***To sign an MSIX app package using a Certificate File (.pfx):***

1. Right-click the MSIX package and select **Sign Package**.
2. The **Certificate Information** panel appears, select **Certificate File (.PFX)** from the **Select Certificate for Signing** list.
3. Enter the **Certificate File** path or browse and select the certificate path and then enter the **Password**.
4. Enter the valid **Time Stamp Server URL**.
5. Click **Sign**.
6. Click **Ok** and agree the confirmation dialog which appears.



Signing an MSIX App Package Using the Certificate Store

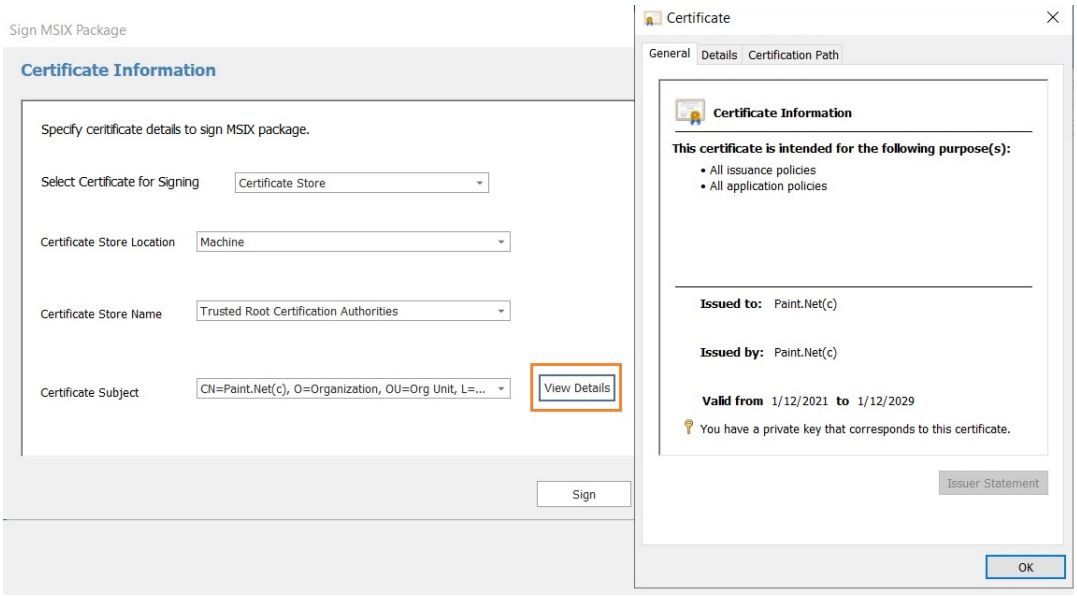
An MSIX package can be signed using the Certificate Store.

Note • *If you select the Certificate Store option, make sure that the Certificate has been imported. For more details on importing the certificate, see Import Certificate.*

Task ***To sign an MSIX app package using the Certificate Store:***

1. Right-click the MSIX package and select **Sign Package**.
2. The **Certificate Information** panel appears, select **Certificate Store** from the Select **Certificate for Signing** list.
3. From the **Certificate Store Location** drop down, select either **User** or **Machine**.
4. From the **Certificate Store Name** drop down, select store name from the list.

5. From the **Certificate Subject** drop down, select certificate subject from the list.
6. After selecting the valid details, **View details** button enables. Click the button and view the certificate details.



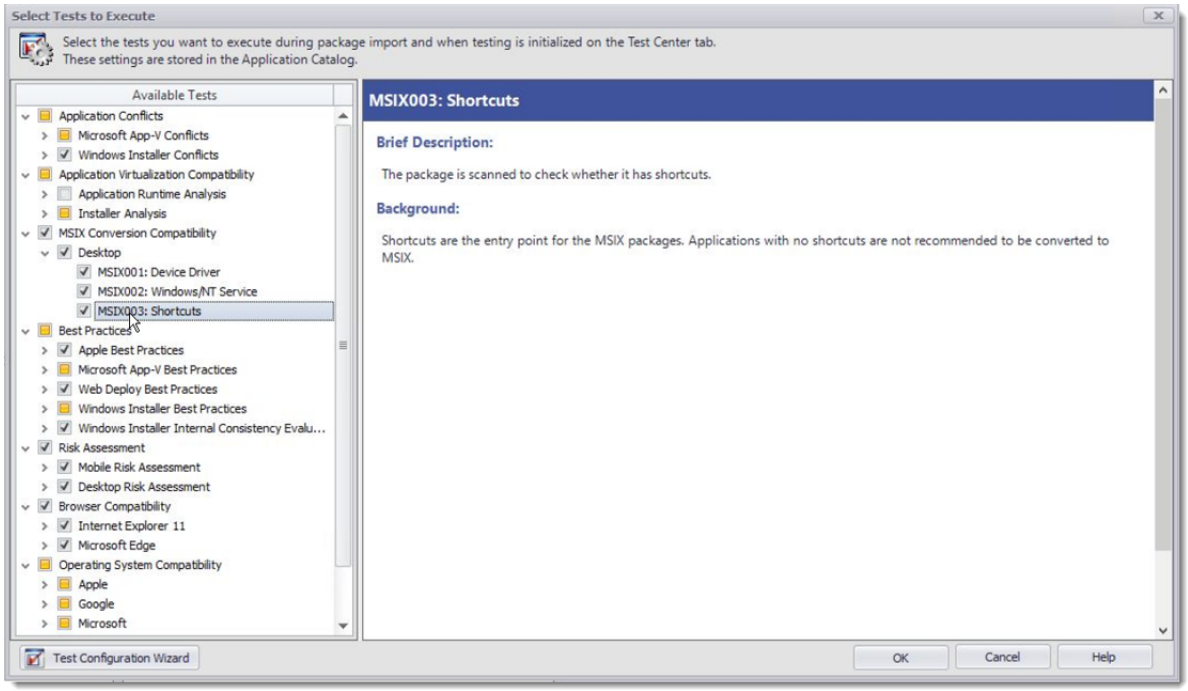
7. Click **Sign**.

Scanning and Checking Whether or Not the Package Has Shortcuts for MSIX Conversion

To scan and check whether or not a package has shortcuts for the MSIX conversion, perform the following steps.

Task ***To scan and check whether a package has shortcuts for MSIX conversion:***

1. Open the **Analyze** tab of Application Manager.
2. Click **Select Tests to Execute**. The **Select Tests to Execute** dialog box appears.



3. Under **MSIX Conversion Compatibility**, expand the **Desktop** tree to view the tests.
4. Select **MSIX003: Shortcuts** check box.
5. Click **OK**.
6. Click E**xecute Tests** in the ribbon.

MSIX App Attach

MSIX app attach is the best way to deliver MSIX applications to both physical and virtual machines. However, MSIX app attach is different from regular MSIX because it's made especially for Windows Virtual Desktop. Since AdminStudio supports importing MSIX package into catalog, it would be helpful for the customer if AdminStudio provides a way to creating App Attach for the imported MSIX packages.

To create MSIX App Attach, perform the following tasks:

- About MSIX App Attach
- Prerequisites for Creating App Attach
- Creating MSIX App Attach

About MSIX App Attach

MSIX app attach is an application layering solution that allows you to dynamically attach apps from an MSIX package to a user session. The MSIX package system separates apps from the operating system, making it easier to build images for virtual machines. MSIX packages also give you greater control over which apps your users can access in their virtual machines. You can even separate apps from the master image and give them to users later.

Prerequisites for Creating App Attach

The prerequisite includes the following:

- To create App Attach, Hyper-V should be enabled.
- MSIX should be signed.

Important • *The Hyper-V service can't be enabled on Windows 32-bit machines, so that you cannot create App Attach if the AdminStudio instance is running on a 32-bit machines.*

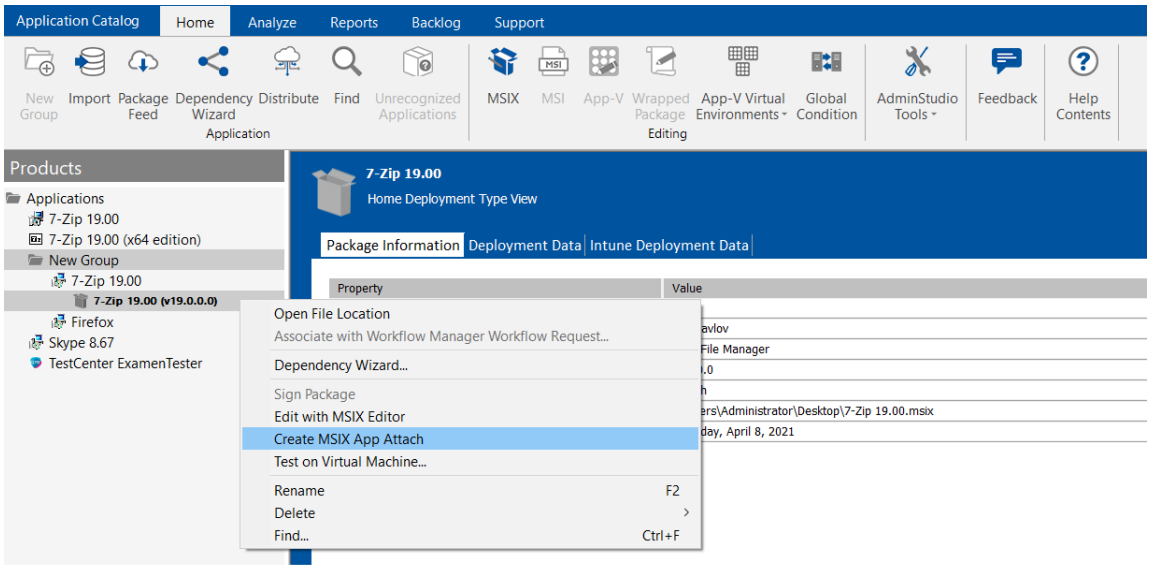
Creating MSIX App Attach

To create MSIX App Attach, perform the following steps:

Task ***To create MSIX App Attach:***

1. In the **Home** tab of Application Manager, select an MSIX package in the Applications tree in the left pane, and then click **Create MSIX App Attach** from the right click menu.

   The **Create MSIX App Attach Wizard** appears.

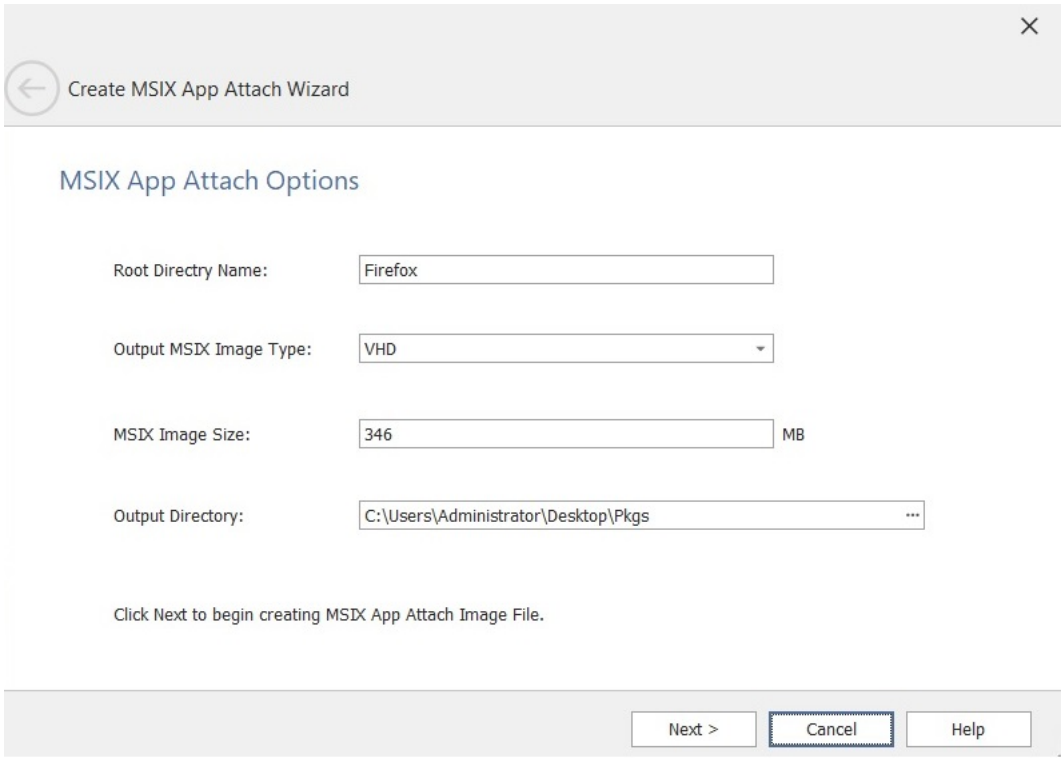


Note • *The **Create MSIX App Attach** option is available only for the MSIX Packages at the package level.*

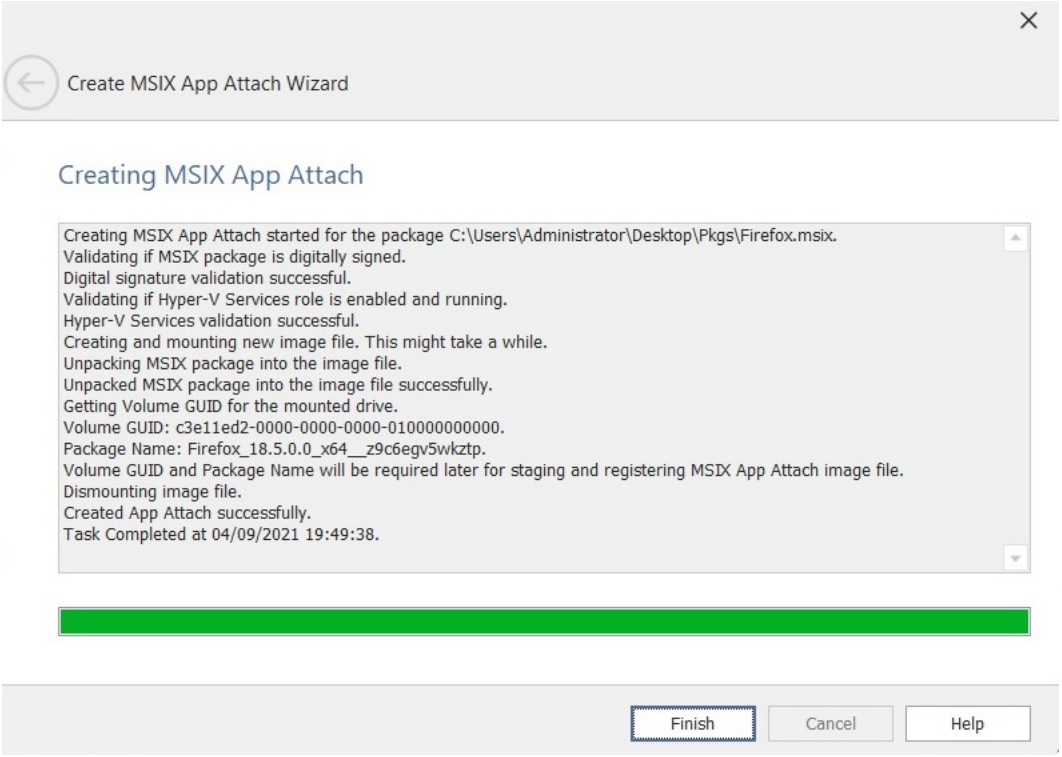
2. In the **Create MSIX App Attach** wizard, the following details are populated by default. If required you can edit the desired fields.

Field	Description
Root Directory Name	This text field enables you to give a root directory name to which the package is being extracted under the Virtual Machines. By default, the package name will be displayed as root directory name. This field is a mandatory field.
Output MSIX Image Type	MSIX packages need to be in a VHD or VHDX format. From the **Output MSIX Image Type** drop down, you can choose any one of the image technologies: • VHD • VHDX
MSIX Image Size	This text field determines the size of the image file to be created. The default value is calculated based on the size of the MSIX file and its four times the file size by default. ***Note*** • *Minimum size for the Image file should be 20MB.*
Output Directory	This field allows you to browse and select a new folder in which the image file will be created. You can enter the folder path as well. This field is validated for a valid folder path and it is mandatory.

3. After selecting, click **Next**.



4. Click **Finish**. The MSIX App Attach is now creating.



5. Upon successful validation, the Image file creation starts in the mentioned output directory.

Note • *You can create multiple App Attach from a single package.*

Note • *If you select the same output directory for the same file, then the image file name will have incremented value.*

For Example: Firefox.msix - Firefox_1.vhd, Firefox_2.vhd

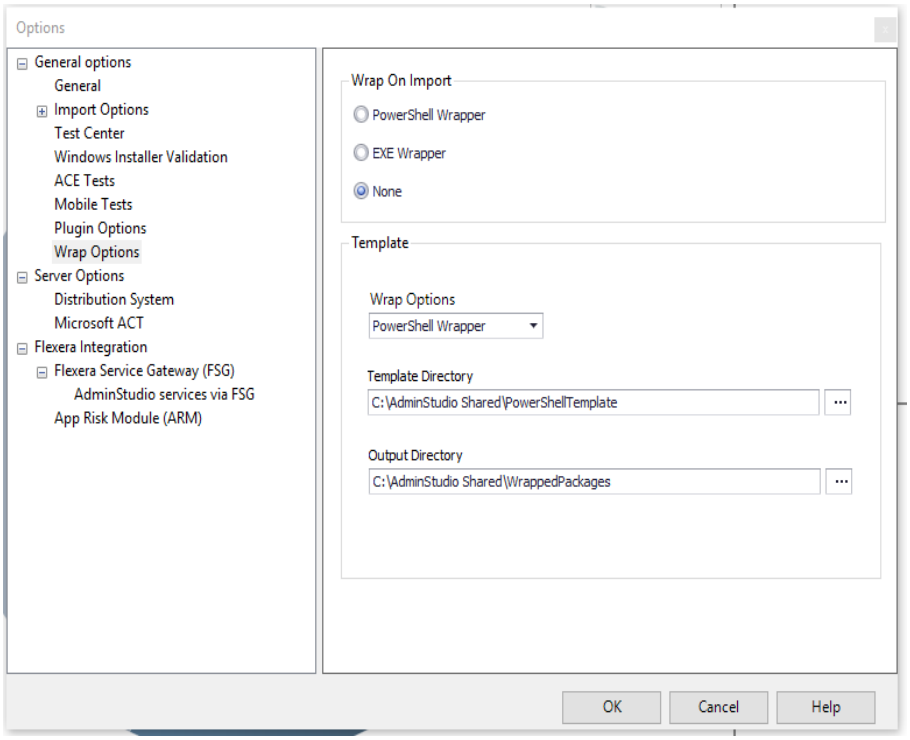
Wrapping MSI/EXE Packages to Wise Script Wrapped EXEs

In AdminStudio, you can now wrap msi/exe packages to Wise script wrapped exes.

Task ***To wrap msi/exe packages to wise script wrapped exes:***

By default, your default template location and a wrapped packages output directory which will help you wrap the packages.

On a new tab of the Application Catalog **Options** dialog box, **Wrap Options**, you can set an option automatically to wrap a package during import from the template, you can specify the location of the template directory and the output directory for PowerShell Wrapping and EXE Wrapping by selecting the option from the “Wrap Options” drop down menu.



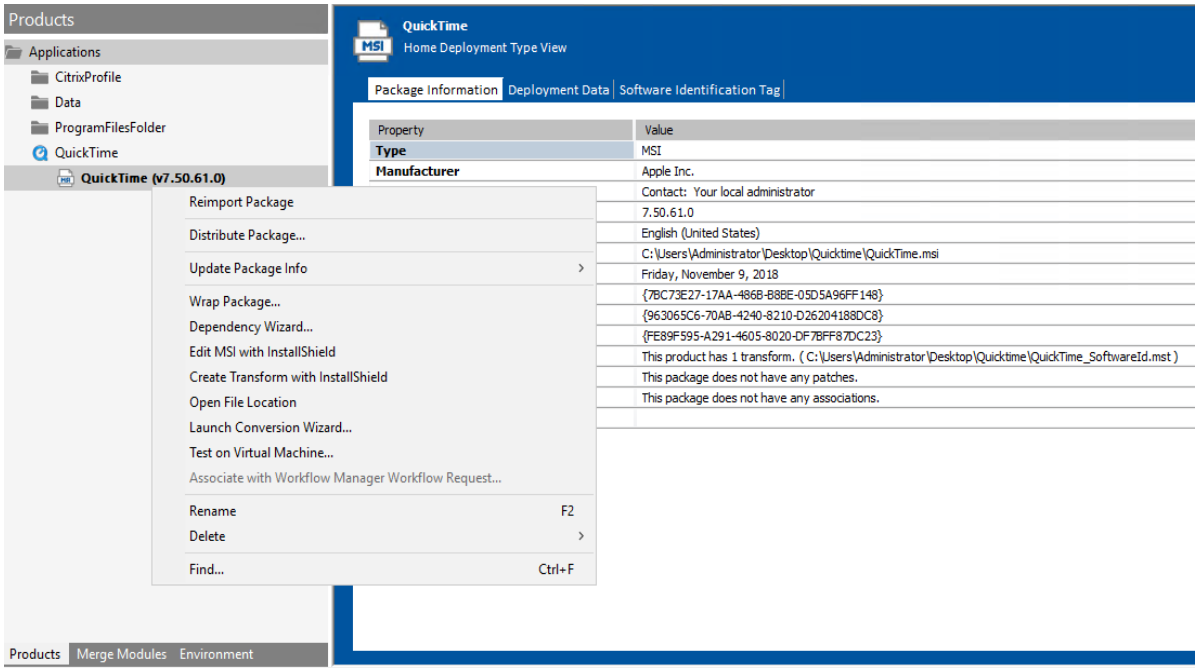
Note • *To use the EXE wrapping feature or WiseScript Package Editor in AdminStudio, which is upgraded from the previous versions, it needs to be re-activated to acquire the license for using WiseScript Package Editor.*

Wrap MSI/EXE Packages Using the Wrap Package Wizard

Edition •

Task ***To wrap msi/exe packages using the Wrap package wizard:***

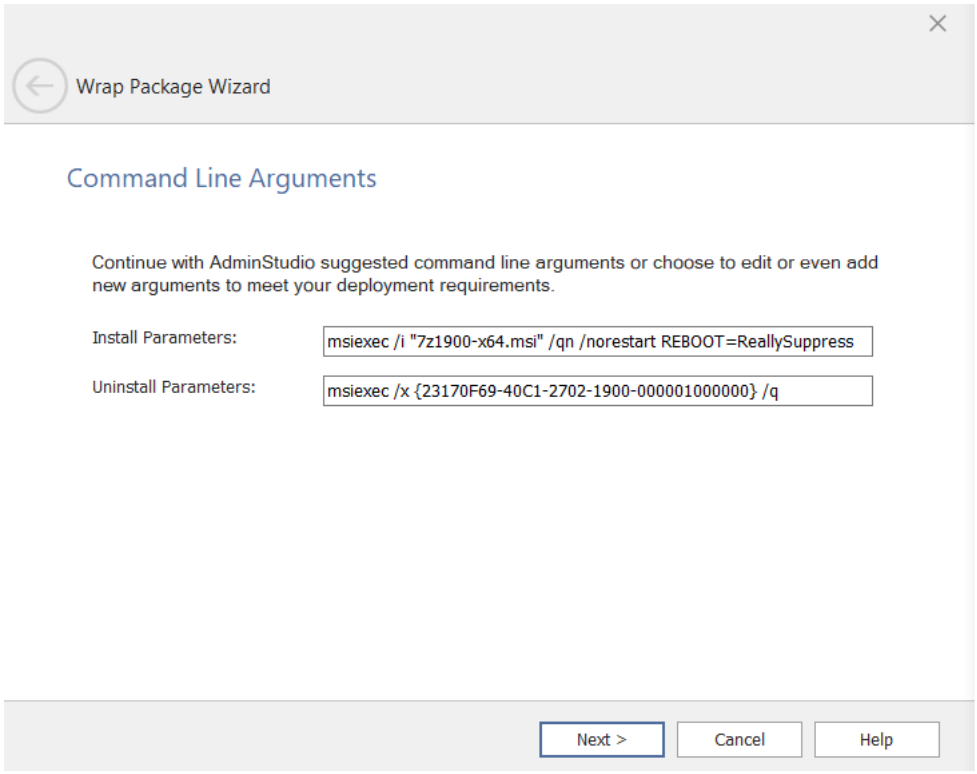
1. Select the Windows Installer (`.msi`) or installation package (`.exe`) package in the Application Catalog tree and select **Wrap Package** from the context menu.



   The Wrap Package Wizard screen appears.

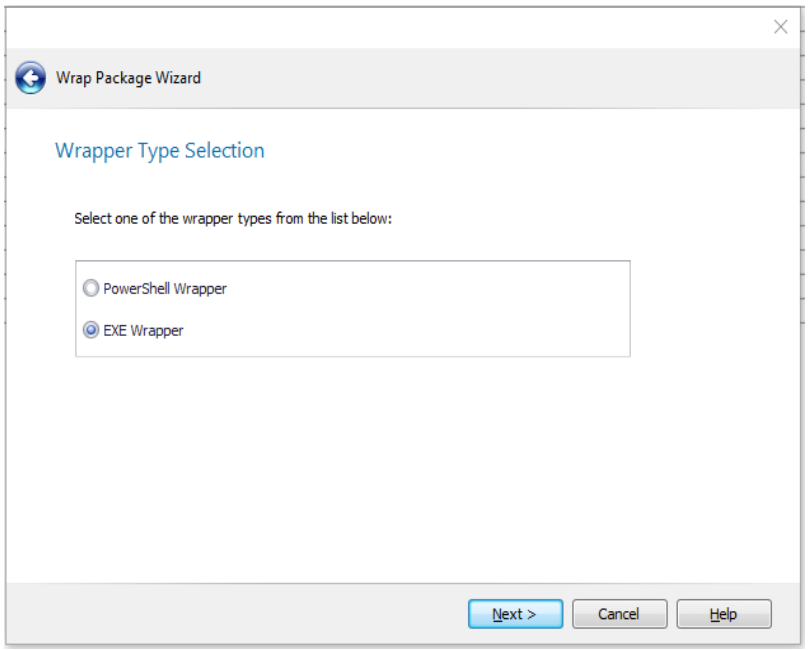
2. Modify the **Install Parameters** and **Uninstall Parameters** and click **Next**.

Note • *Original package will be retained after wrapping and it can be wrapped multiple times to PowerShell and Exe Wrapper Packages under an application.*

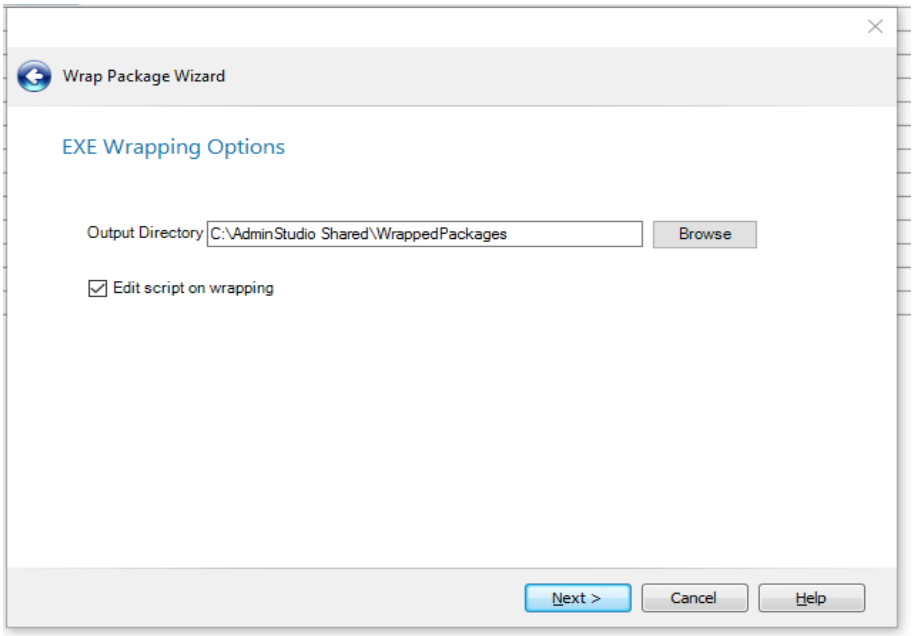


Note • *AdminStudio auto populates the command line - Install Parameters and Uninstall Parameters.*

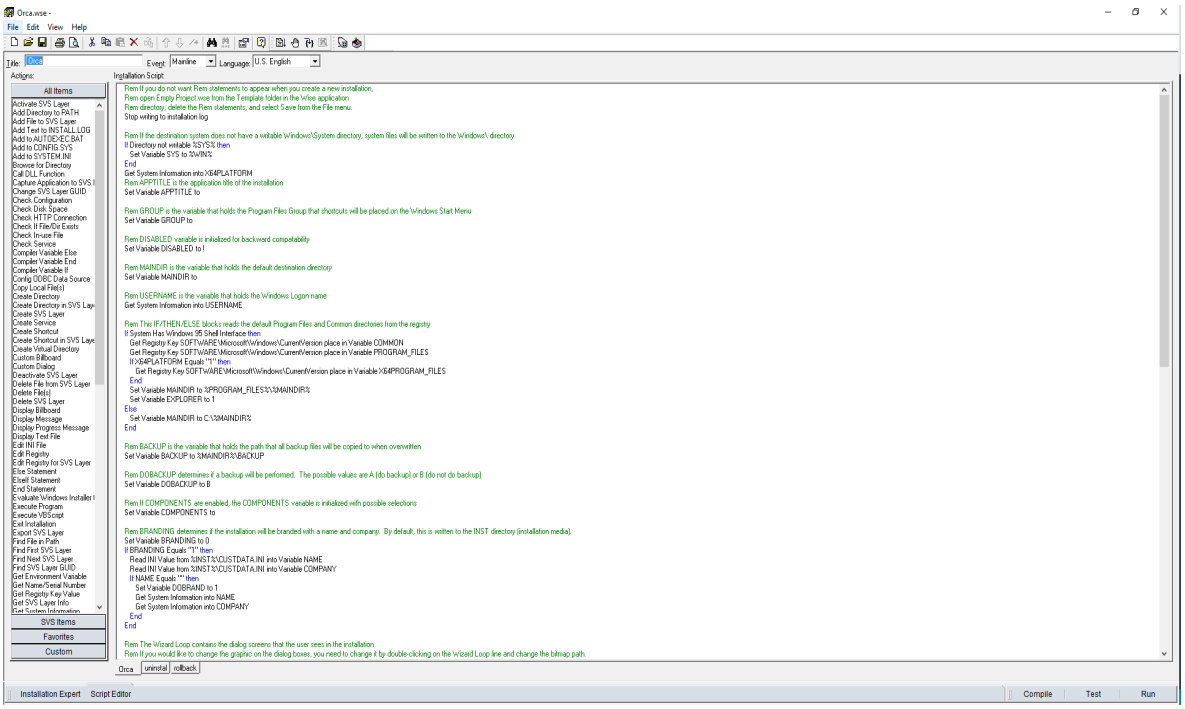
3. On the Wrapper Type Selection, select the required wrapper types from the list and click **Next**.



4. Click **Browser** to select the location for the output directory.



5. Select **Edit script on wrapping** checkbox if you want to view the wise file in the **Wise Script Package Editor** tool and click **Next**.

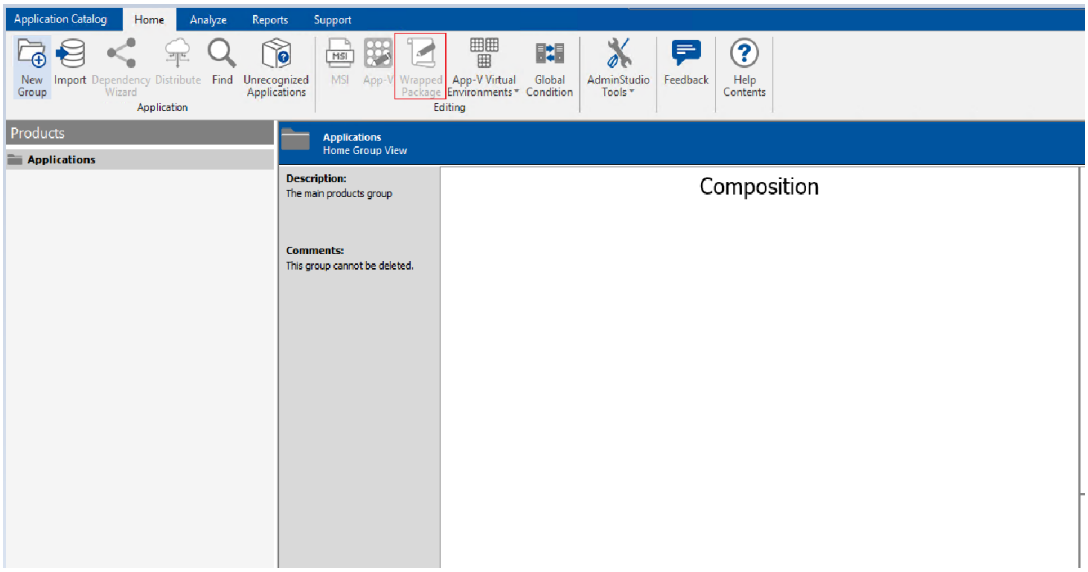


The .wse file appears in the Wise Script Editor tool.

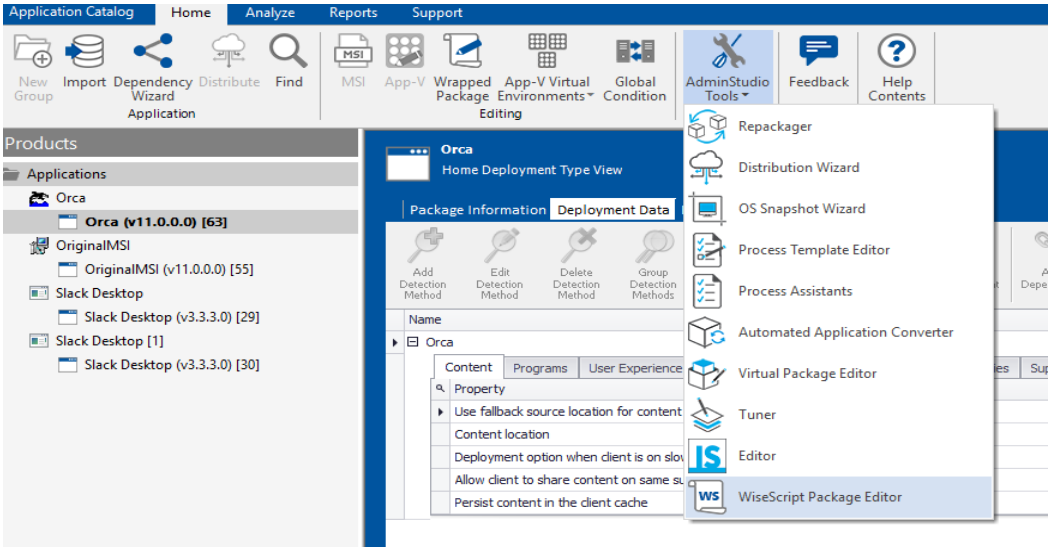
In AdminStudio, you can now edit the wrapped package too. You can select the Windows Installer (`.msi`) or installation package (`.exe`) package in the Application Catalog tree and select **Edit Wrapped Package** from the context menu.

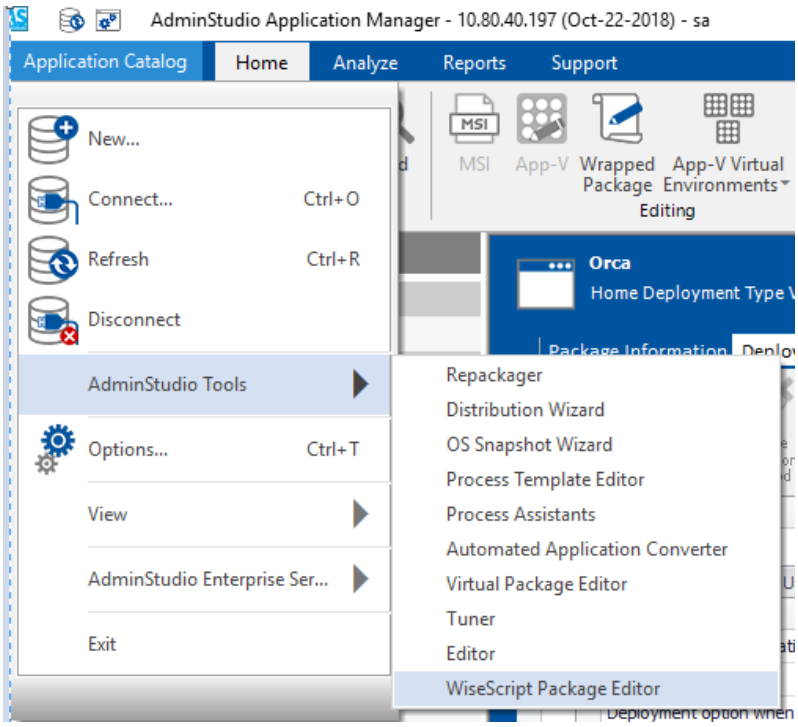
Note • *You will be able to view "Edit Wrapped Package" option in the context menu only if these (Ps1 and Exe) are wrapped packages.*

You can also edit the wrapped package from the ribbon as well.



You can either view the Wise Script Editor tool from the Tools Gallery or Application Manager options.





Performing Application Conflict Testing

You can identify conflicts between packages in the Application Catalog using the Conflict Wizard. You can also check for conflicts between packages and OS Snapshots that have been imported into the Application Catalog.

Note • *Conflict testing can be performed between Windows Installer packages, App-V packages, and OS snapshots in the Application Catalog; other deployment types—such as Citrix XenApp, VMware ThinApp, mobile apps, and legacy applications—are not supported.*

ACE tests are used to detect conflicts between one or more source packages and one or more target packages/OS snapshots in the Application Catalog. Conflict evaluation is done for each source package against each target package.

For information about performing application conflict testing, see the following topics:

- Testing for Conflicts Between Packages
- Testing for Conflicts Between Packages and OS Snapshots

Note • *In previous releases, the Conflict Wizard included a* ***Conflict Rules*** *panel where you could override the default conflict test selections for a specific execution of conflict testing. Starting in AdminStudio 11.5, this panel is no longer included in the Conflict Wizard. Each time you use the Conflict Wizard to perform conflict analysis, the default set of conflict tests will be run. For instructions on how to specify which ACE tests are executed during conflict testing, see Configuring Testing Options on the Analyze Tab and Selecting Tests to Execute.*

Note • *In previous releases, you were able to perform conflict analysis between external packages and packages in the Application Catalog. Starting in AdminStudio 11.5, you can no longer perform conflict testing using external packages; all packages must first be imported into the Application Catalog before conflict testing (or any other type of testing) can be performed.*

Testing for Conflicts Between Packages

To identify conflicts between packages or groups of packages in the Application Catalog, perform the following steps.

Task ***To identify conflicts between packages already in the Application Catalog:***

1. Open Application Catalog and select the **Analyze** tab of the ribbon.
2. In the Application Catalog tree, select the source package (or group of packages) that you want to test and click the **Conflict Wizard** button in the ribbon (or right-click on the package and select **Conflict Wizard** from the shortcut menu). The **Target Information** panel of the Conflict Wizard opens.

   Tip • *When performing conflict analysis using multiple source packages and one or more target packages, Application Catalog will evaluate each source package against each target package. However, if you want Application Catalog to also perform conflict analysis of each source package against every other source package and each target package against every other target package, select the root group in the Application Catalog tree (which is, by default, named the* ***Applications*** *group) and then click the* ***Launch Conflict Wizard*** *button on the* ***Analyze*** *tab of the ribbon.*
3. On the **Target Information** panel, select the packages or groups of packages that you want to use as the target in conflict analysis.

   Note • *If you select the same packages as both source and target, the* ***Next*** *button will be disabled and you will be unable to proceed.*
4. Click **Next**. The **Summary** panel opens.
5. Review the options selected in the **Summary** panel and click **Finish** to begin the conflict identification process. Testing is initiated and messages are displayed in the **Output** window. When testing is finished, a message appears in the **Output** window listing how many warnings and errors were generated.
6. View the test results, as described in Viewing Application Conflicts Test Results.
7. Resolve any issues that were detected, as described in Resolving Issues.

Testing for Conflicts Between Packages and OS Snapshots

To identify conflicts between packages and OS snapshots in the Application Catalog, perform the following steps.

Task ***To identify conflicts between packages and OS snapshots:***

1. Open the Application Catalog **Environment** tab and import an OS Snapshot into the Application Catalog, as described in Importing OS Snapshots.
2. Select the **Analyze** tab of the ribbon.
3. In the Application Catalog tree, select the source package (or group of packages) that you want to test and click the **Conflict Wizard** button in the ribbon (or select **Conflict Wizard** from the shortcut menu). The **Target Information** panel of the Conflict Wizard opens.
4. Select the packages or groups of packages that you want to use as the target in conflict analysis.
5. Click **Next**. The **Target OS Snapshot Information** panel opens.
6. Select the OS snapshots against which you want to compare the source package(s) for conflicts and click **Next**. The **Summary** panel opens.
7. Review the options selected in the **Summary** panel and click **Finish** to begin the conflict identification process. Testing is initiated and messages are displayed in the **Output** window. When testing is finished, a message appears in the **Output** window listing how many warnings and errors were generated.
8. View the test results, as described in Viewing Application Conflicts Test Results.
9. Resolve any issues that were detected, as described in Resolving Issues.

Integrating Analyze With Other Applications

You can integrate Application Catalog Analyze with your Microsoft ACT (Application Compatibility Toolkit) database and display ACT test results. ACT is used to create an inventory of an organization's installed applications, computers, and devices, and to identify and resolve compatibility issues.

- Integrating with Microsoft Application Compatibility Toolkit (ACT)

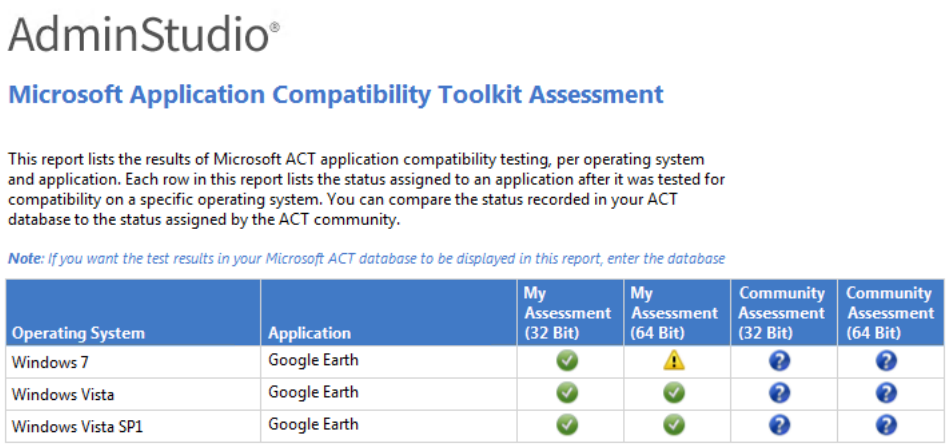
Integrating with Microsoft Application Compatibility Toolkit (ACT)

You can integrate Application Catalog Analyze with your Microsoft ACT (Application Compatibility Toolkit) database and display ACT test results. ACT is used to create an inventory of an organization's installed applications, computers, and devices, and to identify and resolve compatibility issues.

To enable AdminStudio to display data from your Microsoft ACT database in Analyze views and reports, perform the following steps.

Task ***To view Microsoft ACT data in Analyze views and reports:***

1. Enter connection information for your Microsoft ACT database on the **Microsoft ACT** tab of the Application Catalog **Options** dialog box, as described in Entering Microsoft ACT Database Connection Settings.
2. Open the **Analyze** tab.
3. Select a package in the tree that also has associated information in the ACT Database. The **Analyze Deployment Type View** opens.
4. Open the **ACT Summary** tab. Results from the ACT database are displayed on this tab.
5. Open the **Reports** tab.
6. Under **Application Catalog Dashboards** in the ribbon, select **Microsoft Application Compatibility Toolkit**. The **Microsoft Application Compatibility Toolkit Assessment** displays.



AdminStudio®

Microsoft Application Compatibility Toolkit Assessment

This report lists the results of Microsoft ACT application compatibility testing, per operating system and application. Each row in this report lists the status assigned to an application after it was tested for compatibility on a specific operating system. You can compare the status recorded in your ACT database to the status assigned by the ACT community.

Note: If you want the test results in your Microsoft ACT database to be displayed in this report, enter the database

Operating System	Application	My Assessment (32 Bit)	My Assessment (64 Bit)	Community Assessment (32 Bit)	Community Assessment (64 Bit)
Windows 7	Google Earth	✓	⚠	?	?
Windows Vista	Google Earth	✓	✓	?	?
Windows Vista SP1	Google Earth	✓	✓	?	?

7. Review these results, as described in the Microsoft ACT documentation.

Note • *For more information, see Microsoft Application Compatibility Toolkit at:*

http://technet.microsoft.com/library/cc507852.aspx

Viewing and Filtering Test Results

All Application Catalog test results are viewed on the **Analyze** tab. In Analyze views, groups, applications, and packages are assigned a test status in each test group using status icons.

- **Package status**—For packages, the status icon identifies that package's test status.
- **Group/application status**—For groups and applications, Analyze considers all of the packages in that group or application, and displays the status icon for the package that has the status at the highest hierarchical level, as described in Hierarchical Level of Status Icons.

For detailed information about viewing and filtering test results, see the following topics:

- About Status Icons

- Viewing Summary Group/Application Test Results
- Viewing Detailed Package Test Results
  - Viewing Summary Test Results
  - Viewing Operating System Compatibility Test Results
  - Viewing Application Virtualization Compatibility Test Results
  - Viewing Application Virtualization Compatibility Test Results
  - Viewing Remote Application Publishing Compatibility Test Results
  - Viewing Best Practices Test Results
  - Viewing Application Conflicts Test Results
- Viewing Combined Test Results of Bundled Packages
- Filtering Test Results by Suppressing Errors/Warnings

About Status Icons

In **Analyze** views, groups, applications, and packages are assigned a test status in each test group using status icons. For packages, the status icon identifies that package's test status. For groups and applications, Analyze considers all of the packages in that group or application, and displays the status icon for the package that has the status at the highest hierarchical level, as described in Hierarchical Level of Status Icons.

- Status Icons Displayed in the Supportability Risks Columns
- Hierarchical Level of Status Icons

Status Icons Displayed in the Supportability Risks Columns

The following status icons are displayed in the **Supportability Risks** columns of the Analyze Application/Group views.

Table 16-6 • Status Icons Used in Supportability Risks Columns of Analyze Views

Level	Icon	Name	Tested?	Error/Warning Status
1		**Error With Fix**	Yes	• **Errors**—One or more were generated, but at least one has an automated fix which has not yet been applied. See Performing Automatic Issue Resolution. • **Warnings**—One or more could have been generated.
2		**Error**	Yes	• **Errors**—One or more were generated, and none of them has an automated fix. • **Warnings**—One or more could have been generated.

Table 16-6 • Status Icons Used in Supportability Risks Columns of Analyze Views

Level	Icon	Name	Tested?	Error/Warning Status
3		**Warning With Fix**	Yes	• **Errors**—None generated. • **Warnings**—One or more were generated, but at least one has an automated fix which has not yet been applied. See Performing Automatic Issue Resolution.
4		**Warning**	Yes	• **Errors**—None generated. • **Warnings**—One or more were generated, and none of them has an automated fix.
5		**Ready With Suppressed Error(s)**	Yes	• **Errors**—One or more were generated, but all have been suppressed, as described in Filtering Test Results by Suppressing Errors/Warnings. • **Warnings**—One or more could have been generated.
6		**Ready With Suppressed Warning(s)**	Yes	• **Errors**—None generated. • **Warnings**—One or more were generated, but all have been suppressed, as described in Filtering Test Results by Suppressing Errors/Warnings.
7		**Ready**	Yes	• **Errors**—None generated. • **Warnings**—None generated.
8		**Not Applicable**	No	None of the tests in this category were applicable to the package.

Note • *The **Error With Fix** and **Warning With Fix** statuses are considered at a higher level than their **Error**/**Warning** counterparts to indicate that there is an action that you can take (performing automated issue resolution) to alter the final test status of those packages.*

Hierarchical Level of Status Icons

In the previous table, the **Level** column identifies each icon's hierarchical level. To determine which status icon should be displayed for a group or an application, Analyze considers all of the packages in that group or application, and displays the status icon with the highest hierarchical level.

For example, if all of the packages in a group were tested except one, then that group would have a status icon of **Not Run** (level 1). If five packages in a group had no errors or warnings (meaning it they had the **Ready** status, which is level 8), but one contained a warning (level 5), that group would be assigned a status icon of **Warning**.

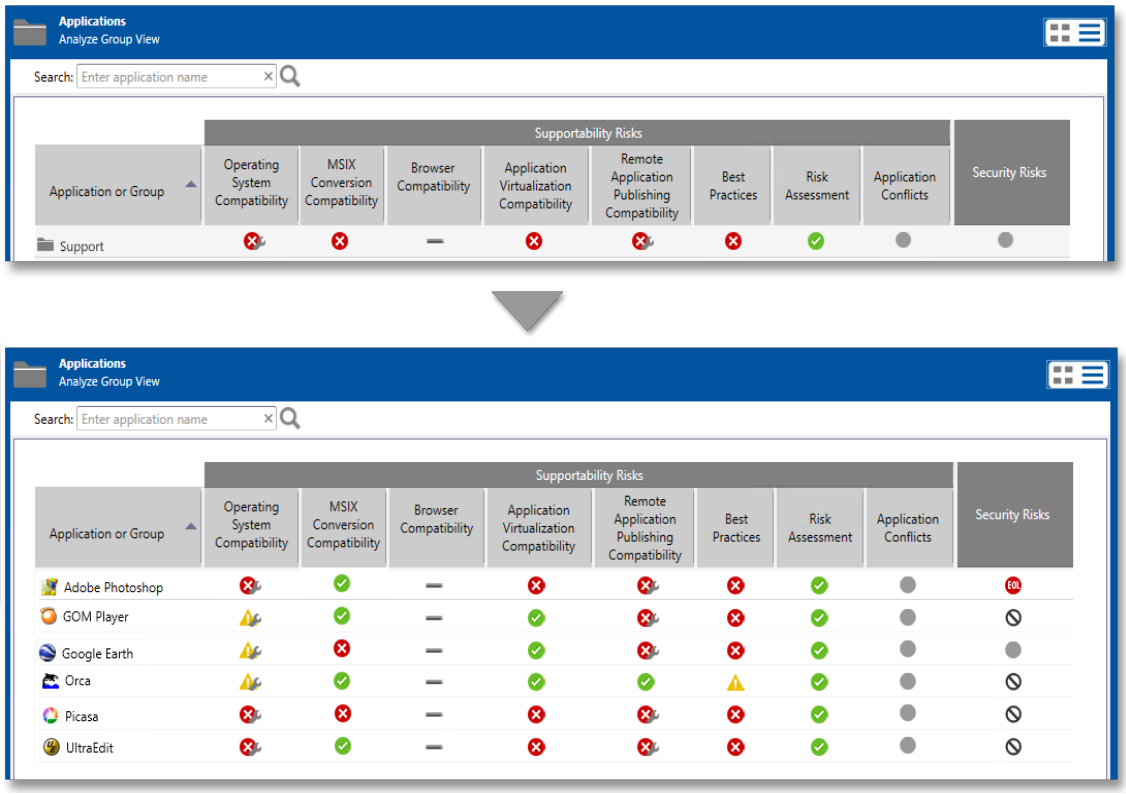


Figure 16-6: Example of Hierarchical Level of Status Icons

Here are a few more examples that are displayed in the Example of Hierarchical Level of Status Icons figure:

- In the **Operating System Compatibility** column, the **Support** group is assigned an overall test status of **Error With Fix** (level 2) because that is the highest level test status of all of the packages in that group (**Warning With Fix** is level 4, while **Warning** is level 5, and **Not Applicable** is level 9).
- In the **Application Conflicts** column, the **Support** group is assigned an overall test status of **Not Run**, because at least one package in that group has a test status of **Not Run** (level 1), which is the highest level test status.
- The Picasa application contains two packages: a Windows Installer package and an App-V package. In the **Best Practices** column, the Picasa application is assigned the overall test status of its Windows Installer package (**Error With Fix**) instead of the status of its App-V package (**Error**) because **Error With Fix** is level 2, while **Error** is level 3.

Viewing Summary Group/Application Test Results

Analyze provides summary views that show display a status icon for groups and applications to indicate their overall test status in each of the following categories:

- Operating System Compatibility
- Application Virtualization Compatibility
- Remote Application Publishing Compatibility
- Best Practices
- Risk Assessment

- Application Conflicts

From these views—**Analyze Group View** and **Analyze Application View**—you can drill-down to access individual package test results on the **Analyze Deployment Type View**, as described in Viewing Detailed Package Test Results.

To view group and application summary test results, perform the following steps.

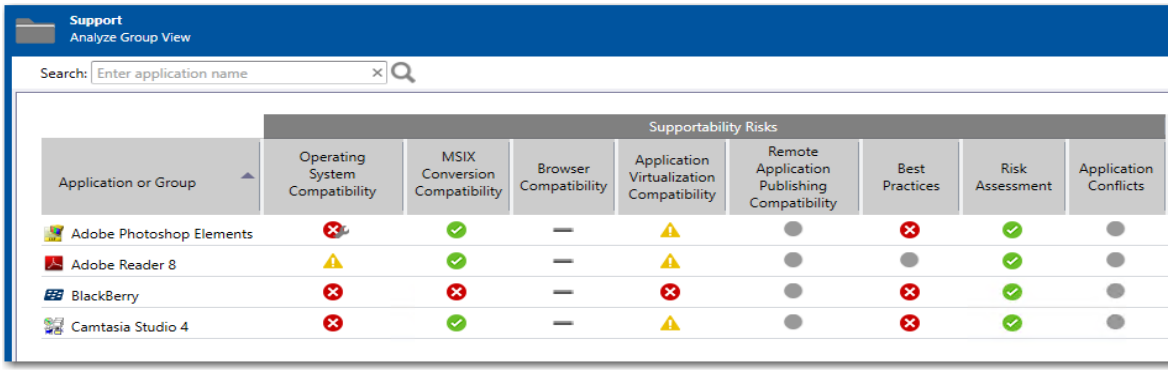
Task ***To view group and application summary test results:***

1. Perform package testing as described in Performing Compatibility, Best Practices, and Risk Assessment Testing, Performing Application Conflict Testing.
2. Open the **Analyze** tab and select a group in the tree.

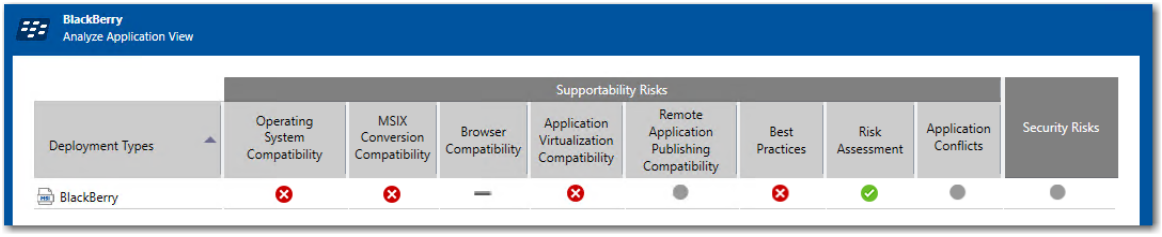
   Tip • *To view test results for the entire Application Catalog, select the root group in the tree (which is **Applications** by default).*

   The **Analyze Group View** opens and displays icons to indicate the overall test status of applications and/or subgroups, as described in About Status Icons, in each of the following columns:

   - Operating System Compatibility
   - Application Virtualization Compatibility
   - Remote Application Publishing Compatibility
   - Best Practices
   - Risk Assessment
   - Application Conflicts
3. In the **Analyze Group View**, you can expand a subgroup node to view its applications, and can expand an application node to view its packages.



   A status icon is displayed in each of the columns to indicate the test status of the tests in that category for the group, application, or package. All of the columns on this view are sortable by clicking on the column heading.
4. To view the status of a single application's packages, select the application in the tree. The **Analyze Application View** opens.



5. To open the **Analyze Deployment Type View** to see the detailed test results for a single package, do one of the following:

   - Select the package in the Application Catalog tree.
   - Double-click the package on the **Analyze Application View**.

Viewing Detailed Package Test Results

Detailed test results for individual packages can be viewed on the **Analyze Deployment Type View** and its subtabs:

- Operating System Compatibility
- MSIX Conversion Compatibility
- Application Virtualization Compatibility
- Remote Application Publishing Compatibility
- Best Practices
- Risk Assessment
- Application Conflicts

For information on viewing test results for an individual package, see the following topics:

- Viewing Summary Test Results
- Viewing Operating System Compatibility Test Results
- Viewing MSIX Conversion Compatibility Test Results
- Viewing Application Virtualization Compatibility Test Results
- Viewing Application Virtualization Compatibility Test Results
- Viewing Remote Application Publishing Compatibility Test Results
- Viewing Best Practices Test Results
- Viewing Application Conflicts Test Results

Viewing Summary Test Results

By clicking on each of the test category in the **Supportability Risks** column in the **Analyze Deployment Type View** lists detailed test totals for each test group and test category, including the number of:

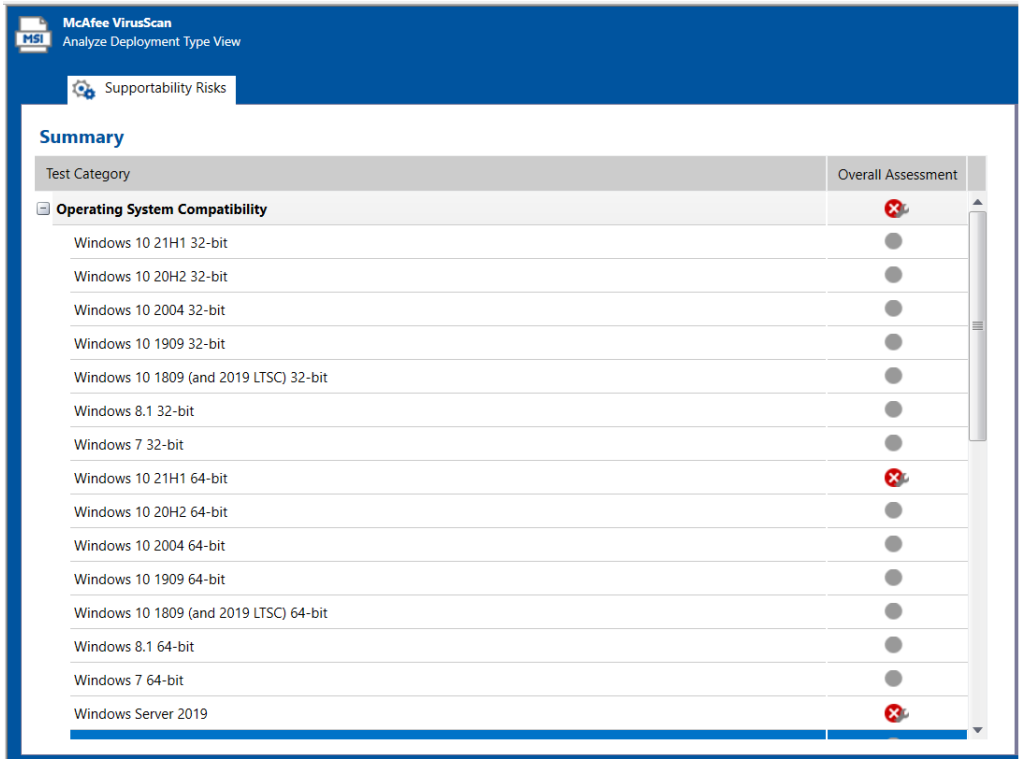
- Tests executed

- Errors and warnings generated
- Errors and warnings for which an auto fix is available
- Errors and warnings that are suppressed

Also, a status icon identifies the package's test status in each of the test groups and test categories.

Task ***To view summary test results:***

1. Select the **Analyze** tab in the Application Manager ribbon.
2. Select a package in the tree. The **Supportability Risks** column of the **Analyze Deployment Type View** opens.



The following information is displayed for each test group and test category:

Property	Description
Executed	Number of tests executed in that test group or test category. This number corresponds to the number of tests selected in that test group/category on the **Select Tests to Execute** dialog box.
Errors	Number of non-suppressed errors generated in that test group/category. See Filtering Test Results by Suppressing Errors/Warnings.
Warnings	Number of non-suppressed warnings generated in that test group/category. See Filtering Test Results by Suppressing Errors/Warnings.

Property	Description
Auto Fix Available	Total number of errors and warnings generated in that test category for which an automatic fix is available. See Resolving Issues for more information.
Issues Suppressed	Total number of errors and warnings generated in that test category that have been suppressed. See Filtering Test Results by Suppressing Errors/Warnings.
Overall Assessment	Icon indicating the overall test status of the package, as described in About Status Icons.

3. To view more detailed results, proceed as described in one of the following topics:
   - Viewing Operating System Compatibility Test Results
   - Viewing MSIX Conversion Compatibility Test Results
   - Viewing Application Virtualization Compatibility Test Results
   - Viewing Remote Application Publishing Compatibility Test Results
   - Viewing Best Practices Test Results
   - Viewing Application Conflicts Test Results

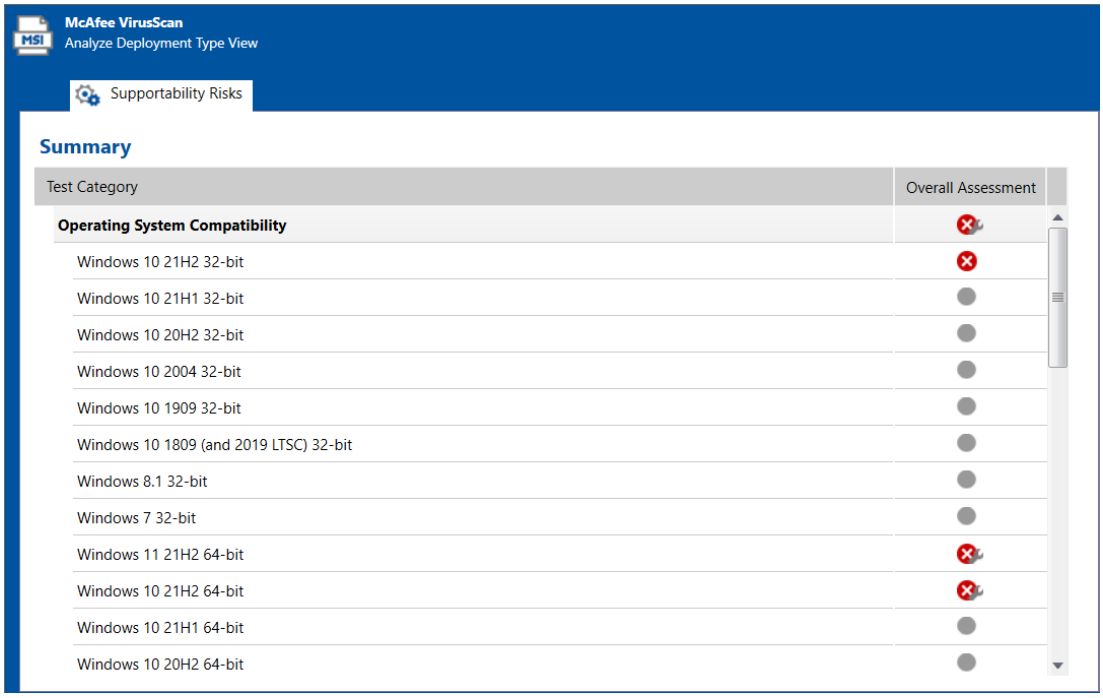
Viewing Operating System Compatibility Test Results

The **Operating System Compatibility** tab of the **Analyze Deployment Type View** lists all of the individual errors and warnings that were generated by tests in the **Operating System Compatibility** test group for the package.

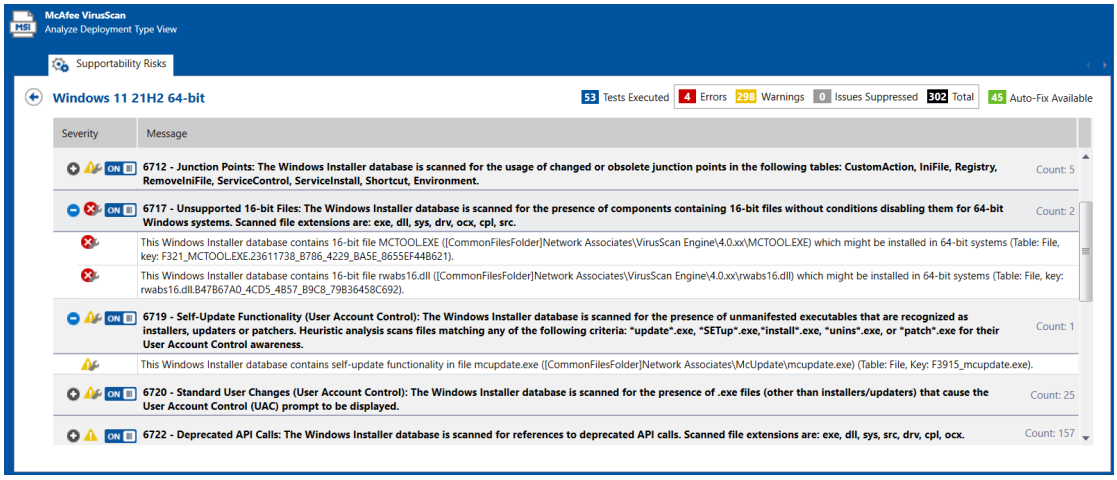
On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Task ***To view operating system compatibility test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Open the **Operating System Compatibility** test group. The errors and warnings generated by tests in the **Operating System Compatibility** test group are listed.



5. Click on one of the test names. The detail view for that test is displayed:



The following information is displayed:

Property	Description
Test Category	Name of test category in the **Operating System Compatibility Test** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.

Property	Description
Test Number and Name	For each test, the number and name is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: • **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. • **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.
Suppress Icon	The ON/OFF icon indicates whether an issue is suppressed. If the ON icon is displayed, the issue is active: If you click the ON icon, it changes to an OFF icon, indicating that the issue is suppressed: For more information, see Filtering Test Results by Suppressing Errors/Warnings.

6. For more detailed information on **Operating System Compatibility** tests and the issues that they generate, including information on how to resolve these issues, see Operating System Compatibility Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the **Operating System Compatibility** tab and selecting **More Info** from the shortcut menu.*

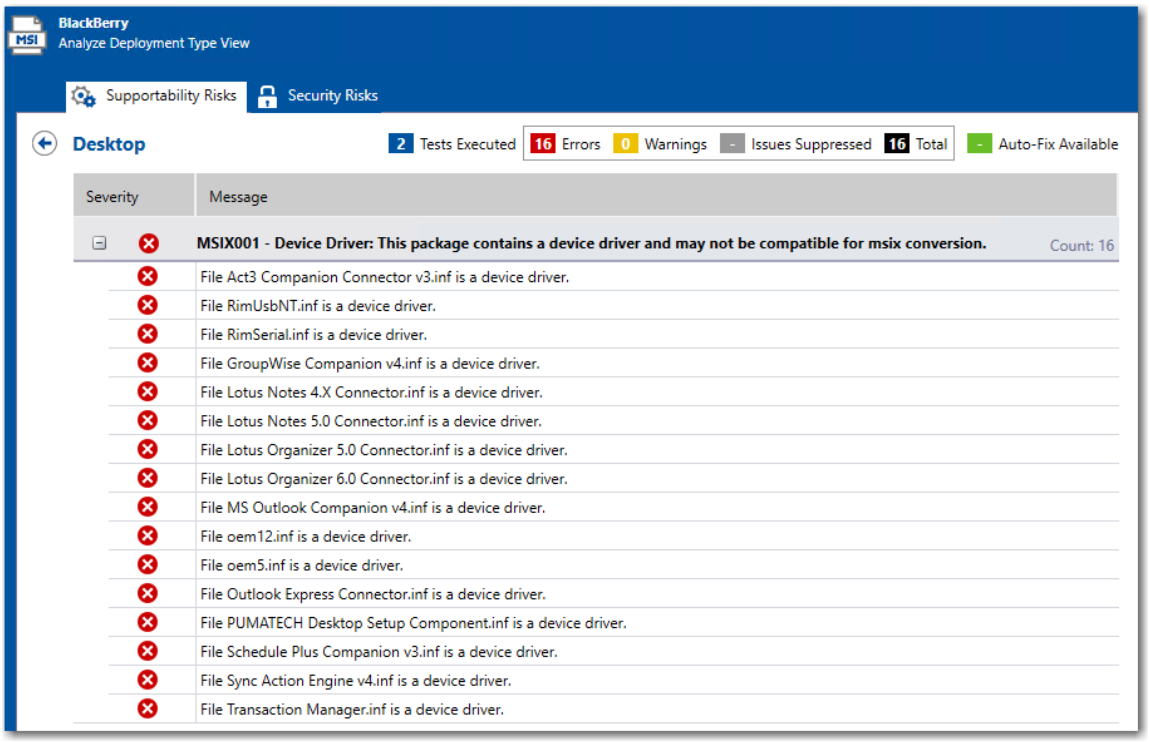
Viewing MSIX Conversion Compatibility Test Results

You can view all of the individual errors and warnings that were generated by tests in the **MSIX Conversion Compatibility** test group on the **Supportability Risks** tab of the **Analyze Deployment Type View**.

If you click on the icon displayed in the **MSIX Conversion Compatibility** area of the **Summary** view of the **Supportability Risks** tab, a detailed view opens where you can read all of the error and warning messages that were generated.

Task ***To view MSIX Conversion compatibility test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** view of the **Supportability Risks** tab of the **Analyze Deployment Type View** opens.
4. Open the **MSIX Conversion Compatibility** test group and click on the **Desktop** entry. A detailed view of the **Supportability Risks** tab opens, listing the errors and warnings generated by the **MSIX Conversion Compatibility** test group.



Viewing Application Virtualization Compatibility Test Results

Application Catalog performs application virtualization compatibility testing to determine if a Windows Installer package is a suitable candidate for conversion to Microsoft App-V, Citrix XenApp, VMware ThinApp.

The **Application Virtualization Compatibility** tab of the **Analyze Deployment Type View** lists all of the individual errors, warnings, and informational messages that were generated when application virtualization compatibility testing was performed.

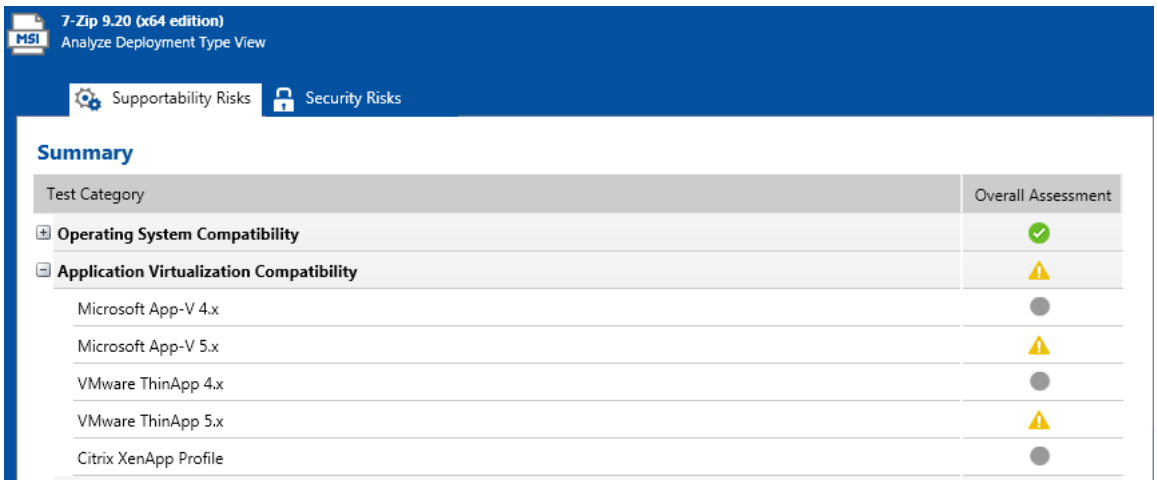
Note • *See also Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter.*

Task ***To view application virtualization compatibility test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.

   Note • *The Application Virtualization Compatibility tests are always run any time that you run tests in Analyze. However, the selections you make on the* ***Select Tests to Execute*** *dialog box determine which virtual formats to display in test results.*

2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Select the **Application Virtualization Compatibility** tab. The errors and warnings generated by application virtualization compatibility testing are listed.



The following information is displayed:

Property	Description
Test Category	Identifies the test category as one of the following virtualization technologies: • Microsoft App-V 4.x • Microsoft App-V 5.x • VMware ThinApp 4.x • VMware ThinApp 5.x • Citrix XenApp When each test category is expanded, the tests in that category that generated errors or warnings are listed.

Property	Description
Test Name	For each test, the name is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: ● **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. ● **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.
Icon	For each issue, one of the following icons is displayed: ● Informational ● Error ● Warning The Application Virtualization Compatibility tab includes one additional status icon that is not used in the other test groups called the informational icon: This icon identifies issues that require that Automated Application Converter will need to automatically repackage this Windows Installer package during the conversion process to a virtual package. Since these issues by themselves do not necessarily indicate a warning or error, they are considered “informational” issues. For more information, see About Status Icons and Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter.

5. For more detailed information on the issues generated by **Application Virtualization Compatibility** tests, including information on how to resolve these issues, see Application Virtualization Compatibility Tests.

Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter

AdminStudio's Automated Application Converter is used to convert packages to a virtual format. When a package is added to its **Packages** tab, Automated Application Converter does its own check to identify that package's virtualization readiness status and assigns it one of the following statuses (which are slightly different than the statuses assigned by Analyze):

Table 16-7 • Automated Application Converter Package Statuses

Status	Icon	Description
Ready		Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.
Requires repackaging		Package must be repackaged before it can be successfully virtualized.
Virtualization not supported		Automated Application Converter has determined that virtualization is not supported.
Virtualization not recommended		Automated Application Converter has determined that this package is not recommended for virtualization.
Unknown		The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

The following table explains how the Application Catalog application virtualization compatibility package statuses of **Ready**, **Not Ready**, and **Fixable Issues** correspond to statuses assigned to individual packages when they are added to Automated Application Converter:

Table 16-8 • Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter

Analyze Status	Automated Application Converter Status	Description
Ready	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** *• If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the* `IniFile` *table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.

Table 16-8 • Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter

Analyze Status	Automated Application Converter Status	Description
Not Ready (Error)	Virtualization not supported	Virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** *• Packages with a status of* ***Virtualization not supported*** *will not be virtualized in Automated Application Converter. In order to virtualize the package, you must first override the status and change it to* ***Ready*** *or* ***Requires repackaging****.* ***Important*** *• For more information, see Application Virtualization Compatibility Tests.*
	Virtualization not recommended	This package is not recommended for virtualization due to one of the following issues: • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** *• For more information, see Application Virtualization Compatibility Tests.*
Fixable Issues (Warning)	Virtualization not recommended	This package is not recommended for virtualization because it does not contain a shortcut. It is possible to resolve this issue by manually creating a shortcut in the package before performing conversion. ***Note*** *• For more information, see Application Virtualization Compatibility Tests.*

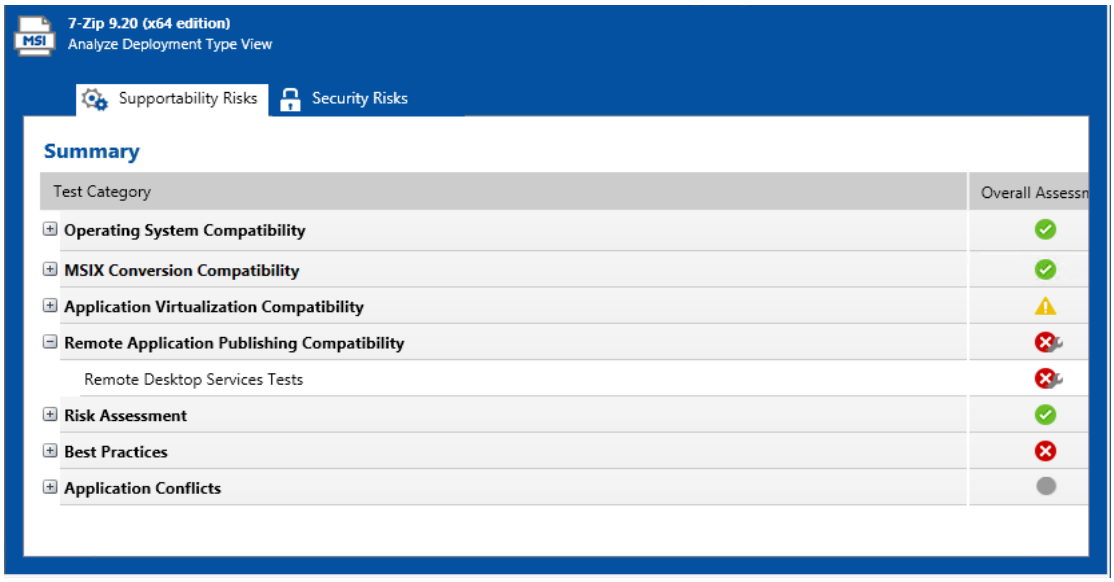
Viewing Remote Application Publishing Compatibility Test Results

The **Remote Application Publishing Compatibility** tab of the **Analyze Deployment Type View** lists all of the individual errors and warnings that were generated by tests in the **Remote Desktop Services Tests** test group for the package.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Task ***To view remote application publishing compatibility test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Select the **Remote Application Publishing Compatibility** tab. The errors and warnings generated by tests in the **Remote Desktop Services Tests** test group are listed.



The following information is displayed:

Property	Description
Test Category	Name of test category for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.
Test Number	For each test, the number is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: • **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. • **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.

Property	Description
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.
Suppress Icon	The ON/OFF icon indicates whether an issue is suppressed. If the ON icon is displayed, the issue is active: If you click the ON icon, it changes to an OFF icon, indicating that the issue is suppressed: For more information, see Filtering Test Results by Suppressing Errors/Warnings.

5. For more detailed information on the issues generated by **Remote Application Publishing Compatibility** tests, including information on how to resolve these issues, see Remote Application Publishing Compatibility Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the* ***Remote Application Publishing Compatibility*** *tab and selecting* ***More Information*** *from the shortcut menu.*

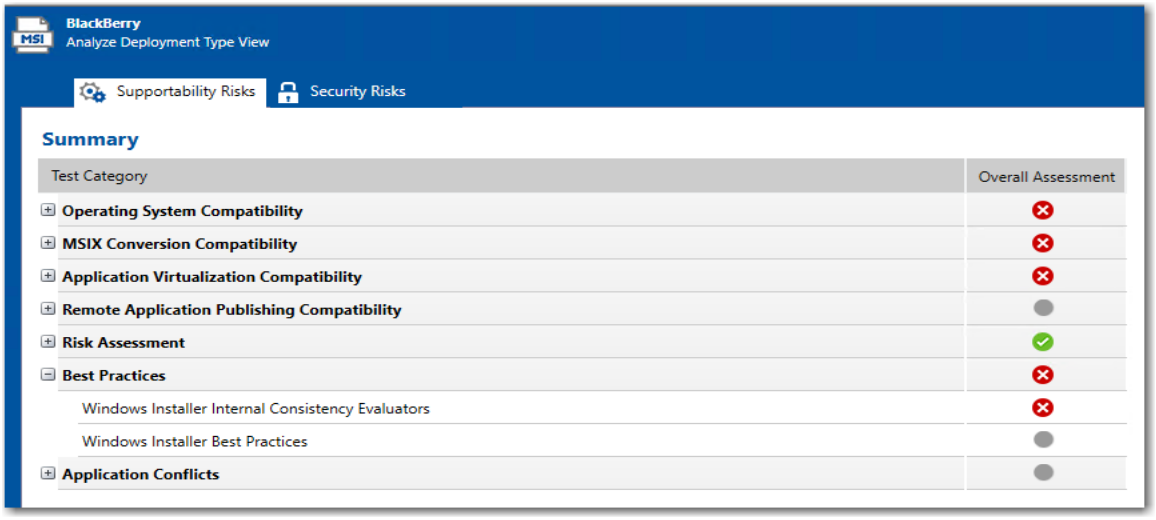
Viewing Best Practices Test Results

The **Best Practices** tab of the **Analyze Deployment Type View** lists all of the individual errors and warnings that were generated by tests in the **Best Practices** test group for the package.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Task ***To view Best Practices test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Select the **Best Practices** tab. The errors and warnings generated by tests in the Best Practices test group are listed.



The following information is displayed:

Property	Description
Test Category	Name of test category in the **Best Practices** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.
Test Number	For each test, the number is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: ● **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. ● **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.
Icon	For each error or warning, one of the following icons is displayed: ● Error ● Error With Fix ● Warning ● Warning With Fix For more information, see About Status Icons.

Property	Description
Suppress Icon	The ON/OFF icon indicates whether an issue is suppressed. If the ON icon is displayed, the issue is active: If you click the ON icon, it changes to an OFF icon, indicating that the issue is suppressed: For more information, see Filtering Test Results by Suppressing Errors/Warnings.

5. For more detailed information on the issues generated by **Best Practices** tests, including information on how to resolve these issues, see Best Practices Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the* ***Best Practices*** *tab and selecting* ***More Information*** *from the shortcut menu.*

Viewing Risk Assessment Test Results

The **Risk Assessment** tab of the **Analyze Deployment Type View** lists all of the individual errors and warnings that were generated by tests in the **Risk Assessment** test group for the package.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Task ***To view Risk Assessment test results:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Select the **Risk Assessment** tab. The errors and warnings generated by tests in the Risk Assessment test group are listed.

   The following information is displayed:

Property	Description
Test Category	Name of test category in the **Risk Assessment** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.

Property	Description
Test Number	For each test, the number is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: • **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. • **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.
Suppress Icon	The ON/OFF icon indicates whether an issue is suppressed. If the ON icon is displayed, the issue is active: If you click the ON icon, it changes to an OFF icon, indicating that the issue is suppressed: For more information, see Filtering Test Results by Suppressing Errors/Warnings.

5. For more detailed information on the issues generated by **Risk Assessment** tests, including information on how to resolve these issues, see Risk Assessment Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the **Risk Assessment** tab and selecting **More Information** from the shortcut menu.*

Viewing Application Conflicts Test Results

The **Application Conflicts** tab of the **Analyze Deployment Type View** lists all of the individual errors and warnings that were generated by tests in the **Application Conflicts** test group for the package when conflict analysis was performed.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Task ***To view application conflict test results:***

1. Perform conflict analysis, as described in Performing Application Conflict Testing.
2. Select the **Analyze** tab in the Application Manager ribbon.
3. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Select the **Application Conflicts** tab. The errors and warnings generated by tests in the Application Conflicts test group are listed.

The following information is displayed:

Property	Description
Test Category	Name of test category in the **Application Conflicts** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.
Test Number	For each test, the number is listed in bold, followed by a description. When the test is expanded, the errors or warnings generated by this test are listed; these error/warning messages list information specific to the package that explains why the error or warning was generated.
Count	Two counts are listed: ● **Test category count**—Total number of errors/warnings that were generated by all of the tests in the test category for the selected package. ● **Test count**—Total number of errors/warnings that were generated by the specific test for the selected package.

Property	Description
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.
Suppress Icon	The ON/OFF icon indicates whether an issue is suppressed. If the ON icon is displayed, the issue is active: If you click the ON icon, it changes to an OFF icon, indicating that the issue is suppressed: For more information, see Filtering Test Results by Suppressing Errors/Warnings.

5. For more detailed information on the issues generated in by **Application Conflicts** tests, including information on how to resolve these issues, see Application Conflicts Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the **Application Conflicts** tab and selecting **More Information** from the shortcut menu.*

Viewing Combined Test Results of Bundled Packages

When a complex installer executable (.exe), Apple disk image package (.dmg), or Apple installer package (.pkg) file is tested, its bundled packages are also tested and the test results are combined and displayed in Analyze.

- Viewing Combined Test Results of Child Windows Installer Packages of Complex Installer Executables
- Viewing Combined Test Results of Child Applications of PKG and DMG Installers

Viewing Combined Test Results of Child Windows Installer Packages of Complex Installer Executables

When a complex installer .exe file is tested, its child Windows Installer packages are also tested and the test results are combined and displayed in on the **Analyze** tab.

To view the detailed test results, click on the individual **Supportability Risk** listed under the test category.

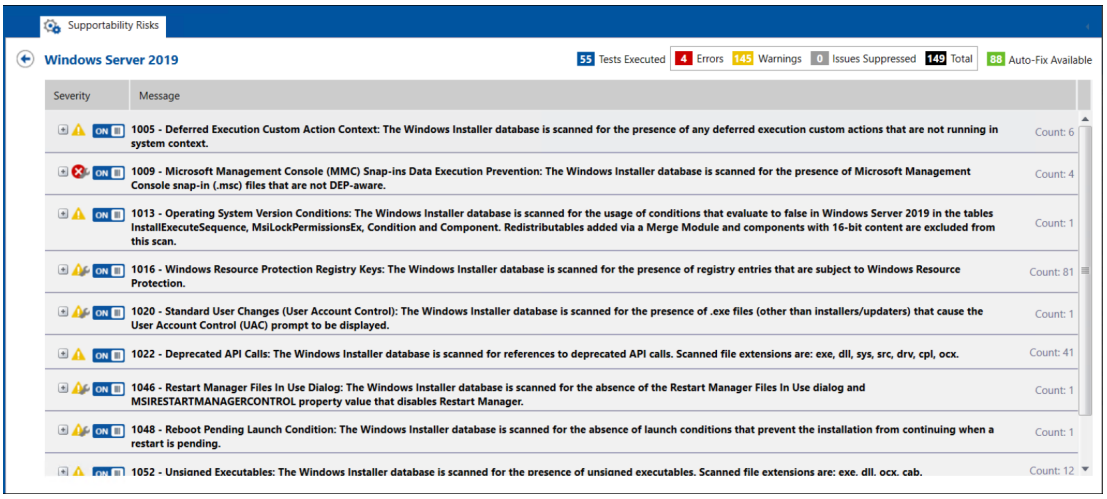


Figure 17: Combined Suite Installer Test Results

Viewing Combined Test Results of Child Applications of PKG and DMG Installers

If an Apple installer package (`.pkg`) or disk image (`.dmg`) contains child packages bundled within it, those child packages are also tested when the parent package is tested. Test results of the parent package and all of its child packages are combined and are displayed in Analyze.

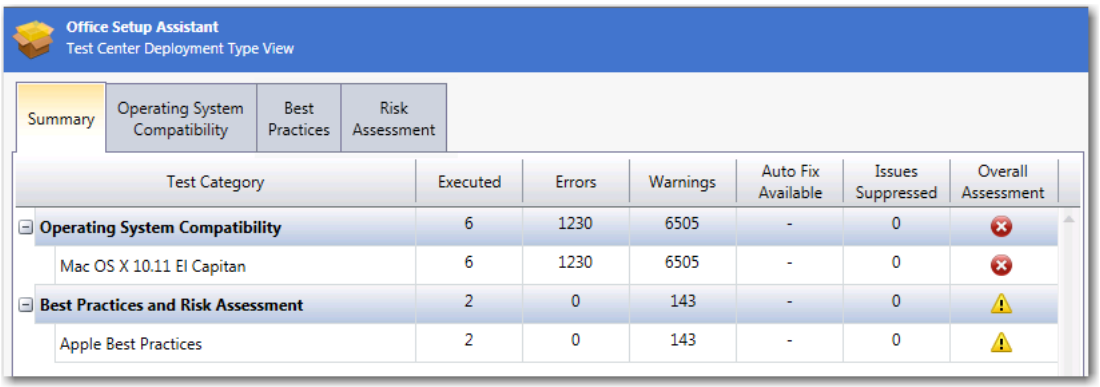


Figure 18: Consolidated Test Results for Apple Installer Package (.pkg)

When you view detailed test results, the name of the child `.pkg`, `.dmg`, or `.app` file that generated the error or warning is listed.

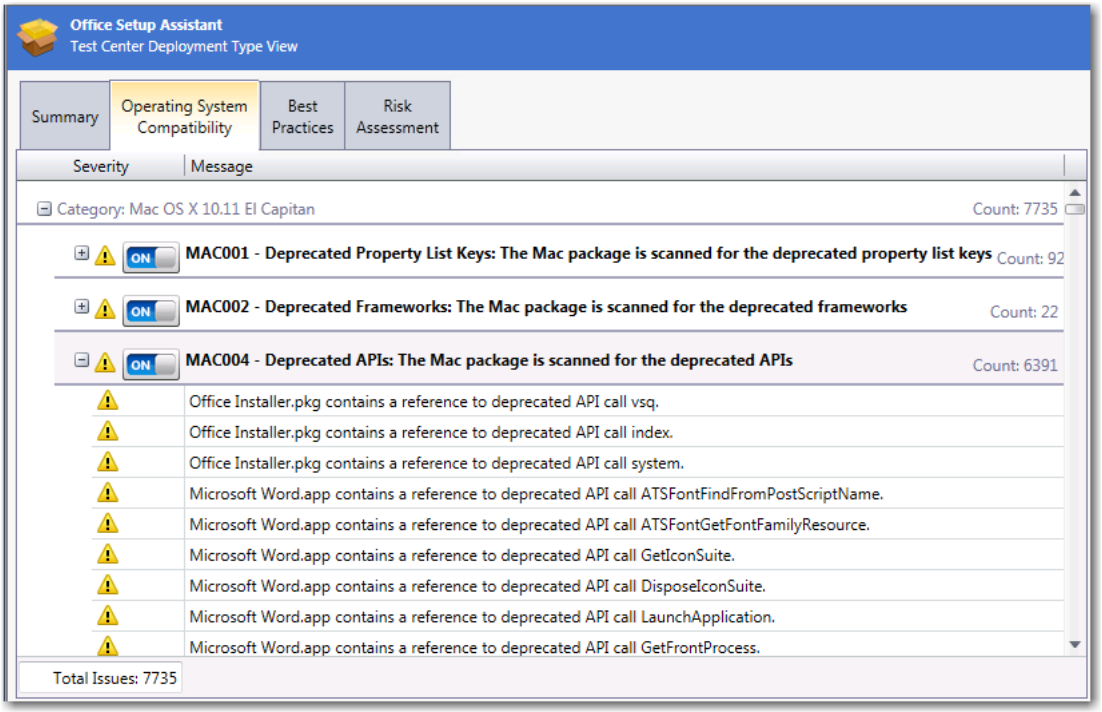


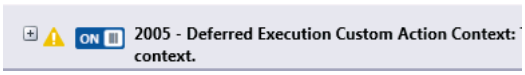
Figure 19: Detailed Consolidated Test Results for Apple Installer Package (.pkg)

Filtering Test Results by Suppressing Errors/Warnings

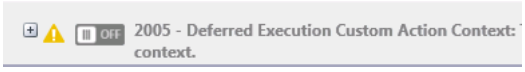
If you do not want Analyze to include the issues generated by a particular test in the overall package status, and you do not want them included in summary issue counts, you can choose to suppress that test by clicking the **Suppress** (ON/OFF) button on the following tabs of the **Analyze Deployment Type View**:

- Operating System Compatibility
- Best Practices
- Risk Assessment
- Application Conflicts

When you suppress a test, its **Suppress** button switches from its ON state:



to its OFF state, and its error or warning icon switches to gray:



You may choose to suppress the test results of a test that generates known issues at your organization which do not need additional corrections.

When a test is suppressed, the following occurs:

- **Errors/warnings not included in summary totals**—The test's issues are no longer included in the overall **Errors** and **Warnings** counts on **Summary** tab of the **Analyze Deployment Type View**. Instead, they are listed on the **Issues Suppressed** column.
- **Test's status is ignored in overall status**—The test's status is ignored when assigning an overall status to a package.
- **Test's issues are not fixed**—Issues generated by suppressed tests are not resolved during automatic issue resolution even if an auto-fix exists.

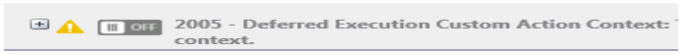
Note • *Even though the test is suppressed, it is still run each time testing is performed. The only way to turn off the execution of a test is to deselect it on the* ***Select Tests to Execute*** *dialog box.*

To suppress a test's issues, perform the following steps:

Task ***To suppress the errors and warnings for a test:***

1. Select the **Analyze** tab in the Application Manager ribbon.
2. Select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
3. Select the tab of the **Analyze Deployment Type View** that contains the issues that you want to suppress.
4. Next to the name of the test that has generated issues that you want to suppress, click its **Suppress** button. The button will switch from its ON state:

   to its OFF state, and its error or warning icon will switch to gray:

5. Open the **Summary** tab of the **Analyze Deployment Type View**, and notice that the count in the **Issues Suppressed** column for that test category has been increased.

Resolving Issues

Some of the tests in the **Operating System Compatibility**, **Best Practices**, **Risk Assessment**, and **Application Conflicts** tests groups have automatic fixes available, and can be automatically fixed—via transform—by clicking the **Resolve Issues** button in the ribbon. For other tests, a manual fix is required.

For information on resolving issues generated by Analyze testing, see the following topics:

- Performing Automatic Issue Resolution
- Performing Manual Issue Resolution

Performing Automatic Issue Resolution

Some of the tests in the **Operating System Compatibility**, **Best Practices**, **Risk Assessment**, and **Application Conflicts** tests groups have automatic fixes available, and can be automatically fixed—via transform—by clicking the **Resolve Issues** button in the ribbon.

- About Automatic Issue Resolution
- Automatically Resolving Issues
- Issue Resolution and the Software Repository

About Automatic Issue Resolution

If an error or warning that was generated for a package has an automatic fix available, it is assigned one of the following icons:

Table 16-9 • Status Icons Indicating Auto-Fix Available

Icon	Meaning
	Error With Fix
	Warning With Fix

To determine whether a test has an associated auto fix, review the **Resolution** section in the description of the test in the Analyze Tests section of the documentation.

For some of the tests in the **Operating System Compatibility** test groups, you have the option of specifying how you want Application Catalog to resolve automatically resolvable issues. You can instruct Application Catalog to perform the *basic* auto fix, the *advanced* auto fix, or not to fix issues generated by the test.

Table 16-10 • Default Fix Options

Fix Type	Description
Do not resolve this issue automatically	Select this option if you do not want Application Catalog to automatically resolve any issues generated by this test.
Apply the basic auto fix	Select this option if you want Application Catalog to resolve issues generated by this test by applying the basic auto fix. Applying the **basic auto fix** is relatively safe. It results in minimal changes to an MSI package via a Windows Installer transform. It does not change the target system's security or a system policy.

Table 16-10 • Default Fix Options

Fix Type	Description
Apply the advanced auto fix	Select this option if you want Application Catalog to resolve issues generated by this test by applying the advanced auto fix. Applying the **advanced auto fix** may result in a loss of functionality, and it may not resolve all types of issues. This type of fix may change the target system's security or a system policy. One example of an advanced auto fix is the removal of a registry key that is protected by Windows Resource Protection.

Note • *For some tests, one of these options is disabled. For others, all options are disabled.*

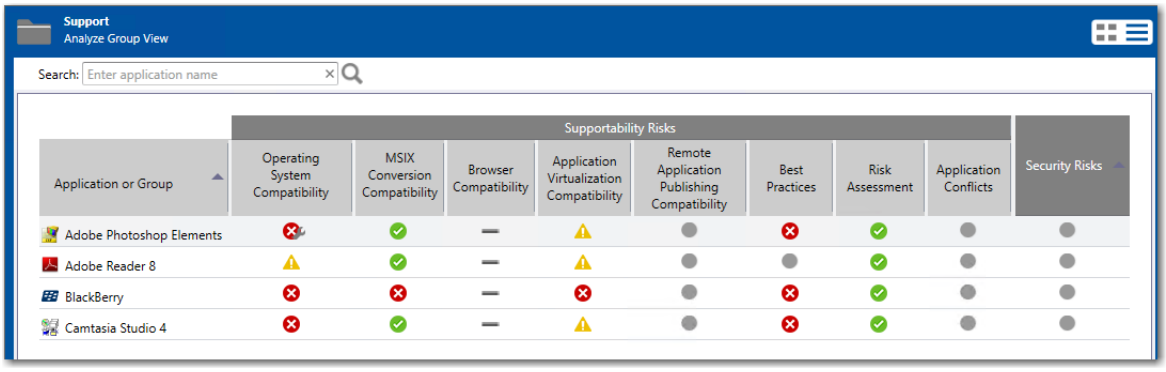
For information on how to specify whether to perform a basic or advanced auto fix for a test, see Setting Automatic Fix Preferences for Operating System Compatibility Tests.

Automatically Resolving Issues

To perform automatic issue resolution, perform the following steps:

Task ***To perform automatic issue resolution:***

1. Select the **Analyze** tab in the Application Manager ribbon.
2. Select a group, application, or package in the tree to open the **Analyze Group View**, **Application View**, or **Deployment Type View**.
3. Review the icons on these views to determine if any test categories are assigned the **Error With Fix** or **Warning With Fix** status icon. In this example, the **Error With Fix** icon is displayed:



4. To automatically resolve issues in the selected package, or in the packages in the selected application or group, click the **Resolve Issues** button (or click F7):

Issue resolution begins, progress messages appear in the Output window, and Application Catalog performs the following tasks:

- **Reruns tests**—Application Catalog reruns all of the selected tests to ensure that the issues that it is going to resolve still exist in the current version of the package and its associated transforms.
- **Creates transform files**—To resolve issues, Application Catalog generates the following fix transform files:

  *PackageName*_AS_Fixed.mst
  *PackageName*_AS_Conflicts.mst

  The file ending in AS_Fixed.mst fixes operating system compatibility issues, while the file ending in AS_Conflicts.mst fixes conflict and Windows best practices and mobile app risk assessment issues.
- **Reimports packages**—Application Catalog then automatically reimports each package and its fix transform files into the Application Catalog.

5. When issue resolution and reimporting is complete, look at the **Analyze Group View**, **Application View**, or **Deployment Type View** of the package, application, or group that you tested. You will see that the Error With Fix and Warning With Fix icons have been replaced with the status icon with the next highest level (as described in Hierarchical Level of Status Icons) in that test category.
6. Open the Application Catalog **Home** tab and select one of the fixed packages in the tree. You will notice that in the **Transforms** property on the **Package Information** tab of the **Home Deployment Type View**, the name of the fix transform is now listed, such as:

```
This product has 2 transforms.
    (C:\Applications\AdobeAcrobat9\AdobeAcrobat9_AS_Fixed.mst
     C:\Applications\AdobeAcrobat9\AdobeAcrobat9_SoftwareId.mst)
```

Issue Resolution and the Software Repository

When you attempt to automatically resolve issues for a package in the Software Repository, Application Catalog will automatically check out the package, create the fix transform, reimport the package and transform, and check the package back in. However, if the package is already checked out by someone else, Application Catalog will be unable to perform the auto fix.

Performing Manual Issue Resolution

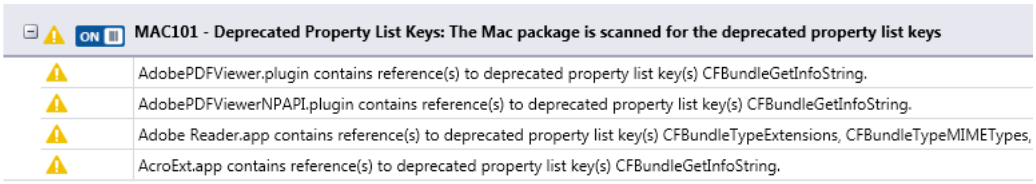
Due to their complexity, some conflicts require manual resolution using InstallShield Editor or the Virtual Package Editor.

To perform manual issue resolution for Windows Installer or App-V packages, perform the following steps.

Task ***To perform manual issue resolution:***

1. Perform testing, as described in Performing Compatibility, Best Practices, and Risk Assessment Testing and Performing Application Conflict Testing.
2. Perform automatic issue resolution, as described in Performing Automatic Issue Resolution.
3. Open the **Analyze** tab, and select a package in the tree. The **Summary** tab of the **Analyze Deployment Type View** opens.
4. Open the tab of the test group that contains an issue that you want to resolve.

5. Locate the test that contains an issue that you want to resolve, and expand it to display the errors or warnings were generated by this test:



   These error/warning messages list information specific to the package that explains why the error or warning was generated, such as the table name and/or the component name that is causing the issue.

6. Review the information in the error or warning message, and consult the **Manual Fix** subsection of this test's topic in About Analyze Tests.

7. If you determine that it is possible to manually resolve this issue, do one of the following:

   - **Windows Installer packages**—Open the package in InstallShield Editor and create a fix transform to resolve this issue.

     Note • *For information on using InstallShield Editor, see Customizing and Authoring Installations Using InstallShield.*

   - **Microsoft App-V packages**—Open the package in the Virtual Package Editor and fix the issue.

8. After resolving the issue, delete the package from the Application Catalog, and then reimport the package (and, for Windows Installer packages, the fix transform file) into the Application Catalog.

9. Perform testing again to confirm that the fix resolved the issue.

Viewing Test Summary Reports on Reports Tab

Edition • *The Application Catalog Reports tab is included with AdminStudio Enterprise Edition.*

AdminStudio uses Microsoft SQL Reporting Services to provide a wide array of summary reports that are available on the **Reports** tab. Many of these reports display Analyze test results. On most reports, you can click categories in the charts to open more detailed reports, and view information at the package and issue-level.

For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab.

Analyze Reference

This section contains information on the Application Catalog views, dialog boxes and wizards that are accessible when the **Analyze** tab is selected in the ribbon. This Application Catalog functionality is used when testing packages for best practices, conflicts, application virtualization compatibility, operating system compatibility, and remote application publishing compatibility.

Note • *For information on the Application Catalog interface, see Application Manager Interface.*

Reference information is organized into the following areas:

Table 16-11 • Conflict Analysis and Resolution Reference Organization

Section	Description
Analyze Views	Views used when performing conflict analysis and resolution using Application Catalog are covered in this section.
Analyze Dialog Boxes	Specific help for dialog box used when performing conflict analysis and resolution is provided in this section.
Analyze Wizards	This section contains a panel-by-panel reference for wizards used to perform conflict analysis and resolution.

Analyze Views

The following views are associated with performing conflict analysis on the Products tab in Application Catalog:

- Analyze Group View
- Analyze Application View
- Analyze Deployment Type View

Analyze Group View

The **Analyze Group View** has two formats: standard list format and a graphical, pie chart view of the same data. You can toggle between the list and chart view by clicking the toggle button at the top right of the page:

On the Chart view, you can make selections from the drop down lists to filter each pie chart by various criteria: operating systems, virtual format, or dependencies type.

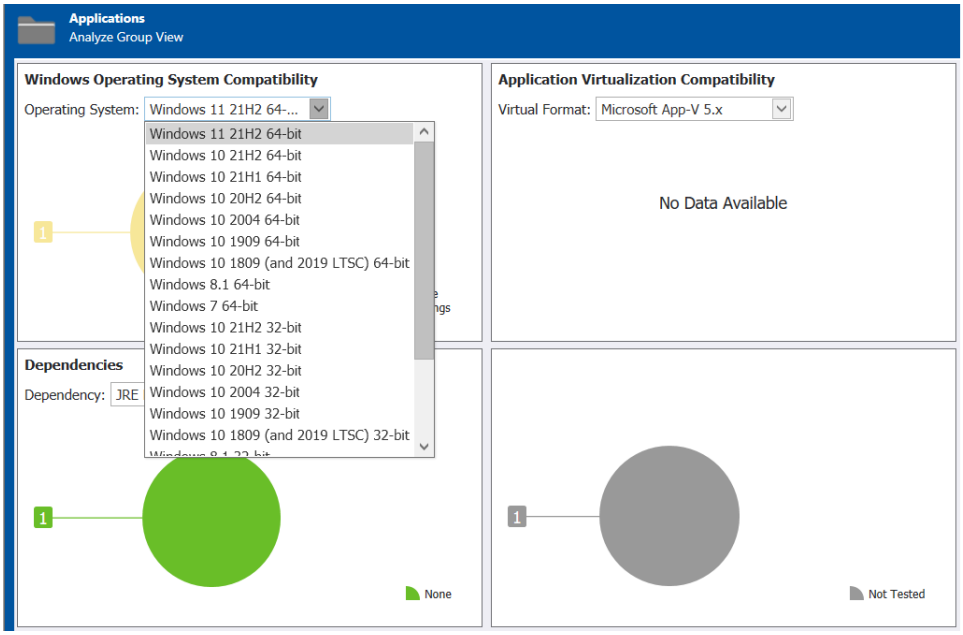


Figure 16-1: Analyze Group View - Pie Chart View

The list format of the **Analyze Group View**, which opens when you select a group in the tree, displays icons to indicate the overall test status of applications and/or subgroups, as described in About Status Icons, in each of the following columns:

- Operating System Compatibility
- MSIX Conversion Capability
- Application Virtualization Compatibility
- Remote Application Publishing Compatibility
- Best Practices
- Risk Assessment
- Application Conflicts

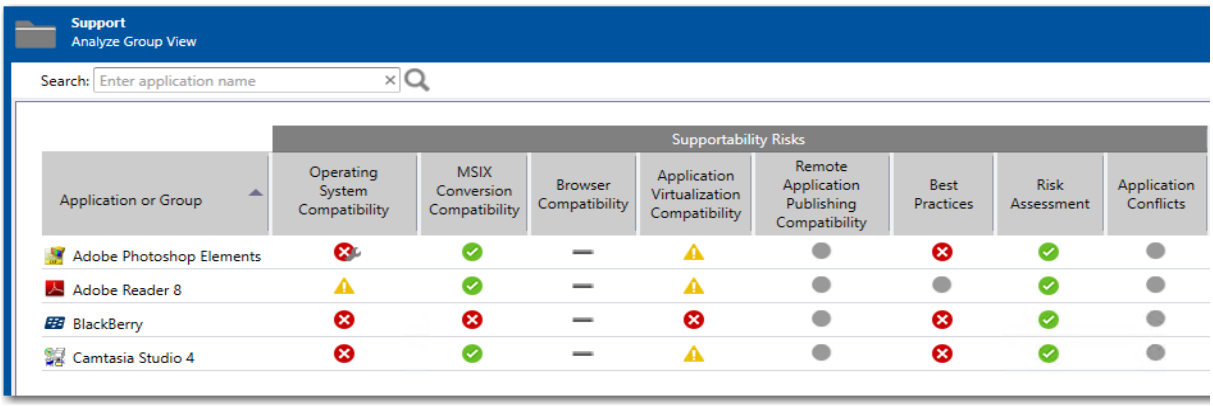


Figure 16-2: Analyze Group View - List View

A status icon is displayed in each of the columns to indicate the test status of the tests in that category for the group, application, or package. All of the columns on this view are sortable by clicking on the column heading.

To view the detailed test results for a single package, select the package in the Application Catalog tree to open the Analyze Deployment Type View.

Analyze Application View

The **Analyze Application View**, which opens when you select an application in the tree, displays icons to indicate the overall test status of an application's packages, as described in About Status Icons, in each of the following columns:

- Operating System Compatibility
- MSIX Conversion Compatibility
- Application Virtualization Compatibility
- Remote Application Publishing Compatibility
- Best Practices
- Risk Assessment
- Application Conflicts

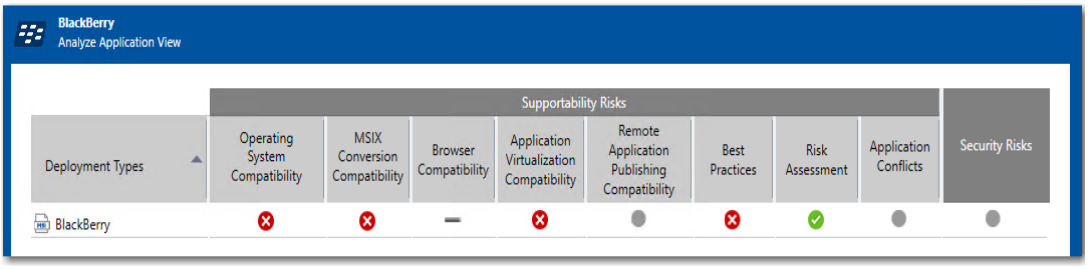


Figure 16-3: Analyze Application View

A status icon is displayed in each of the columns to indicate the test status of the tests in that category for the package. All of the columns on this view are sortable by clicking on the column heading.

To open the Analyze Deployment Type View to see the detailed test results for a single package, double-click the package.

Analyze Deployment Type View

The **Analyze Deployment Type View**, which is displayed when you select a package in the tree, contains detailed test results for individual packages.

Information on the **Analyze Deployment Type View** is displayed on the following tabs:

- Supportability Risks

Supportability Risks

The **Supportability Risks** tab displays a summary of test categories and corresponding overall assessment which are displayed in the following sections:

- Operating System Compatibility
- MSIX Conversion Compatibility
- Application Virtualization Compatibility
- Remote Application Publishing Compatibility
- Risk Assessment
- Best Practices
- Application Conflicts

A status icon, as described in About Status Icons, identifies the package's test status in each of the test groups and test categories.

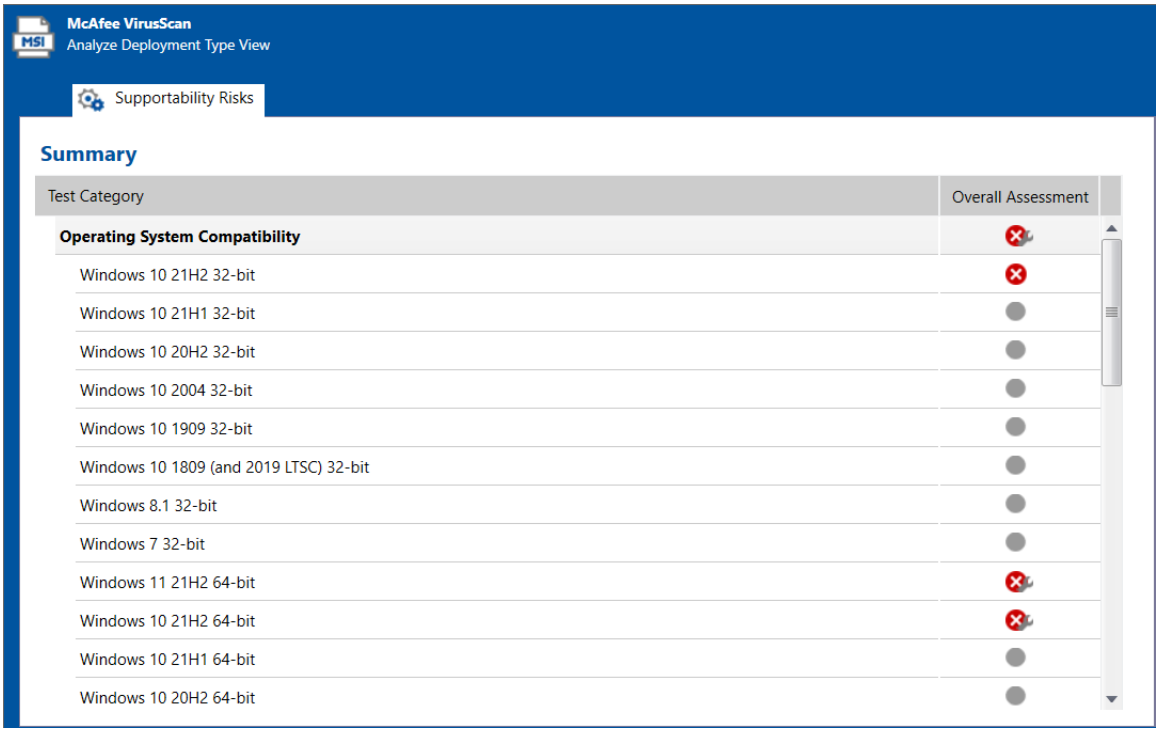


Figure 16-4: Analyze Deployment Type View

When you click on an individual test on the **Supportability Risks** tab, you will see the detailed test results for that test.

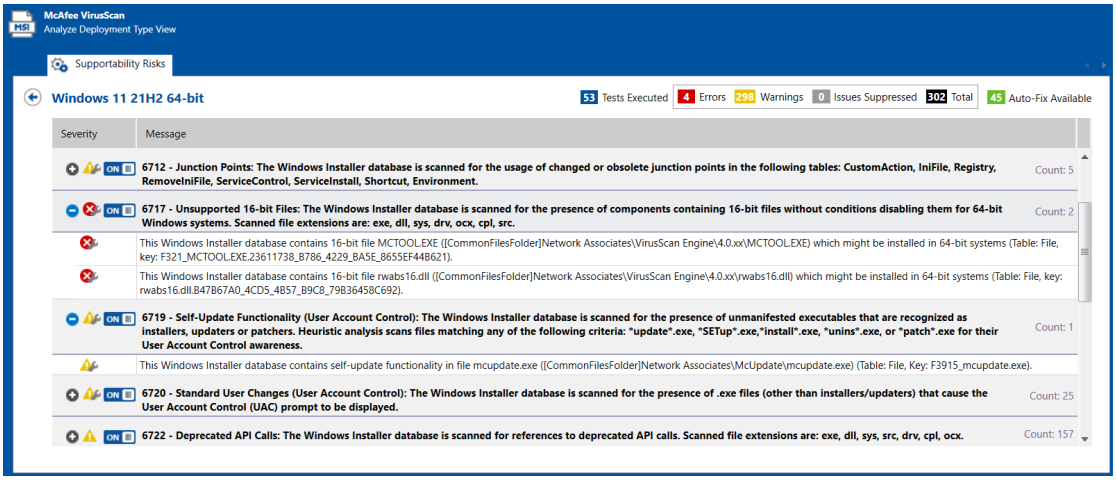


Figure 16-5: Detailed Test Results on Supportability Risks Tab

Details about the test run are listed and the errors/warnings that were generated. Also, the following information is displayed:

Table 16-12 • Supportability Risks Properties

Property	Description
Tests Executed	Number of tests executed in that test group or test category. This number corresponds to the number of tests selected in that test group/category on the **Select Tests to Execute** dialog box.
Errors	Number of non-suppressed errors generated in that test group/category. See Filtering Test Results by Suppressing Errors/Warnings.
Warnings	Number of non-suppressed warnings generated in that test group/category. See Filtering Test Results by Suppressing Errors/Warnings.
Issues Suppressed	Total number of errors and warnings generated in that test category that have been suppressed. See Filtering Test Results by Suppressing Errors/Warnings.
Total	Total number of issues.
Auto Fix Available	Total number of errors and warnings generated in that test category for which an automatic fix is available. See Resolving Issues for more information.
Overall Assessment	Icon indicating the overall test status of the package, as described in About Status Icons.

Operating System Compatibility Tabs

The **Operating System Compatibility** tabs of the Analyze Deployment Type View lists all of the individual errors and warnings that were generated by tests in the **Operating System Compatibility** test group for the package.

On these tabs, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

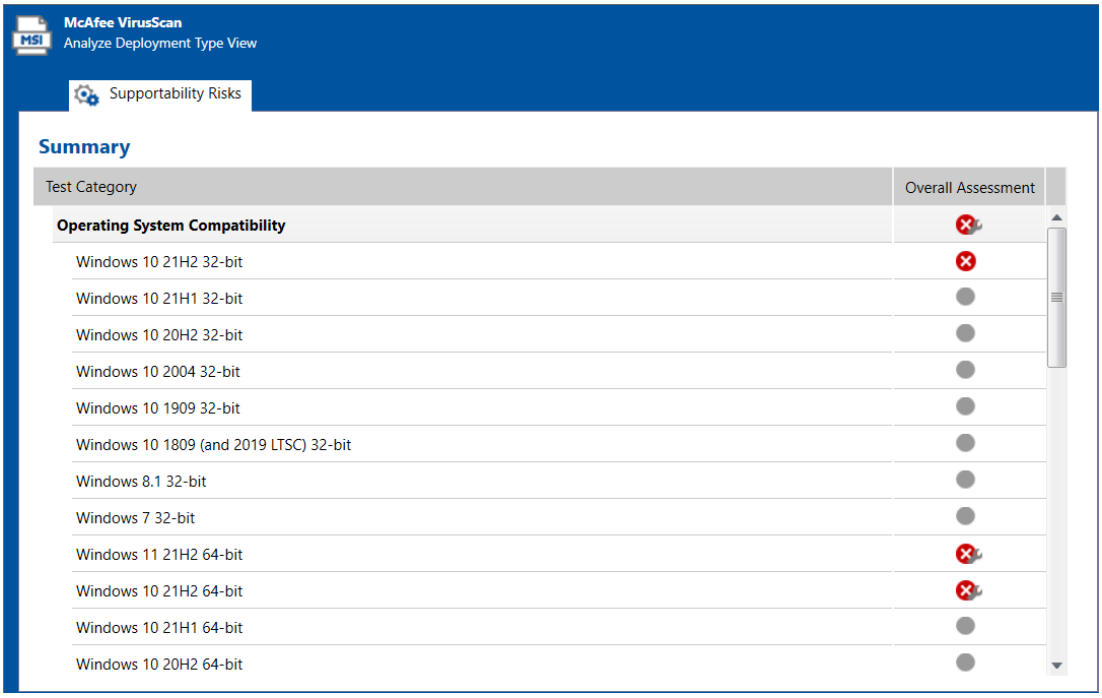


Figure 16-6: Operating System Compatibility Tab / Analyze Deployment Type View

On the **Operating System Compatibility** tabs, the following information is displayed:

Table 16-13 • Operating System CompatibilityTabs

Property	Description
Test Category	Name of test category in the **Operating System Compatibility** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed.
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.

For more detailed information on the issues generated in by **Operating System Compatibility** tests, including information on how to resolve these issues, see Operating System Compatibility Tests .

Tip *• You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the* ***Operating System Compatibility*** *tab and selecting* ***More Information*** *from the shortcut menu.*

Application Virtualization Compatibility Tab

Application Catalog performs application virtualization compatibility testing to determine if a Windows Installer package is a suitable candidate for conversion to Microsoft App-V, Citrix XenApp, VMware ThinApp.

The **Application Virtualization Compatibility** section on the **Supportability Risks** tab of the Analyze Deployment Type View lists all of the individual errors, warnings, and informational messages that were generated when application virtualization compatibility testing was performed.

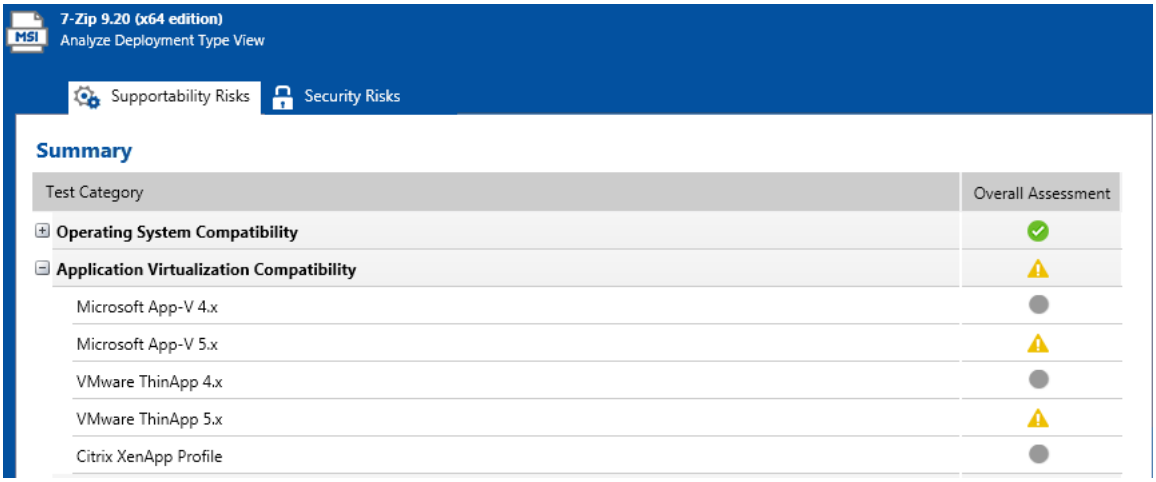


Figure 16-7: Application Virtualization Compatibility Tab / Analyze Deployment Type View

Note • *The Application Virtualization Compatibility tests are always run any time that you run tests in Analyze. However, the selections you make on the* ***Select Tests to Execute*** *dialog box determine which virtual formats to display in test results.*

The **Application Virtualization Compatibility** tab displays the following information:

Table 16-14 • Application Virtualization Compatibility Tab

Property	Description
Test Category	Identifies the test category as one of the following virtualization technologies: • VMware ThinApp • Microsoft App-V • Citrix XenApp When each test category is expanded, the tests in that category that generated errors or warnings are listed and explains why the error or warning was generated.

Table 16-14 • Application Virtualization Compatibility Tab (cont.)

Property	Description
Icon	For each issue, one of the following icons is displayed: • Informational • Error • Warning The Application Virtualization Compatibility tab includes one additional status icon that is not used in the other test groups called the informational icon: This icon identifies issues that require that Automated Application Converter will need to automatically repackage this Windows Installer package during the conversion process to a virtual package. Since these issues by themselves do not necessarily indicate a warning or error, they are considered “informational” issues. For more information, see About Status Icons and Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter.

For more detailed information on the issues generated in by **Application Virtualization Compatibility** tests, including information on how to resolve these issues, see Application Virtualization Compatibility Tests.

Note • *See also Application Virtualization Compatibility Status: Analyze vs. Automated Application Converter.*

Best Practices Tab and Risk Assessment Tab

The **Best Practices** tab and the **Risk Assessment** tab of the Analyze Deployment Type View list all of the individual errors and warnings that were generated by tests in the **Best Practices** test group and **Risk Assessment** test group for the package.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

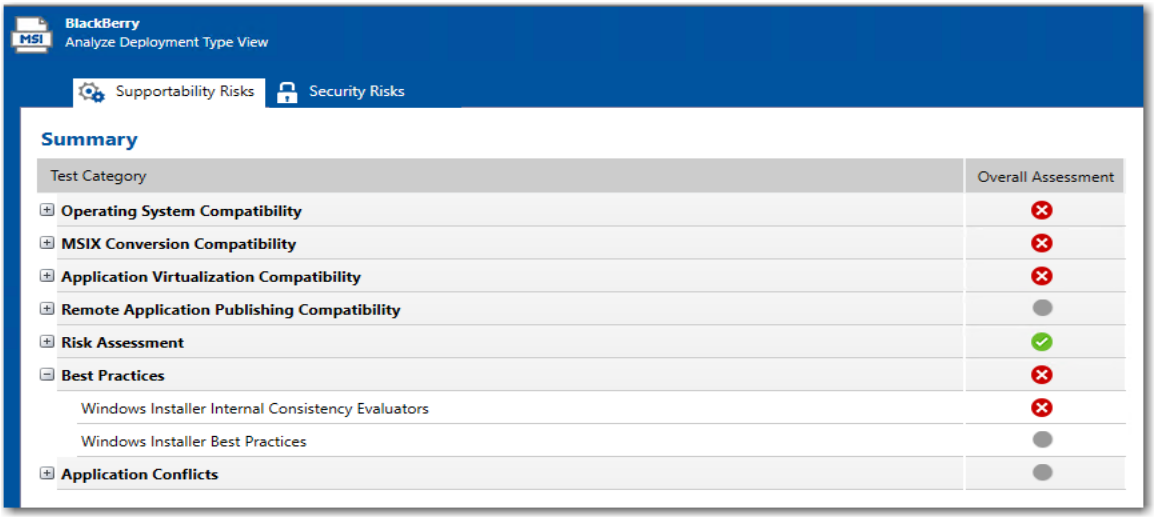


Figure 16-8: Best Practices Tab / Analyze Deployment Type View

The **Best Practices** tab and the **Risk Assessment** tab display the following information:

Table 16-15 • Best Practices Tab and Risk Assessment Tab

Property	Description
Test Category	Name of test category in the **Best Practices** or **Risk Assessment** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed and explains why the error or warning was generated.
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.

For more detailed information on the issues generated by **Best Practices** tests and **Risk Assessment** tests, including information on how to resolve these issues, see Best Practices Tests and Risk Assessment Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the **Best Practices** tab or **Risk Assessment** tab and selecting **More Information** from the shortcut menu.*

Application Conflicts Tab

The **Application Conflicts** tab of the Analyze Deployment Type View lists all of the individual errors and warnings that were generated by tests in the **Application Conflicts** test group for the package when conflict analysis was performed.

On this tab, you can read the detailed error and warning messages, and can choose to suppress any errors or warnings that you feel are not important at your organization, as described in Filtering Test Results by Suppressing Errors/Warnings.

Figure 16-9: Application Conflicts Tab / Analyze Deployment Type View

On the **Application Conflicts** tab, the following information is displayed:

Table 16-16 • Application Conflicts Tab

Property	Description
Test Category	Name of test category in the **Application Conflicts** test group for which errors or warnings were generated. When this test category is expanded, the tests in that category that generated errors or warnings are listed and explains why the error or warning was generated.
Icon	For each error or warning, one of the following icons is displayed: • Error • Error With Fix • Warning • Warning With Fix For more information, see About Status Icons.

For more detailed information on the issues generated by **Application Conflicts** tests, including information on how to resolve these issues, see Application Conflicts Tests.

Tip • *You can also quickly access detailed test information directly from the Analyze interface by right-clicking on the test on the* ***Application Conflicts*** *tab and selecting* ***More Information*** *from the shortcut menu.*

Analyze Subnode Views

The following views are opened by expanding a Windows Installer package node in the tree and selecting a subnode:

- Patch Impact View
- Associated Patches View

Patch Impact View

The **Patch Impact View** is displayed when you select the **Patch Impacts** node under a package in the tree on the **Analyze** tab.

Summary View

The information displayed on the **Patch Impact View** is dependent upon the selection made in the **Impact category** list. The **Summary View**, which is displayed when **Summary** is selected from the **Impact category list**, displays a list of patches for which there is patch impact data persisted against the product. The following information is displayed:

Table 16-17 • Summary View Information

Option	Description
Id	Number identifying this patch's associated Microsoft Security Bulletin.
Name	Name of the patch file.
Title	Title of the patch.
Release Date	Date this patch was published by Microsoft.

File Impacts View

The **File Impacts View**, which is displayed when **File Impacts** is selected from the **Impact category** list and you have file impacts persisted, lists all impacts against this product or OS Snapshot and identifies the patch that caused the impact. If you double-click on one of the patches, the Patch View for that patch will open. The following information is displayed:

Table 16-18 • File Impacts View Information

Option	Description
Description	Description of the impact.
Id	Number identifying this patch's associated Microsoft Security Bulletin.

Table 16-18 • File Impacts View Information

Option	Description
Name	Name of the patch file.
Title	Title of the patch.

Associated Patches View

On the Associated Patches View, you can view a list of imported patches that, if installed, would update the selected product. Application Catalog examines the patches in the catalog and attempts to identify those patches which will impact this package.

Due to differences in the way versions are compared, it is possible that other patches that impact this package may exist. For more definitive information, open the **Patch Properties** dialog box and compare the product and OS snapshot version information of the patch against the specific product and version information.

In the Associated Patches View, the following information is displayed:

Table 16-19 • Associated Patches View Information

Option	Description
Name	Name of the patch that is associated with this product.
Information	Microsoft Security Bulletin identification number and description of the patch.

If you double-click on a patch in the **Associated Patches View**, the **Patch View** (on the **Environment** tab) for that patch opens, listing general information on the selected patch.

Analyze Dialog Boxes

The following dialog boxes are accessible from Application Catalog:

- About Application Catalog Dialog Box
- ACE Rule Properties Dialog Box
- Add Ignore Table Dialog Box
- Expression Builder Dialog Box
- Rules Viewer Dialog Box

About Application Catalog Dialog Box

The About Application Manager dialog box can be accessed by selecting **About Application Manager** on the **Support** tab of the Application Manager ribbon. This dialog box displays information about the product, including the full version number (essential if you need technical support).

To upgrade your edition of AdminStudio, click the **Upgrade** button. For more information, see Upgrading Your Product Edition.

ACE Rule Properties Dialog Box

The ACE Rule Properties dialog box allows you to edit an existing user-defined ACE rule. You can display the dialog by clicking **Edit** on the on the **Rules Viewer** dialog box.

The following tabs are part of the ACE Rule Properties dialog box:

- General Information Tab
- Additional Information Tab
- Custom Options Tab
- Where Clause Tab
- DLL Information Tab

Note • *You can only edit user-defined ACE rules; you are not permitted to edit the ACE rules that were installed with Application Catalog.*

General Information Tab

From the **General Information** tab, you can configure information about the new ACE rule.

This information is used primarily for display information (Name, Brief Description, Description, and Information URL).

Table 16-20 • General Information Tab Information

Option	Description
Name	The name of the ACE, used to organize the rule in Application Catalog. This is displayed in several places, including the **Output Window**, the **Rules Viewer** dialog box, and the **ACE Tests** tab of the **Options** dialog box.
Associated Table	Select the table in the Application Catalog which will be queried in the user-defined ACE. This also determines which columns are available in the **Expression Builder** dialog box, and which tokens are available for the **Error** and **Display** strings on the **Custom Options** panel of the **Rules Wizard**.
Package Type	Select **MSI** or **App-V** to identify the type of package that this rule will be run on.
Brief Description	Enter a brief description which will be displayed in the **Rules Viewer** dialog box, the **ACE Tests** tab of the **Options** dialog box, and in the **Output Window**. This description should be clear enough so users can understand when to use this ACE.
Description	Enter a description of the ACE, which is displayed at the bottom of the Rules Viewer dialog box when the ACE is selected and in the **Output Window** during conflict identification when the ACE executes.

Table 16-20 • General Information Tab Information

Option	Description
Information URL	Provide a URL to get further information for the ACE.

To continue editing **ACE Properties**, click the **Additional Information**, **Custom Options**, **Where Clause**, or **DLL Information** tabs. To save your edits and close this dialog box, click OK.

Additional Information Tab

From the **Additional Information** tab, you can edit information for categorizing the ACE in relation to other ACEs in Application Catalog.

Table 16-21 • Additional Information Tab Information

Option	Description
Category	Either select an existing category for this new rule, or enter the name for a new category. These categories are displayed in the Conflict View and the **ACE Tests** tab of the **Options** dialog box. Ideally, any user-defined ACEs should be put in their own category.
Rule Type	The Rule Type of this ACE is displayed (read only). • **Custom - Source Only Packages ACEs** allow you to quickly test any column or any value of a table. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a user-defined ACE, you use an SQL “Where” clause. • **Custom - Source and Target Packages ACEs** allow you to compare columns or values of Source package tables (new packages that you want to install onto a user’s system) to columns or values of Target package tables (packages already installed on a user’s system). For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would produce duplicate registry entries. To define a Source and Target Packages ACE, you must define an SQL “Where” clause, and specify a Join Column (a table column in the Application Catalog database that has a matching value for both the Source and Target packages). • **DLL - User Provided DLL based ACEs** allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and other programming commands. With DLL-Based ACEs, you can use a Conflict Application Resolution Definitions (CARDs) to fix the conflict.

To continue editing ACE Properties, click the General Information, Custom Options, Where Clause, or DLL Information tabs. To save your edits and close this dialog box, click OK.

Custom Options Tab

From the Custom Options tab, you can edit this ACE’s display strings for the Output window and Conflict Details.

The following options are included:

Table 16-22 • Custom Options Tab Properties

Option	Description
Error String	This string appears in the Output window when a violation of this ACE rule is detected during conflict identification. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter the following in this field: `Failure in creating desktop icon` ***Note*** • *Tokens allow you to insert values at run-time from the installation package into the string, such as specifying a file name. To use token replacement in the error string, use the arrow to the right of the Error String field. For more information, see Token Grammar.*
Display String	This string appears on the **Application Conflicts** tab of the **Analyze Deployment Type View** after conflicts have been identified. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter the following in this field: `Duplicate desktop icon found` ***Note*** • *Tokens allow you to insert values at run-time from the installation package into the string, such as specifying a file name. To use token replacement in the error string, use the arrow to the right of the Display String field. For more information, see Token Grammar.*
Severity	Specify whether this ACE should be an Error or a Warning.
Report 'No' results	User-defined ACEs report conflicts based on the provided query. However, you may want to report the absence of the data if it could not be found. If you select this option, if the ACE does not return any results, it will be reported as an error (or warning), with the description and error strings as specified. If you expect a **No** result, do not use tokens in your display or error strings.

To continue editing ACE Properties, click the General Information, Additional Information, or Where Clause tabs. To save your edits and close this dialog box, click OK.

Where Clause Tab

From the Where Clause tab, you can edit the Where clause for the ACE. If you do not know how to build a Where clause, you can click the Build Expressions button to launch the **Expression Builder** dialog box. You can also click Test to validate the Where clause syntax.

If you selected Custom - Source and Target Packages when you created this ACE, you must have also selected a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages. For example, if you wanted to evaluate records from two tables that have a installation directory of `C:\ProgramFiles`, then you would specify Directory as the Join column. To change the Join Column, select a different column name from the list.

To continue editing ACE Properties, click the General Information, Additional Information, or Custom Options tabs. To save your edits and close this dialog box, click OK.

Tip • *To improve query performance, enclose table names in square brackets ([]).*

DLL Information Tab

From the **DLL Information** tab, you can edit specific information about the ACE/CARD DLL file and the ACE and CARD Function Names that DLL-based ACEs require to operate.

The following options are included:

Table 16-23 • DLL Information Tab Information

Option	Description
ACE/CARD DLL File	Select the name of the ACE DLL that you are testing.
ACE Function Name	Enter the name that you chose to “export” for this ACE function.
CARD Function Name	Enter the name that you chose to “export” for this CARD function.
Test	Click the Test button next to the ACE Function Name or CARD Function name to validate that the exported function does exist.

To continue editing ACE Properties, click the General Information or Additional Information tabs. To save your edits and close this dialog box, click OK.

Add Ignore Table Dialog Box

This dialog box allows you to specify a custom table to ignore during import of a package into the Application Catalog. You can also provide comments about the table.

Expression Builder Dialog Box

The Expression Builder dialog box, available by clicking Build Expression on the **Where Clause** panel in the Rules Wizard (when creating a new user-defined ACE) or from the Where Clause Tab of the ACE Rule Properties Dialog Box (when editing a user-defined ACE), allows you to build simple Where clause expressions for Application Catalog user-defined ACEs.

Set values for the following options:

Table 16-24 • Expression Builder Dialog Box Properties

Option	Description
Table Columns	This list is populated from the table columns in the table defined in the Rules Wizard General Information panel. Select the table column used in this Where clause.

Table 16-24 • Expression Builder Dialog Box Properties (cont.)

Option	Description
Comparison Operator	Pick an operator to use for comparison in the Where clause. You can pick from the following: • = (Equal To) • <> (Not Equal To) • > (Greater Than) • < (Less Than) • >= (Greater Than or Equal To) • <= (Less Than or Equal To)
Constant	This constant can be a numerical value or string value. The property label will change based on the expected constant type. This value is compared against the data in the specified table. ***Note*** • *When using the Expression Builder dialog box to create a Source and Target Packages custom ACE to compare the value of a column in the source table to the value of a column in the target table, you can select the first table column name from the Table Columns list. However, you have to manually enter the second table column name in the Constant text box. When doing so, enter the table column name using the same syntax that is used in the Table Columns list: [Source].[ColumnName] or [Target].[ColumnName].* *See Creating a Custom/Source and Target Packages ACE.*
Expression Operator	If there is more than one expression in the Where clause, you can specify an operator to join the current expression to the previous expression.

Tip • *When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the Where Clause text box on the Where Clause panel of the Rules Wizard or on the Where Clause tab of the ACE Rule Properties Dialog Box.*

Rules Viewer Dialog Box

The **Rules Viewer** dialog box, accessible by clicking **View Rules** on the **ACT Tests** tab of the Application Catalog **Options** dialog box, allows you to view the current categorization of ACEs used for conflict identification. More importantly, it allows you to access the Rules Wizard to include user-defined ACEs in Application Catalog.

The primary window in the **Rules Viewer** dialog box displays a tree view containing each available ACE, grouped by category. If you click **New**, the **Rules Wizard** launches, allowing you to configure information for a new ACE.

When a user-defined ACE is selected on the **Rules Viewer** dialog box, the **Edit** button is enabled. When you click **Edit**, the ACE Rule Properties Dialog Box appears, where you can reconfigure the ACE. You can also delete user-defined ACEs by selecting them and clicking **Delete**.

Select Tests to Execute Dialog Box

On the **Select Tests to Execute** dialog box, which is opened by clicking the **Select Tests to Execute** button in the ribbon on the **Analyze** tab, you can select the tests that you want to execute when the **Execute Tests** button is clicked.

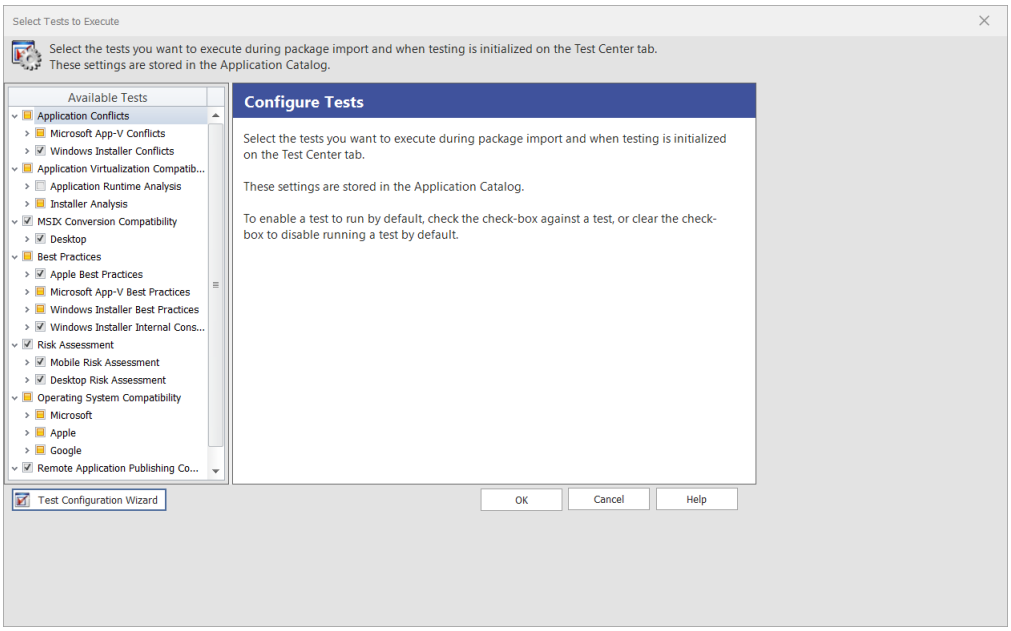


Figure 16-10: Select Tests to Execute Dialog Box

For more information on these tests, see the following sections:

- Operating System Compatibility Tests
- Application Virtualization Compatibility Tests
- Best Practices Tests
- Risk Assessment Tests
- Application Conflicts Tests
- Remote Application Publishing Compatibility Tests

Analyze Wizards

The following wizards related to conflict analysis and resolution are included in Application Catalog:

- AdminStudio Test Configuration Wizard
- Conflict Wizard
- Rules Wizard

AdminStudio Test Configuration Wizard

Instead of selecting individual Operating System Compatibility tests to run on the **Select Tests to Execute** dialog box, you have the option of using the **AdminStudio Test Configuration Wizard** to identify the tests to run by selecting one of three compliance levels, which are based on industry standard compliance rule sets:

- **Complete Analysis**—Test applications for all potential Operating System Compatibility issues.
- **Industry Standard Analysis**—Test for the Operating System Compatibility issues that would cause an application to fail.
- **Industry Standard Analysis With Auto-Fixes**—Only test applications for potential Operating System Compatibility issues for which an automatic fix is available.

Using the Test Configuration Wizard, you can also further refine the tests that are run by specifying an OS Snapshot to test against.

You open the **AdminStudio Test Configuration Wizard** by clicking **Test Configuration Wizard** on the **Select Tests to Execute** dialog box.

The AdminStudio Test Configuration Wizard consists of the following panels:

- Compliance Level Panel
- OS Snapshot(s) Panel
- Summary Panel

Compliance Level Panel

On the **Compliance Level** panel of the AdminStudio Test Configuration Wizard, which is opened by clicking **Test Configuration Wizard** on the **Select Tests to Execute** dialog box, you can identify the Analyze tests to run by selecting one of three compliance levels, which are based on industry standard compliance rule sets:

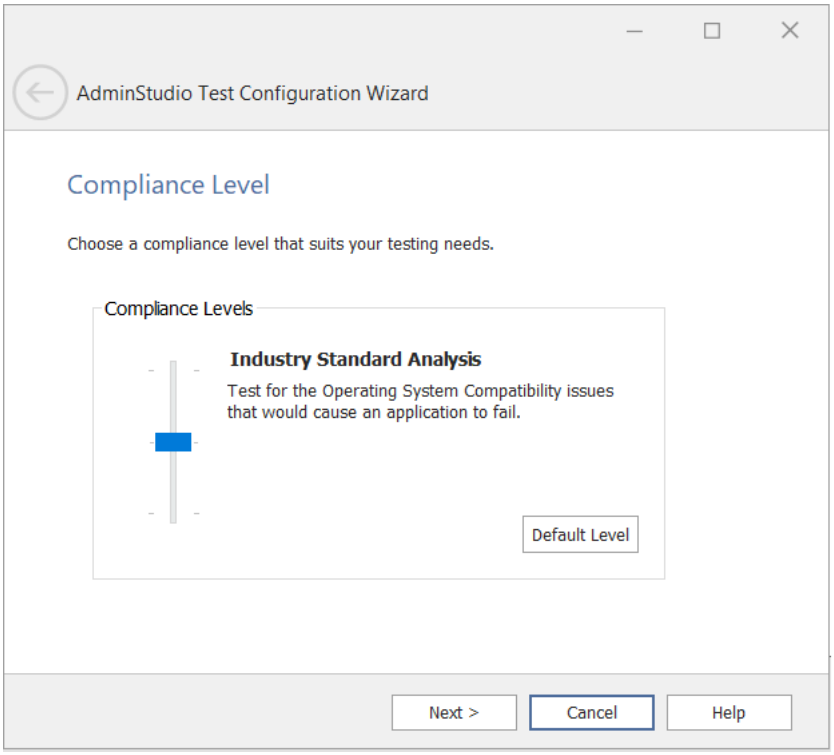


Figure 16-11: Choose a Compliance Level Panel

The Choose a Compliance Level panel has the following options:

Table 16-25 • Choose a Compliance Level Panel

Option	Description
Complete Analysis	Select this option to test applications for all potential Operating System Compatibility issues.
Industry Standard Analysis	Select this option to test for the Operating System Compatibility issues that would cause an application to fail.
Industry Standard Analysis With Auto-Fixes	Select this option to only test applications for potential Operating System Compatibility issues for which an automatic fix is available.
Default Level	Click to reset the slider to In**dustry Standard Analysis**, the default compliance level.

Important • *The **Compliance Level** selection you make on this panel does not affect the selection of tests in the Application Conflicts, Application Virtualization Compatibility, Best Practices, Risk Assessment, or Remote Application Publishing Compatibility test categories.*

OS Snapshot(s) Panel

The **OS Snapshot(s)** panel of the AdminStudio Test Configuration Wizard, which is opened by clicking **Test Configuration Wizard** on the **Select Tests to Execute** dialog box, lists all of the OS Snapshots you have imported into the Application Catalog and prompts you to select one or more to filter the test selection.

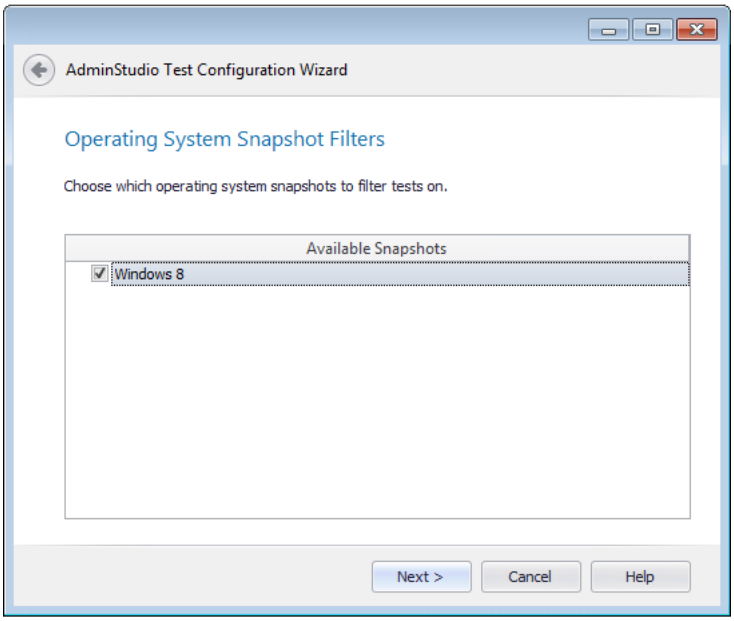


Figure 16-12: OS Snapshot(s) Panel

Note • *For more information, see Taking OS Snapshots and Importing OS Snapshots.*

If desired, select an OS Snapshot to test against. When you select an OS Snapshot to use to filter the test selection, the following items are considered:

- Operating system version
- Operating system patches applied
- Internet Explorer version installed
- .NET framework version installed

In addition to the level selected on the **Compliance Level** panel, the selection of Operating System Compatibility tests will be further filtered if you select an OS Snapshot on the **OS Snapshot(s)** panel. The only Operating System Compatibility test categories that will have any selected tests will be the categories corresponding to the selected operating systems. For example, if you choose a Windows 8.1 64-bit OS Snapshot, Operating System Compatibility tests will be selected only in the **Windows 8.1 64-bit** test category, will be selected only in the **Internet Explorer 11** test category:

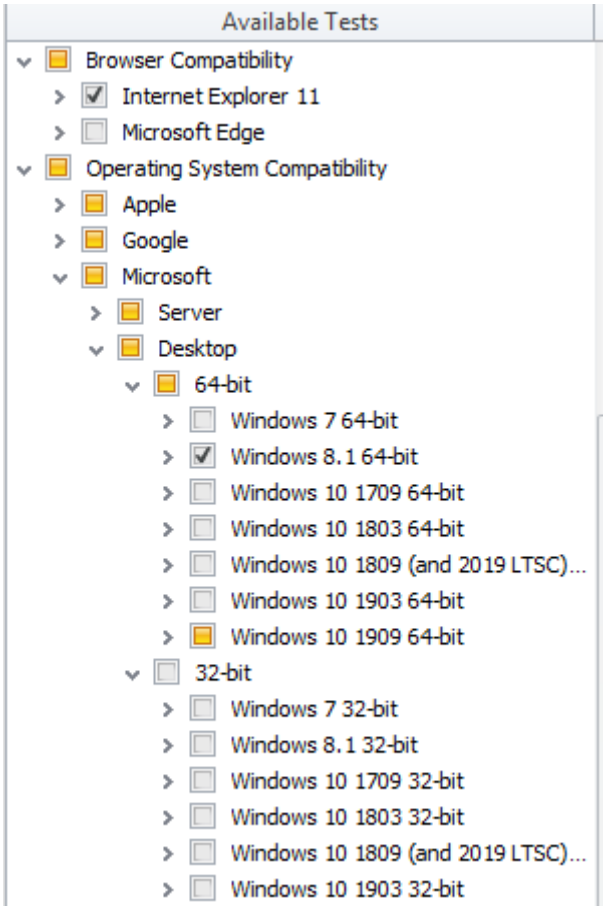


Figure 16-13: Operating System Compatibility Tests Filtered by Windows 8.1 64-bit

Summary Panel

The **Summary** panel of the AdminStudio Test Configuration Wizard, which is opened by clicking **Test Configuration Wizard** on the **Select Tests to Execute** dialog box, lists a summary of the selections you have made in the wizard.

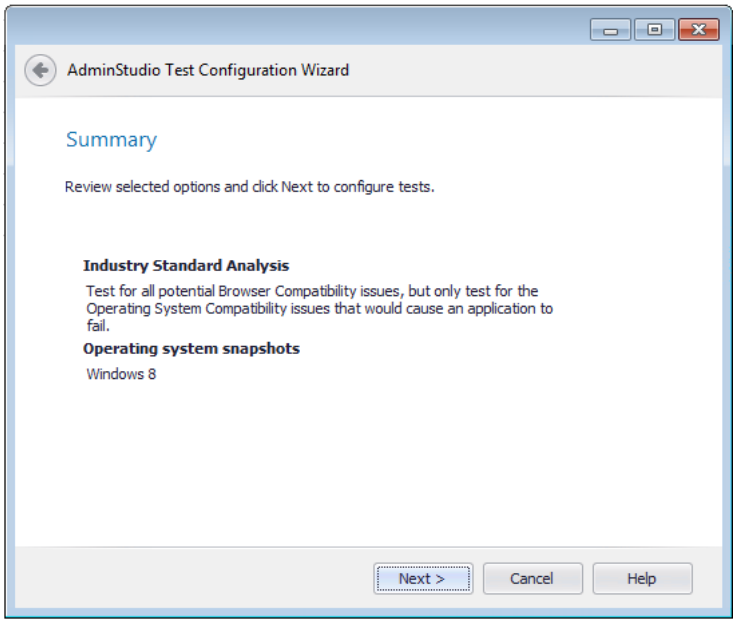
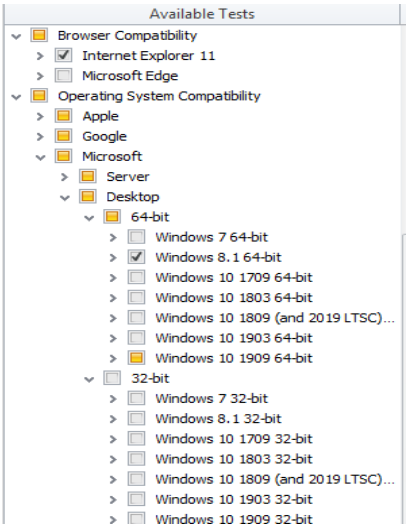


Figure 16-14: Summary Panel

Click **Next** to apply the selected settings. A message appears stating that the test configuration has been updated.

Click **Finish** to close the wizard. You will then be able to notice the following changes that were made in the **Available Tests** list:

Table 16-26 • Results of Using the AdminStudio Test Configuration Wizard

Test Category	Change
Operating System Compatibility	The Operating System Compatibility tests that are selected depend upon the level you chose on the **Choose a Compliance Level** panel. The selection of Operating System Compatibility tests will be further filtered if you selected an OS Snapshot on the **OS Snapshot(s)** panel. The only Operating System test categories that will have any selected tests will be the categories of the selected operating systems. For example, if you choose a Windows 8 64-bit OS Snapshot, tests will be selected only in the **Windows 8 64-bit** test category: 
Application Conflicts **Application Virtualization Compatibility** **Best Practices** **Risk Assessment** **Remote Application Publishing Compatibility**	Test selection in these test categories are not affected by any selections made in the **AdminStudio Test Configuration Wizard**.

Conflict Wizard

Although a Windows Installer package or merge module may be built to guidelines put forth by Microsoft, it is possible that the interaction between packages, or between a package and the base operating system, may cause unwanted results in your production environment. You can use the Conflict Wizard to identify these conflicts before you deploy packages, and resolve the problems before they affect your end users.

The Conflict Wizard allows you to identify conflicts between a Windows Installer package and packages already imported into the Application Catalog. You can check for a variety of conflict types, including file, component, and registry conflicts. In many cases, Application Catalog can resolve the issues automatically. You can also create your own custom rules to ensure packages conform to your internal standards and practices. Application Catalog has rules to detect conflicts involving: Components, Files, Registry Entries, Shortcuts, INI Files, ODBC Resources, NT Services, File Extensions, and Product Properties.

The Conflict Wizard consists of the following panels:

- Target Information Panel
- Target OS Snapshot Information Panel
- Summary Panel

When run, Application Catalog displays the output report in the Conflicts tab of the Output Window.

Target Information Panel

In the Target Information panel, select the individual packages or groups of packages in Application Catalog that you want to compare the source package(s) against.

Each package selected will be compared against the packages you specified in the Source Package panel (for internal comparisons) or MSI Source Information panel (for comparisons with an external Windows Installer package).

Note • *The Target Information panel excludes all packages that you selected on the Source Package panel. Empty groups are also excluded.*

You can also select all packages in the Application Catalog or clear all selected packages using the Select All and Clear All buttons.

Target OS Snapshot Information Panel

When you launch the Conflict Wizard from the **Environment** tab with an OS Snapshot selected in the tree, the **Target Snapshot Information** panel opens, prompting you to select the OS Snapshots against which you want to compare the source package in conflict analysis.

Select the snapshots and click **Next** to continue.

Summary Panel

The Summary panel provides a detailed summary of the options that were selected in the previous panels of the Wizard.

Click Finish to run the Conflict Wizard using the options specified.

Mobile Test Wizard

AdminStudio's mobile risk assessment tests enable you to find out which features a specific mobile app uses, such as telephone, location services, camera, microphone, etc. You can enhance this testing by using the **Mobile Test Wizard** to create custom tests that combine risk assessment checks with AND or OR operators. For example, you could create a custom test to see if a mobile application uses a gyroscope OR accelerometer. Or you could create a test that determines whether a mobile application uses location services AND allows location tracking.

The **Mobile Test Wizard** is opened by clicking **New** on the **Mobile Tests** tab of the Application Catalog **Options** dialog box.

The following panels are part of the Mobile Test Wizard:

- Select the Tests Panel
- Provide the Test Details Panel
- Summary Panel

Select the Tests Panel

On the **Select the Tests** panel of the Mobile Test Wizard, you select the tests and connecting operators that will define your custom mobile test. The Mobile Test Wizard is opened by selecting **New** or **Edit** on the **Mobile Tests** tab of the Application Catalog **Options** dialog box.

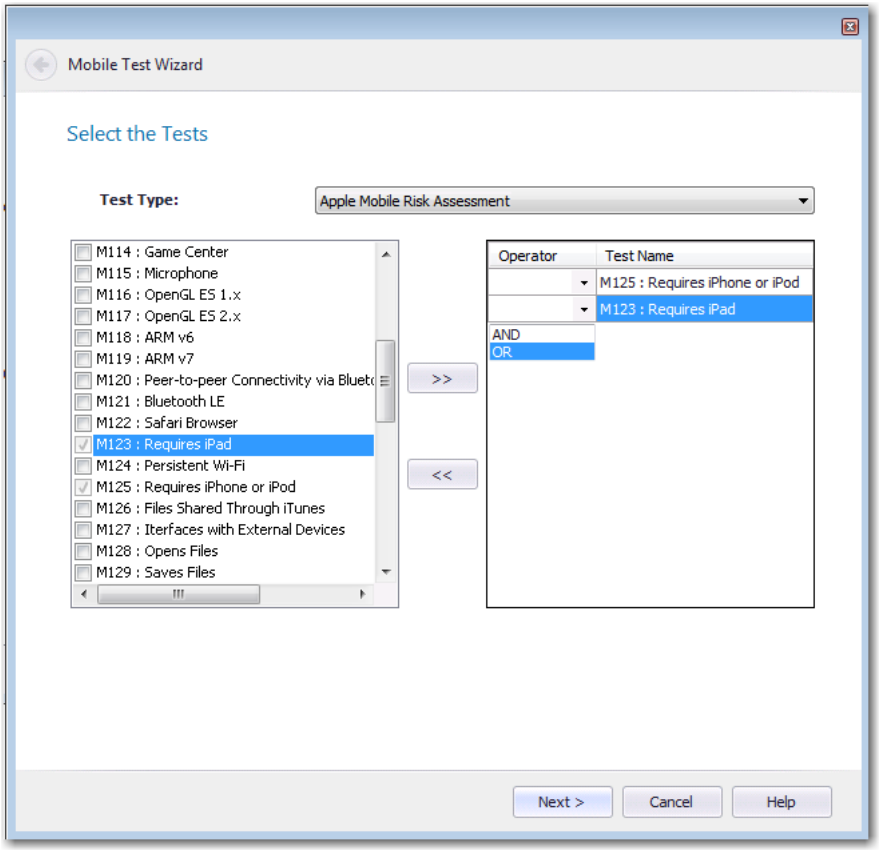
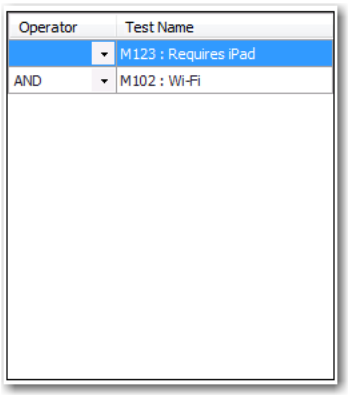


Figure 16-15: Select the Test Panel / Mobile Tests Wizard

The **Select the Tests** panel includes the following properties:

Table 16-27 • Select the Tests Panel Properties

Property	Description
Test Type	Select a mobile test type from the list (Android, Apple, etc.). The available tests in that category are then listed in the box on the left.
Available tests list	To configure a custom mobile test, use the arrow buttons to move tests you want to include in the test from the box on the left to the box on the right.
Custom mobile test list	After adding tests to this list, then join the tests using AND or OR operators by making selections from the **Operator** drop down list. For example, the following test would test a iOS mobile app to see if it requires iPad and WiFi. 
Next	Click to continue to the next panel of the wizard.

Provide the Test Details Panel

On the **Provide the Test Details** panel of the Mobile Test Wizard, you are prompted to give the test a name, description, and a link to more information.

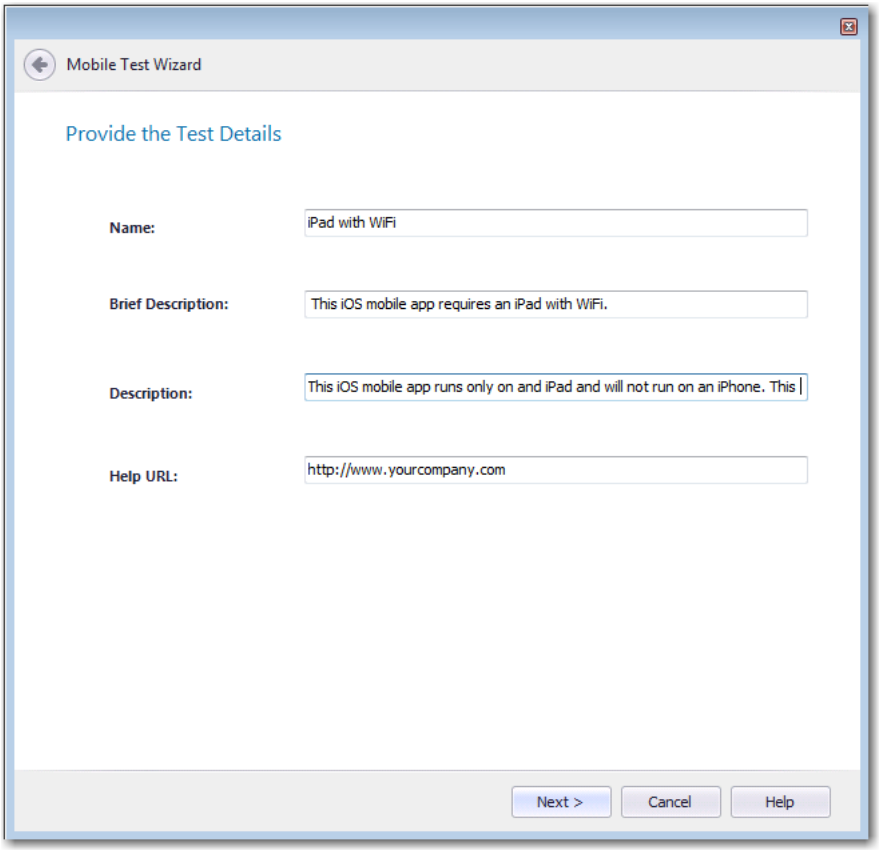


Figure 16-16: Provide the Test Details Panel / Mobile Test Wizard

After the custom mobile test is added, the text you entered in the **Name** field will be displayed on the **Mobile Tests** tab of the **Options** dialog box.

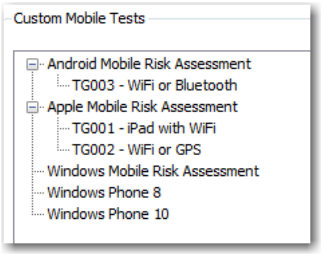


Figure 16-17: Custom Mobile Tests on Mobile Tests Tab of Options Dialog Box

The custom mobile test name is also listed on the **Select Tests to Execute** dialog box.

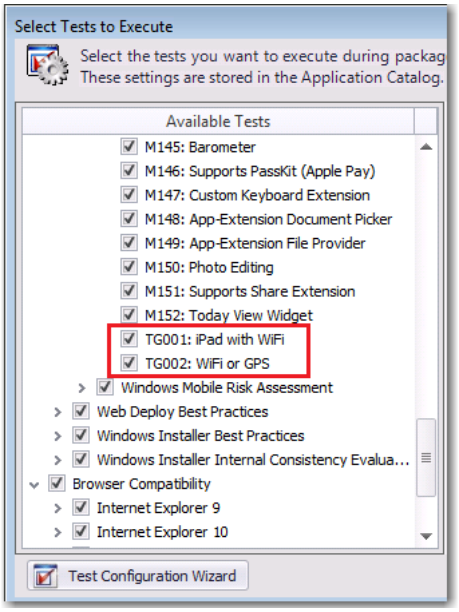


Figure 16-18: Custom Mobile Tests on Select Tests to Execute Dialog Box

The information you entered in the **Brief Description**, **Description**, and **Help URL** fields of the **Provide the Test Details** panel will be displayed in the panel on the right that opens when the test is selected in the **Select Tests to Execute** dialog box.

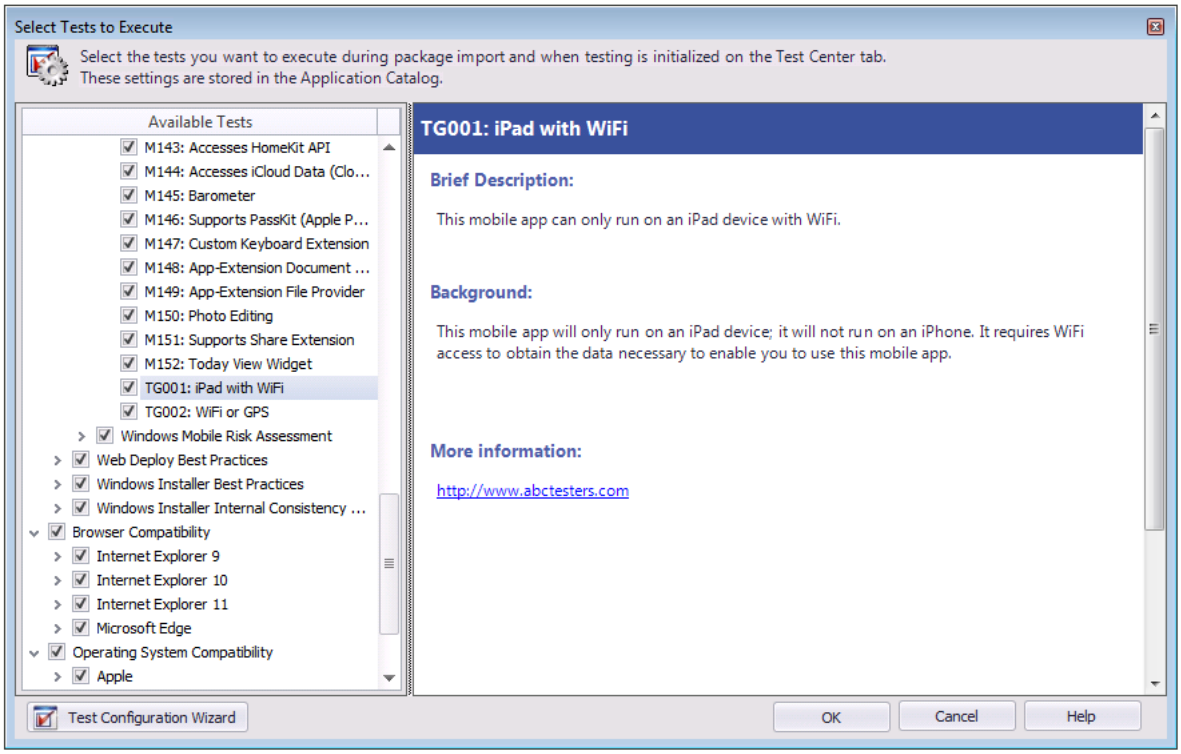


Figure 16-19: Select Tests to Execute Dialog Box

When this issue is detected during the testing of a mobile app, the custom mobile test name is listed on the **Best Practices** tab:

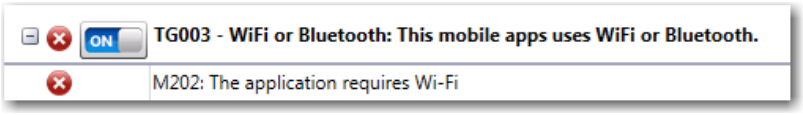


Figure 16-20: Custom Test Displayed on Best Practices Tab of Analyze

Summary Panel

When using the Mobile Test Wizard to create custom mobile tests, the **Summary** panel appears after you have entered all necessary information and you click **Next** on the **Provide the Test Details** panel.

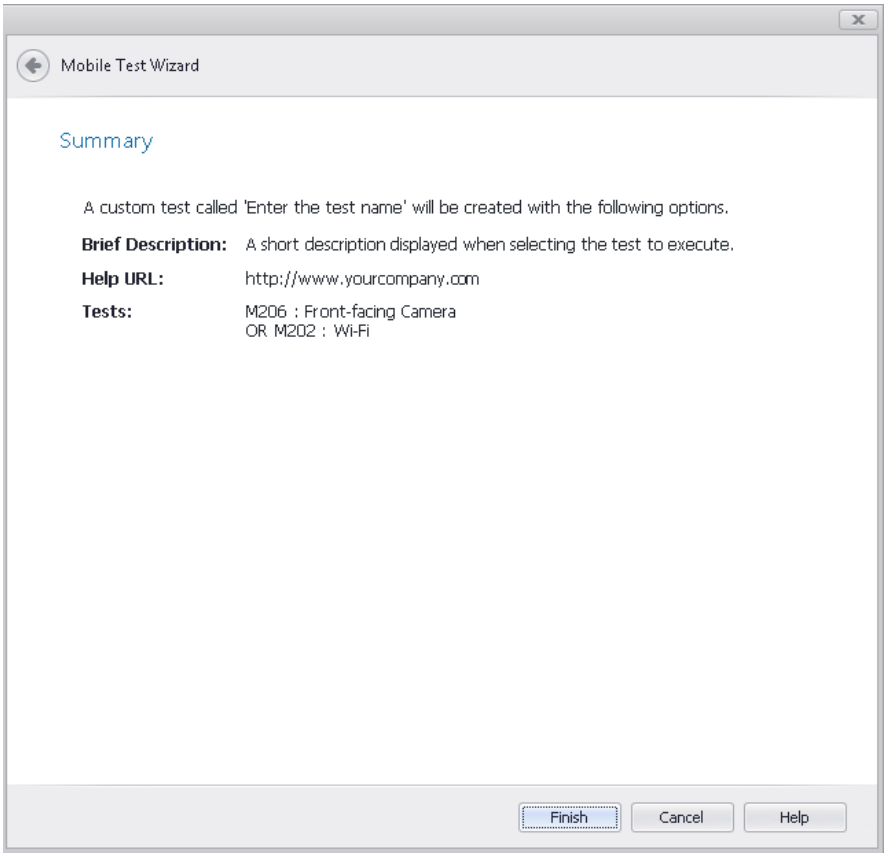


Figure 16-21: Summary Panel / Mobile Test Wizard

Review the summary data and click **Finish** to create or update the custom mobile test.

Rules Wizard

The **Rules Wizard** allows you to create user-defined ACE rules for later use by the Conflict Wizard. It is accessible by clicking **New** on the **Rules Viewer** dialog box.

The following panels are part of the Rules Wizard:

- Welcome Panel
- General Information Panel
- Additional Information
- Custom Options Panel
- Where Clause Panel
- DLL-Based ACEs Panel
- Summary Panel

Welcome Panel

The Rules Wizard allows you to create user-defined ACE rules for later use in conflict identification. The first panel displayed is the Welcome panel.

Click Next to proceed to the General Information panel.

General Information Panel

From the **General Information** panel, you can configure information about the new ACE rule. This information is used primarily for display information (Name, Brief Description, Description, and Information URL).

Also on the **General Information** panel, you specify whether you are creating a rule for a Windows Installer package (**MSI**) or a Microsoft App-V package (**AppV**).

Table 16-28 • General Information Panel Options

Option	Description
Name	The name of the ACE, used to organize the rule in Application Catalog. This is displayed in several places, including the Output Window, the Rules Viewer dialog box, and the **ACE Tests** tab of the **Options** dialog box.
Associated Table	Select the table in the Application Catalog which will be queried in the user-defined ACE. This also determines which columns are available in the Expression Builder dialog box, and which tokens are available for the Error and Display strings on the Custom Options panel of the Rules Wizard.
Package Type	Select **MSI** or **App-V** to identify the type of package that this rule will be run on.
Brief Description	Enter a brief description which will be displayed in the Rules Viewer, the **ACE Tests** tab of the **Options** dialog box, and in the Output Window. This description should be clear enough so users can understand when to use this ACE.
Description	Enter a description of the ACE, which is displayed at the bottom of the Rules Viewer dialog box when the ACE is selected and in the Output Window during conflict identification when the ACE executes.
Information URL	Provide a URL to get further information for the ACE.

Click Next to proceed to the Additional Information panel; click Back to return to the Welcome panel.

Additional Information

From the Additional Information panel, you can provide information for categorizing the ACE in relation to other ACEs in Application Catalog.

Table 16-29 • Additional Information Panel Option

Option	Description
Category	Either select an existing category for this new rule, or enter the name for a new category. These categories are displayed in the Conflict View and the **ACE Tests** tab of the **Options** dialog box. Ideally, any user-defined ACEs should be put in their own category.
Rule Type	Specify the type of ACE you are creating: • **Custom - Source Only Packages ACEs** allow you to quickly test any column or any value of a table to support your business logic. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a Source Only Packages ACE, you must define an SQL "Where" clause. Application Catalog supports external package conflict checking for **Custom - Source Only Packages ACEs**. The Source package can be selected from the Application Catalog Database or from an external MSI package. See Creating a Custom/Source Only Packages ACE for more information. • **Custom - Source and Target Packages ACEs** allow you to compare columns or values of Source package tables (new packages that you want to install onto a user's system) to columns or values of Target package tables (packages already installed on a user's system). For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the System directory of the Target system. To define a Source and Target Packages ACE, you must define a SQL "Where" clause, and specify a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages. Application Catalog *does not support* external package conflict checking for **Custom - Source and Target Packages ACEs**. Both the Source and Target Packages must be selected from the Application Catalog Database. See Creating a Custom/Source and Target Packages ACE for more information. • **DLL - User Provided DLL Based ACEs** allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and various programming languages to construct a Windows DLL. With DLL-Based ACEs, you can use a Conflict Application Resolution Definitions (CARDs) to fix the conflict. See Creating a User Provided DLL-Based ACE.

Click Next to proceed to the Custom Options panel or the DLL-Based ACEs panel; click Back to return to the General Information panel.

Custom Options Panel

From the Custom Options panel, you can create display strings for the Output Window and Conflict Details.

Table 16-30 • Custom Options Panel Option

Option	Description
Error String	This string appears in the Output Window when a violation of this ACE rule is detected during conflict identification. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter **`Failure in creating desktop icon`** in this field. ***Note*** • *Tokens allow you to insert values at run-time from the internal Application Catalog Database or an external MSI package into the string, such as specifying a file name. To use token replacement in the error string, click on the arrow to the right of the Error String field and pick a value from the list, or just type the values directly in the text box, in the following format:* • *Source Only Packages ACEs: [ColumnName]* • *Source and Target Packages ACEs: [Source.ColumnName] and [Target.ColumnName]* *For more information, see Token Grammar.*
Display String	This string appears in the **Application Conflicts** tab of the **Analyze Deployment Type View** after conflicts have been identified. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter **`Duplicate desktop icon found`** in this field. ***Note*** • *Tokens allow you to insert values at run-time from the internal Application Catalog Database or an external MSI package into the string, such as specifying a file name. To use token replacement in the error string, click on the arrow to the right of the Display String field and pick a value from the list, or just type the values directly in the text box, in the following format:* • **Source Only Packages ACEs***: [ColumnName]* • **Source and Target Packages ACEs***: [Source.ColumnName] and [Target.ColumnName]* *For more information, see Token Grammar.*
Severity	Specify whether this ACE should be an Error or a Warning.

Table 16-30 • Custom Options Panel Option (cont.)

Option	Description
Report 'No' results	User defined ACEs report conflicts based on the provided query. However, you may want to report the absence of the data if it could not be found. If you select this option, if the ACE does not return any results, it will be reported as an error (or warning), with the description and error strings as specified. If you expect a **No** result, do not use tokens in your display or error strings.

Click Next to proceed to the Where Clause panel; click Back to return to the Additional Information panel.

Token Grammar

What are Tokens?

Tokens represent data in the database that is inserted at runtime. In Application Catalog, tokens are used to insert values at runtime from the Application Catalog Database or an external MSI package into an Error or Display String.

How to Insert Tokens

Tokens are specified on the Custom Options panel of the Rules Wizard. To use token replacement in a string, click the arrow to the right of the Error String and Display String text boxes and select a column name from the list. The column name is then inserted into the string in the following format:

- **Source Only Packages ACEs**—`[ColumnName]`
- **Source and Target Packages ACEs**—`[Source.ColumnName]` and `[Target.ColumnName]`, with the prefix identifying whether the column is in the Source or Target package. If no prefix is used, Application Catalog assumes the "Source." prefix.

Note *• The Token list on the Custom Options panel is provided for your convenience; if you prefer, you can type the variables directly in the text boxes. For more information see, Token Grammar.*

Caution *• While you are creating a user-defined ACE in the Rules Wizard, if you initially select a Rule Type of Custom - Source and Target Packages, and then insert tokens in the Error String and Display String fields, the "Source." prefix will be used. But, before you finish creating this ACE, if you go back and change your Rule Type selection to Custom - Source Only Packages, the tokens that you initially entered into the Display and Error string text boxes will not automatically be updated to remove the "Source." prefix. For Application Catalog to correctly interpret this ACE, you need to manually go back to the Error and Display String fields and delete the "Source." prefix.*

Using the ProductName Pseudo-tokens

You can use the pseudo-tokens of [ProductName], [Source.ProductName] and [Target.ProductName] to insert the name of the Source or Target package in an Error or Display String, even though ProductName is not a table column name.

Where Clause Panel

From the Where Clause panel, you must define a valid Where clause for the ACE. If you do not know how to build a Where clause, you can click the Build Expressions button to launch the Expression Builder dialog box. You can also click Test to validate the Where clause syntax.

If you selected **Custom - Source and Target Packages** on the Additional Information panel, you must also select a **Join Column**—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages. For example, if you wanted to evaluate Source and Target packages that write files to the same directory, you might specify Directory as the Join column.

Tip • *To improve query performance, enclose table names in square brackets ([]).*

DLL-Based ACEs Panel

DLL-based ACEs require specific information about the ACE/CARD DLL file and the ACE and CARD Function Names to operate. Enter the following information:

Table 16-31 • DLL-Based ACEs Panel Options

Option	Description
ACE/CARD DLL File	Select the name of the ACE DLL that you are testing.
ACE Function Name	Enter the name that you chose to “export” for this ACE function.
CARD Function Name	Enter the name that you chose to “export” for this CARD function.
Test	Click the Test button next to the ACE Function Name or CARD Function Name to validate that the exported function does exist.

Click Next proceed to the Summary panel; click Back to return to the Additional Information panel.

Summary Panel

Once you have configured information for your ACE rule, the Summary panel displays information for final review.

Click Finish to accept this configuration and make the ACE available for conflict identification. Click Back to return to either the Where Clause panel or the DLL-Based ACEs panel.

17

Analyze Tests

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

The tests that are used to perform application conflict testing, operating system and best practices testing, and risk assessment testing on packages in the Application Catalog using Analyze are described in this section. Tests are listed by group.

Table 17-1 • Analyze Tests

Test Group	Description
Operating System Compatibility Tests	Test for application readiness on the following operating systems: • Microsoft Windows 8.1, 10-1809 (and 2019 LTSC), 10-1909, 10-2004, 10-20H2, 10-21H1 (32-bit and 64-bit), 10-21H2 (32-bit and 64-bit), 11-21H2 (64-bit) • Microsoft Windows Server 2012, 2012 R2, 2016, and 2019 • macOS 10.11 (El Capitan) and macOS 10.12 (Sierra) • Apple iOS 6 (32-bit) • Apple iOS 7, iOS 8, iOS 9, and iOS 10 (32-bit and 64-bit) • Microsoft Windows Phone 8.1 and 10 • Google Android 4.1, 4.2, 4.3, 4.4, 5.0, 6.0, and 7.0
MSIX Conversion Compatibility Tests	Test Windows Installer packages to determine if they are suitable candidates for conversion to msix: • Desktop

Table 17-1 • Analyze Tests (cont.)

Test Group	Description
Application Virtualization Compatibility Tests	**Installer Analysis Tests** Test Windows Installer packages to determine if they are suitable candidates for virtualization to the following formats: • Microsoft App-V • VMware ThinApp • Citrix XenApp
Best Practices Tests	Perform checks of the structure of Windows Installer packages, App-V packages, and Apple iOS mobile apps to determine if they violate best-practice guidelines. This category includes the following areas: • Windows Installer Internal Consistency Evaluators (ICEs) • Windows Installer Best Practices • Microsoft App-V Best Practices • Apple Best Practices

Risk Assessment Tests	Perform risk assessment checks for both mobile and desktop apps. This category includes the following areas:
	• Mobile Risk Assessment
	  • Android Mobile
	  • Apple Mobile
	  • Windows Mobile
	• Desktop Risk Assessment

Application Conflicts Tests	Identify conflicts between packages in the Application Catalog, as well as between packages and OS Snapshots. This category includes tests for Windows Installer packages and Microsoft App-V packages.
Remote Application Publishing Compatibility Tests	Examine Windows Installer packages for compatibility with: • Azure Application services • Windows Remote Desktop services

The Analyze Tests Reference section also describes the following test-related details:

- Analyze Resolutions
- Creating Your Own Custom ACE Tests
- Viewing ACE Metrics
- Location of ACE Files

Operating System Compatibility Tests

Use the Operating System Compatibility tests to check for application readiness on Microsoft Windows (Server, Desktop, and Phone), Apple iOS, Apple OS X, and Google Android operating systems.

The following subcategories of Operating System Compatibility tests are available:

- Windows Server and Desktop Tests
- Windows Phone Tests
- Apple iOS Tests
- Apple OS X Tests
- Google Android Tests

Windows Server and Desktop Tests

The Windows server and desktop operating system categories contain the following operating system compatibility tests:

- 32-Bit Driver
- 32-Bit Shell Extensions
- 64-Bit Files
- AdminUser or Privileged Launch Condition
- Application Requires Specific Minimum OS Version
- Application Requires VCLibs 12.0
- Application Requires VCLibs 14.0
- Application Requires WinJS 2.0 or Higher
- Applications with Known Windows OS Compatibility Issues
- Compatibility Issues with Known Issues at Startup
- Conditions Using AdminUser Property
- Conflicting Permission Tables
- Deferred Execution Custom Action Context
- Deprecated API Calls
- Deprecated Cluster Automation Server Functionality
- Deprecated Distributed File System Tool
- Deprecated Nested Windows Installer Packages
- Deprecated NETDDE Functionality
- Deprecated Proxy Configuration Tools
- Deprecated Server Manager Command-Line Tool
- Deprecated Windows Library Feature
- Drivers with Known Windows Compatibility Issues
- Excluded .NET Framework Payload Files
- ForceReboot Action
- Hard-Coded Paths
- Hard-Coded Paths in Script-Based Custom Actions
- IIS VBScripting Configuration
- Immediate Execution System-Context Custom Actions
- Installation to Secure Location
- Installers with Known Windows OS Compatibility Issues

- Interactive Services in Session 0
- Invalid Component Identifiers
- Junction Points
- Manifest Files Using Operating System Identifier
- Maximum Version of the OS Where This App Was Tested by the Developer
- Microsoft Management Console (MMC) Snap-ins Data Execution Prevention
- Mixed Per-User and Per-Machine Data
- Nested SendTo Menus
- Obsolete API Calls
- Obsolete File Associations
- Operating System Version Conditions
- Operating System Version Launch Conditions
- Quick Launch Bar
- Reboot Pending Launch Condition
- Reorganized Start Screen
- Restart Manager FilesInUse Dialog
- rundll32 Calls (User Account Control)
- Self-Update Functionality (User Account Control)
- Standard User Changes (User Account Control)
- Unmanifested Control Panel (.cpl) Files (User Account Control)
- Unmanifested Control Panel Applications (User Account Control)
- Unsigned Drivers
- Unsigned Executables
- Unsigned Windows Installer Database
- Unsupported .NET Framework 1.0/1.1 Applications
- Unsupported 16-Bit Files
- Unsupported 32-Bit Windows Help Files
- Unsupported DHTML Editing Control
- Unsupported GINA Functionality
- Windows Desktop Gadgets
- Windows Internet Explorer Protected Mode
- Windows Resource Protection Files
- Windows Resource Protection Registry Keys

32-Bit Driver

For the Operating System Compatibility test, the Windows Installer database is scanned for the presence of 32-bit drivers.

Test Group/Test Category

- **0437**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **0537**—Operating System Compatibility/Windows Server 2012
- **0637**—Operating System Compatibility/Windows Server 2016
- **1037**—Operating System Compatibility/Windows Server 2019
- **2937**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5337**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5537**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5737**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5937**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7137**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6737**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Message

```
ERROR_MSG_1: This Windows Installer database contains 32-bit driver (FILE_PATH) (Table: File, Key:
    FILE_NAME).
```

Background

Hardware devices require 64-bit drivers on a 64-bit versions of Windows. Legacy 32-bit drivers may not work on 64-bit Windows systems.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

The manufacturer of the driver should deliver a 64-bit version.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

32-Bit Shell Extensions

Note • *This test is not applicable to App-V packages.*

The Windows Installer database is scanned for the presence of 32-bit shell extensions, which cannot be loaded on 64-bit operating systems.

Test Group/Test Category

- **0451**—Operating System Compatibility/Windows 8.1 64-Bit
- **0551**—Operating System Compatibility/Windows Server 2012
- **0651**—Operating System Compatibility/Windows Server 2016
- **1051**—Operating System Compatibility/Windows Server 2019
- **2951**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5351**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5551**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5751**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5951**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7151**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6751**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Message

```
This Windows Installer database contains a 32-bit shell extension registered with file [FILE_NAME]
    (Table: File, Key: [FILE_KEY]).
```

Background

Since the introduction of 64-bit operating systems, some program features that are available on Windows 32-bit operating systems are not available on computers that are running an x64-based version of Windows. A common problem is that third-party Windows Explorer shell extensions are not added to the Windows Explorer menu, such as the Windows Explorer shell extensions for WinZip and for WinRAR. These symptoms occur because Windows Explorer cannot load the 32-bit .DLL files that are required by the Windows Explorer shell extensions feature.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the desired operating system should be delivered by its manufacturer. Alternatively, the 32-bit version of Windows Explorer can be used, which is located in the `%windir%\Syswow64` folder on the computer that is running the x64-based version of Windows.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

64-Bit Files

For this Operating System Compatibility test, the Windows Installer database is scanned for the presence of components that contain 64-bit files without conditions that enable them only for 64-bit Windows systems. The file extensions that are scanned are `.exe`, `.dll`, `.sys`, `.drv`, `.ocx`, `.cpl`, and `.src`.

Test Group/Test Category

- **0318**—Operating System Compatibility/Windows 8.1 32-Bit
- **2818**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5218**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5418**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5618**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5818**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7018**—Operating System Compatibility/Windows 10-21H2 (32-Bit)

Severity

Error

Message

```
This Windows Installer database contains 64-bit file [FILE_NAME] which might be installed in 32-bit
    systems (Table: File, key: [FILE_KEY]).
```

Background

Some software is intended to run only on 64-bit operating systems. If the launch conditions are missing or incorrect, 64-bit files might be installed on 32-bit Windows systems. If a user attempts to launch such a file, they encounter an error message stating that the file is not a valid Win32 application.

Resolution

The following resolutions are available.

Manual Fix

A 32-bit application compatible with the desired operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered by replacing 64-bit code with the appropriate 32-bit code.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

The 64-bit files that are configured to be installed on 32-bit systems are moved to separate 64-bit components with conditions that enable them only for 64-bit systems; this is done in a Windows Installer transform.

This fix is enabled by default.

Caution • *On 32-bit systems, those files will not be installed.*

AdminUser or Privileged Launch Condition

Note • *This test is not applicable to App-V packages.*

For this Operating System Compatibility test, the Windows Installer database is scanned for the presence of launch conditions that use the `AdminUser` or `Privileged` properties and that may prevent installation on the specified Windows operating system.

Test Group/Test Category

- **0349**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0449**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **0549**—Operating System Compatibility/Windows Server 2012
- **0649**—Operating System Compatibility/Windows Server 2016
- **1049**—Operating System Compatibility/Windows Server 2019
- **2849**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5249**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5449**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5649**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5849**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7049**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2949**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5349**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5549**—Operating System Compatibility/Windows 10-2004 (64-Bit)

- **5749**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5949**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7149**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6749**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Warning

Messages

- `This Windows Installer database contains LaunchCondition [LAUNCHCONDITION_NAME] using AdminUser property (Table: LaunchCondition, Key: [LAUNCHCONDITION_CONDITION]).`
- `This Windows Installer database contains LaunchCondition [LAUNCHCONDITION_NAME] using Privileged property (Table: LaunchCondition, Key: [LAUNCHCONDITION_CONDITION]).`

Background

The Windows Installer properties `AdminUser` and `Privileged` may be used to prevent users who do not have administrator or elevated privileges from running the installation. These properties should not be used in the `LaunchCondition` table because Windows Installer may initially spoof their value during evaluation of the `LaunchCondition` table and set both to 1 even if the user is not an administrator and has not received elevated privileges yet. This behavior is present because privileges are elevated much later, and the result of authentication is not known when initial launch conditions are evaluated.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

A LaunchCondition table entry that uses the `AdminUser` or `Privileged` properties should be migrated to a type 19 custom action that uses the following condition:

```
NOT Privileged
```

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Application Requires Specific Minimum OS Version

For this Operating System Compatibility test, the application is scanned to determine if it requires a specific minimum OS version.

Test Group/Test Category

- **3003**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **3103**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **4101**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **4501**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **4701**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **6001**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **6201**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **8001**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **4201**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **4601**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **4801**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **6101**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **6301**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **8101**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6501**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Resolution

Application should only be installed on a device with a compatible OS version.

Application Requires VCLibs 12.0

For this Operating System Compatibility test, the application is scanned to determine if it requires version VCLibs 12.0 or higher installed on the specified Windows operating system.

Test Group/Test Category

- **3007**—Operating System Compatibility/Windows 8.1 (32-bit)
- **3107**—Operating System Compatibility/Windows 8.1 (64-Bit)

Severity

Error

Resolution

Application should only be installed on a device where VCLibs 12.0 is installed.

Application Requires VCLibs 14.0

For this Operating System Compatibility test, the application is scanned to determine if it requires version VCLibs 14.0 or higher installed on the specified Windows operating system.

Test Group/Test Category

- **4107**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **4507**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **4707**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **6007**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **6207**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **8007**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **4207**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **4607**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **4807**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **6107**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **6307**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **8107**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6507**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Resolution

Application should only be installed on a device where VCLibs 14.0 is installed.

Application Requires WinJS 2.0 or Higher

For this Operating System Compatibility test, the application is scanned to determine if it requires version WinJS 2.0 or higher installed on the specified Windows operating system.

Test Group/Test Category

- **3008**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **3108**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **4108**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **4508**—Operating System Compatibility/Windows 10-1909 (32-Bit)

- **4708**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **6008**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **6208**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **8008**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **4208**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **4608**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **4808**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **6108**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **6308**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **8108**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6508**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Resolution

Application should only be installed on a device where WinJS 2.0 or higher is installed.

Applications with Known Windows OS Compatibility Issues

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the application is scanned for known executable compatibility issues against the Windows Microsoft Application Compatibility Database.

Test Group/Test Category

- **0660**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0760**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2860**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5260**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5460**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5660**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5860**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7060**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2960**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)

- **5360**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5560**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5760**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5960**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7160**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6760**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0560**—Operating System Compatibility/Windows Server 2012
- **0960**—Operating System Compatibility/Windows Server 2016
- **1060**—Operating System Compatibility/Windows Server 2019

Severity

Error or Warning: This test will generate an Error if Microsoft has determined that the detected issue would prevent the installer from installing on this operating system, or it will generate a Warning if Microsoft has provided a workaround for the detected issue that will enable the installer to install on this operating system.

Message

```
[FILE_NAME]: Update has known compatibility issues with Windows [VERSION]. For more information, contact
    [VENDOR_NAME].
```

Resolution

Because Microsoft has detected a compatibility issue with this application, upgrade to a more recent version of this application, if possible.

Compatibility Issues with Known Issues at Startup

For this operating system compatibility test, the Windows Installer database is scanned for the presence of content that may trigger Application Help (Apphelp) messages during installation or at application startup. These types of messages warn end users that an application may have compatibility problems.

Test Group/Test Category

- **0339**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0439**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2839**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5239**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5439**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5639**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5839**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7039**—Operating System Compatibility/Windows 10-21H2 (32-Bit)

- **2939**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5339**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5539**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5739**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5939**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7139**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6739**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0539**—Operating System Compatibility/Windows Server 2012
- **0639**—Operating System Compatibility/Windows Server 2016
- **1039**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains a program known to have compatibility issues. It may trigger
    Application Help (Apphelp) message during installation.
```

Background

When an application is launched, the Program Compatibility Assistant warns end users if the application is known to have compatibility issues. The list of these applications is stored in the System application compatibility database. These messages that the Program Compatibility Assistant displays are called Application Help (Apphelp) messages.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the desired Windows operating system should be delivered by its manufacturer.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Conditions Using AdminUser Property

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of the `AdminUser` property in conditions in the Install UI sequence.

Test Group/Test Category

- **0350**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0450**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2850**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5250**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5450**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5650**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5850**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7050**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2950**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5350**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5550**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5750**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5950**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7150**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6750**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0550**—Operating System Compatibility/Windows Server 2012
- **0650**—Operating System Compatibility/Windows Server 2016
- **1050**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains condition [CONDITION_NAME] using AdminUser property (Table:
    InstallUISequence, Key: [InstallUISequence_ACTION])
```

Background

The Windows Installer properties `AdminUser` and `Privileged` may be used to prevent users who do not have administrator or elevated privileges from running certain parts of the installation. The installation sets the `Privileged` property to 1 if the user has elevated privileges; it sets the `AdminUser` property to 1 only if the user was an administrator. The differences between these properties may have been used in some legacy packages. On Windows Vista/Windows Server 2008 and later systems, these two Windows Installer properties are always the same and the `Privileged` property is recommended.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

The Windows Installer property `AdminUser` should be avoided; the `Privileged` property should be used instead. Packages that require distinct `Privileged` and `AdminUser` properties can restore the difference by setting the `MSIUSEREALADMINDETECTION` property.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Conflicting Permission Tables

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the usage of the LockPermissions table in conjunction with the MsiLockPermissionsEx table.

Test Group/Test Category

- **0328**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0428**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2828**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5228**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5428**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5628**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5828**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7028**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2928**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5328**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5528**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5728**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5928**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7128**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6728**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

- **0528**—Operating System Compatibility/Windows Server 2012
- **0628**—Operating System Compatibility/Windows Server 2016
- **1028**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains both LockPermissions and MsiLockPermissionsEx tables (Table:
    LockPermissions, MsiLockPermissionsEx).
```

Background

Since Windows Installer 5 (introduced with Windows 7), the MsiLockPermissionsEx table should replace the use of the LockPermissions table for managing access permissions. The extended functionality provided by the MsiLockPermissionsEx table enables a package to secure Windows Services, files, folders, and registry keys. Beginning with Windows Installer 5, the installation fails with error message 1941 if the Windows Installer package contains both a LockPermissions table and a MsiLockPermissionsEx table. Existing Windows Installer packages that contain only the LockPermissions table can still be installed using Windows Installer 5.

Note • *Windows Installer 4.5 and earlier ignore the* MsiLockPermissionsEx *table.*

Resolution

The following resolutions are available.

Manual Fix

If the LockPermissions and MsiLockPermissionsEx tables are not empty, all entries from LockPermissions table should be migrated to the MsiLockPermissionsEx table, and the LockPermissions table should be removed. Otherwise, at least one of those empty tables (either LockPermissions or MsiLockPermissionsEx) should be removed.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

The conflict between the LockPermissions and the MsiLockPermissionsEx table is resolved in a Windows Installer transform. If either one of the tables is empty, it is removed. If both tables are populated, entries from the LockPermissions table are converted to the MsiLockPermissionsEx table, and afterwards the empty LockPermissions table is removed.

This fix is enabled by default.

Deferred Execution Custom Action Context

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of any deferred execution custom actions that are not running in system context.

Test Group/Test Category

- **0305**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0405**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2805**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5205**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5405**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5605**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5805**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7005**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2905**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5305**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5505**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5705**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5905**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7105**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6705**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0505**—Operating System Compatibility/Windows Server 2012
- **0605**—Operating System Compatibility/Windows Server 2016
- **1005**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a deferred execution custom action [CUSTOM_ACTION_NAME] that is
    not running in system context (with no impersonation) (Table: CustomAction, Key:
    [CUSTOM_ACTION_KEY]).
```

Background

The purpose of a deferred execution custom action is to delay a system change until the installation script runs. This differs from a regular custom action or a standard action, where the installation executes the action immediately. Deferred execution custom actions that make system changes should be running in system context (with no impersonation).

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

All deferred execution custom actions that make system changes should be adjusted to run in system context (with no impersonation).

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Deferred execution custom actions are marked to run in system context (with no impersonation) in a Windows Installer transform.

Deprecated API Calls

For this operating system compatibility test, the Windows Installer database is scanned for references to deprecated APIs. The file extensions that are scanned are `.exe`, `.dll`, `.sys`, `.src`, `.drv`, `.cpl`, and `.ocx`.

Test Group/Test Category

- **0322**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0422**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2822**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5222**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5422**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5622**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5822**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7022**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2922**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5322**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5522**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5722**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5922**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7122**—Operating System Compatibility/Windows 10-21H2 (64-Bit)

- **6722**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0522**—Operating System Compatibility/Windows Server 2012
- **0622**—Operating System Compatibility/Windows Server 2016
- **1022**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a reference to a deprecated API call [CALL_NAME] in file
    [FILE_NAME] (Table: File, Key: [FILE_KEY]).
```

Background

A number of APIs that were previously used in Microsoft Windows are no longer supported on Windows 7 systems. Any application that calls these deprecated APIs might behave in unexpected ways.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

A Windows 7-compatible application should be delivered by its manufacturer. Self-developed applications should be re-engineered to avoid calling deprecated APIs. Deprecated APIs are documented in Microsoft SDK updates.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Deprecated Cluster Automation Server Functionality

For this operating system compatibility test, the Windows Installer database is scanned for the presence of calls to the APIs **BackupClusterDatabase** or **RestoreClusterDatabase**. The extensions of the files that are scanned are `.exe`, `.dll`, `.sys`, `.src`, `.drv`, `.cpl`, and `.ocx`.

Test Group/Test Category

- **0533**—Operating System Compatibility/Windows Server 2012
- **0633**—Operating System Compatibility/Windows Server 2016
- **1033**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains a reference to a deprecated Cluster Automation Server API call
    [CALL_NAME] in file [FILE_NAME] (Table: File, Key: [FILE_KEY]).
```

Background

The Cluster Automation Server provides a set of automation objects that expose a complete cluster management interface to scripting languages, allowing independent software vendors (ISVs) to develop web-based remote administration tools. The Cluster Automation Server simplified and enhanced the process of creating a cluster management application. The APIs **BackupClusterDatabase** and **RestoreClusterDatabase** are deprecated on Windows Server 2008 R2 systems.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

A Windows Server 2008 R2-compatible application should be delivered by its manufacturer. Self-developed applications should be re-engineered to use the new Cluster VSS Writer to perform backups and restores of the cluster configuration.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Deprecated Distributed File System Tool

For this operating system compatibility test, the package is scanned for the presence of references to `dfscmd.exe` inside shortcuts, custom actions, and script files. Scanned file extensions are: `cmd` and `vbs`.

Test Group/Test Category

- **0857**—Operating System Compatibility/Windows Server 2012
- **0957**—Operating System Compatibility/Windows Server 2016
- **1057**—Operating System Compatibility/Windows Server 2019

Severity

Error

Messages

One of the following error messages is displayed:

```
This Windows Installer database contains a reference to Dfscmd.exe in script [FILE_NAME] (Table: File,
    Key: [FILE_KEY]).

This Windows Installer database contains a reference to Dfscmd.exe in shortcut [SHORTCUT_NAME] (Table:
    Shortcut, Key: [SHORTCUT_NAME]).

This Windows Installer database contains a reference to Dfscmd.exe in custom action [CUSTOM_ACTION_NAME]
    (Table: CustomAction, Key: [CUSTOM_ACTION_NAME]).
```

Background

Distributed File System (DFS) is a set of client and server services that allow an organization using Microsoft Windows Server to organize many distributed file shares into a distributed file system. Starting with Windows Server 2012 R2, Microsoft has deprecated the command line tool that configures DFS folders and folder targets in a DFS namespace.

Resolution

The following resolutions are available.

Manual Fix

A Windows Server 2012 R2 compatible application should be delivered by its manufacturer. Self-developed applications should be re-engineered to use Windows PowerShell cmdlets for Distributed File Namespaces or the `dfsutil.exe` command set.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Deprecated Nested Windows Installer Packages

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of the following custom actions types:

- 7 (concurrent installation of an embedded Windows Installer package)
- 23 (concurrent installation of a source Windows Installer package)
- 39 (concurrent installation of an advertised Windows Installer package)

Test Group/Test Category

- **0306**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0406**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2806**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5206**—Operating System Compatibility/Windows 10-1909 (32-Bit)

- **5406**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5606**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5806**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7006**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2906**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5306**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5506**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5706**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5906**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7106**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6706**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0506**—Operating System Compatibility/Windows Server 2012
- **0606**—Operating System Compatibility/Windows Server 2016
- **1006**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains custom action type 7 (concurrent installation of an embedded Windows Installer Package) (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`
- `This Windows Installer database contains custom action type 23 (concurrent installation of a source Windows Installer package) (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`
- `This Windows Installer database contains custom action type 39 (concurrent installation of an advertised Windows Installer package) (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`

Background

Concurrent installations, also called *nested installations*, install another Windows Installer package during a currently running installation. Microsoft has deprecated this Windows Installer feature since it might cause several issues that range from patch and upgrade problems to unwanted reboots.

Resolution

The following resolutions are available.

Manual Fix

Custom actions type 7, 23, and 39 should be disabled. A setup application and an external UI handler should be created to install all needed Windows Installer packages sequentially.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Custom actions type 7, 23, and 39 are disabled in a Windows Installer transform. Nested Windows Installer databases are extracted and put in a subfolder called `Dependencies_%Source%` adjacent to the original Windows Installer database. Additionally, an installation script called `Install_Dependencies_<name_of_original_msi>.cmd` is created; it sequentially launches the dependent Windows Installer packages that were originally called from within the parent.

This fix is enabled by default.

Deprecated NETDDE Functionality

For this operating system compatibility test, the Windows Installer database is scanned for the presence of any registry entries that reference `NETDDE.EXE`.

Test Group/Test Category

- **0329**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0429**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2829**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5229**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5429**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5629**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5829**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7029**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2929**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5329**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5529**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5729**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5929**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7129**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6729**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0529**—Operating System Compatibility/Windows Server 2012
- **0629**—Operating System Compatibility/Windows Server 2016
- **1029**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains a Network DDE call [REGISTRY_VALUE] in registry entry
    [REGISTRY_KEY]\[REGISTRY_NAME] (Table: Registry, Key: [REGISTRY_ENTRY]).
```

Background

Microsoft deprecated Network DDE (NetDDE) in Windows Vista. NetDDE is a technology that allows applications that use the DDE transport to exchange data over the network. Applications that use this technology might fail.

Note • *Regular DDE is still supported.*

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to use a different networking technology, such as DCOM or Windows Communication Foundation.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Deprecated Proxy Configuration Tools

For this operating system compatibility test, the Windows Installer database is scanned for the presence of registry entries or files that refer to `ProxyCfg.exe`. The extensions of files that are scanned are .vbs and .cmd.

Test Group/Test Category

- **0338**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0438**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2838**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5238**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5438**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5638**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5838**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7038**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2938**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)

- **5338**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5538**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5738**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5938**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7138**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6738**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0538**—Operating System Compatibility/Windows Server 2012
- **0638**—Operating System Compatibility/Windows Server 2016
- **1038**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains a reference to deprecated ProxyCfg.exe file in script [FILE_NAME] (Table: File, Key: [FILE_KEY]).`
- `This Windows Installer database contains a reference to deprecated ProxyCfg.exe file in registry entry [REGISTRY_KEY]\[REGISTRY_NAME] (Table: Registry, Key: [REGISTRY_ENTRY]).`

Background

`ProxyCfg.exe` is a process that is associated with the Proxy Configuration Tool for Windows HTTP Services from Microsoft. In Windows Vista and later systems, this file is not a part of the operating system; it was replaced by `netsh.exe`.

Resolution

The following resolutions are available.

Manual Fix

References to `ProxyCfg.exe` should be replaced with the corresponding `netsh.exe` commands.

Basic Auto Fix

References to `ProxyCfg.exe` are replaced with the corresponding `netsh.exe` commands in a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Deprecated Server Manager Command-Line Tool

For this operating system compatibility test, the Windows Installer database is scanned for the presence of references to ServerManagerCmd.exe inside shortcuts, custom actions, and script files. The file extensions that are scanned are .cmd, and .vbs.

Test Group/Test Category

- **0531**—Operating System Compatibility/Windows Server 2012
- **0631**—Operating System Compatibility/Windows Server 2016
- **1031**—Operating System Compatibility/Windows Server 2019

Severity

Error

Messages

- `This Windows Installer database contains a reference to ServerManagerCmd.exe in script [FILE_NAME] (Table: File, Key: [FILE_KEY]).`
- `This Windows Installer database contains a reference to ServerManagerCmd.exe in shortcut [SHORTCUT_NAME] (Table: Shortcut, Key: [SHORTCUT_KEY]).`
- `This Windows Installer database contains a reference to ServerManagerCmd.exe in custom action [CUSTOM_ACTION_KEY] (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`

Background

ServerManagerCmd.exe is a tool that has been part of Server Manager. It can run queries, perform installations, and remove roles and features. The ServerManagerCmd.exe command-line tool is deprecated on Windows Server 2008 R2 systems.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified version of Windows Server should be delivered by its manufacturer. Self-developed applications should be re-engineered to use the Server Manager PowerShell cmdlets instead of the ServerManagerCmd.exe command-line tool.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Deprecated Windows Library Feature

For this operating system compatibility test, the package is scanned for the presence of Windows Library files.

Test Group/Test Category

- **0656**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0756**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2856**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5256**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5456**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5656**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5856**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7056**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2956**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5356**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5556**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5756**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5956**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7156**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6756**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0856**—Operating System Compatibility/Windows Server 2012
- **0956**—Operating System Compatibility/Windows Server 2016
- **1056**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains deprecated Windows Library file (Table: File, Key: [FILE_KEY])
```

Background

Libraries were introduced with the release of Windows 7 to organize files across the PC or network. Starting with Windows 8.1, the Windows Library feature has been replaced with the Skydrive, so, by default, after creating the library, it is not displayed in Windows Explorer.

Resolution

The following resolutions are available.

Manual Fix

An application compatible with the specified version of Windows should be delivered by its manufacturer. Self-developed applications should not install Windows libraries.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Drivers with Known Windows Compatibility Issues

For this operating system compatibility test, the application is scanned for known driver compatibility issues against the Windows Microsoft Application Compatibility Database.

Test Group/Test Category

- **0659**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0759**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2859**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5259**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5459**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5659**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5859**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7059**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2959**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5359**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5559**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5759**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5959**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7159**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6759**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0559**—Operating System Compatibility/Windows Server 2012
- **0959**—Operating System Compatibility/Windows Server 2016
- **1059**—Operating System Compatibility/Windows Server 2019

Severity

Error or Warning: This test will generate an Error if Microsoft has determined that the detected issue would prevent the installer from installing on this operating system, or it will generate a Warning if Microsoft has provided a workaround for the detected issue that will enable the installer to install on this operating system.

Message

```
[FILE_NAME] AND [BINARY_PRODUCT_VERSION] as Numeric<=[NUMBER] OR NOT [FILE_NAME] AND
    [BINARY_PRODUCT_VERSION] as Numeric<=[NUMBER]: A driver is installed that causes stability problems
    with your system. This driver will be disabled. Please contact the driver manufacturer for an update
    that is compatible with this version of Windows.
```

Resolution

Because Microsoft has detected a compatibility issue with this driver, use a more recent version of this driver, if possible.

Excluded .NET Framework Payload Files

For this operating system compatibility test, the Windows Installer database is scanned for the presence of .NET assemblies that were compiled with Microsoft .NET Framework 2.0, 3.0, or 3.5.

Test Group/Test Category

- **0341**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0441**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2841**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5241**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5441**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5641**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5841**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7041**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2941**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5341**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5541**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5741**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5941**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7141**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6741**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0541**—Operating System Compatibility/Windows Server 2012
- **0641**—Operating System Compatibility/Windows Server 2016
- **1041**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a .NET assembly [FILE_NAME] compiled with Microsoft .NET
    Framework version [VERSION] (Table: File, Key: [FILE_KEY]).
```

Background

On Windows 8/Windows Server 2012 and later systems, Microsoft .NET Framework 4.5 is enabled by default. The manifests for .NET Framework 3.5 (including .NET 2.0 and 3.0) are also included, but without the supporting payload files. With a clean installation of Windows 8/Windows Server 2012 or later, applications that require .NET Framework 2.0 or 3.5 might trigger a request for the necessary files.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to use Microsoft .NET Framework 4.0 or later. Where this is not feasible, Microsoft provides a downloadable .NET Framework 3.5 (including .NET 2.0 and 3.0) feature available through Windows Update or on the original installation media.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

ForceReboot Action

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of a ForceReboot action that may be launched on Windows systems.

Test Group/Test Category

- **0347**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0447**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2847**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5247**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5447**—Operating System Compatibility/Windows 10-2004 (32-Bit)

- **5647**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5847**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7047**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2947**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5347**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5547**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5747**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5947**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7147**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6747**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0547**—Operating System Compatibility/Windows Server 2012
- **0647**—Operating System Compatibility/Windows Server 2016
- **1047**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains ForceReboot action that may run on Windows [VERSION] (Table:
    InstallExecuteSequence, Key: ForceReboot).
```

Background

The ForceReboot action prompts the user for a restart of the system during the installation. If the installation is displaying a user interface, the installation shows a dialog at each ForceReboot action; the dialog prompts the user to restart the system. The user must respond to this prompt before continuing with the installation. If the installation has no user interface, the system automatically restarts at the ForceReboot action. When Windows Installer determines that a restart is necessary, it automatically prompts the user to restart at the end of the installation, regardless of whether there are any ForceReboot or ScheduleReboot actions in the sequence.

Resolution

The following resolutions are available.

Manual Fix

A condition that suppresses the ForceReboot action on the specified Windows operating system systems should be added.

This fix is enabled by default.

Basic Auto Fix

A condition is added to the ForceReboot action to disable the action on Windows systems; this is done through a Windows Installer transform.

Advanced Auto Fix

No resolution is available.

Hard-Coded Paths

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for hard-coded paths in the following tables: Registry, IniFile, Shortcut, ServiceControl, ServiceInstall, and CustomAction (excluding script-based custom actions).

Test Group/Test Category

- **0327**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0427**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2827**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5227**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5427**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5627**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5827**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7027**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2927**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5327**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5527**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5727**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5927**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7127**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6727**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0527**—Operating System Compatibility/Windows Server 2012
- **0627**—Operating System Compatibility/Windows Server 2016
- **1027**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- This Windows Installer database contains a hard-coded path "[REGISTRY_VALUE]" in registry entry [REGISTRY_KEY] (Table: Registry, Key: [REGISTRY_ENTRY]).
- This Windows Installer database contains a hard-coded path "[INI_FILE_KEY]" in a key of an INI file (Table: IniFile, Key: [INIFILE_KEY]).
- This Windows Installer database contains a hard-coded path "[INI_FILE_VALUE]" in a value of an INI file (Table: IniFile, Key: [INIFILE_KEY]).
- This Windows Installer database contains a hard-coded path "[CUSTOM_ACTION_TARGET]" in the CustomAction table (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).
- This Windows Installer database contains a hard-coded path "[SHORTCUT_ARGUMENTS]" in shortcut [SHORTCUT_NAME] (Table: Shortcut, Key: [SHORTCUT_KEY]).
- This Windows Installer database contains a hard-coded path "[SERVICECONTROL_ARGUMENTS]" in the ServiceControl table (Table: ServiceControl, Key: [SERVICECONTROL_KEY]).
- This Windows Installer database contains a hard-coded path "[SERVICEINSTALL_ARGUMENTS]" in the ServiceInstall table (Table: ServiceInstall, Key: [SERVICEINSTALL_KEY]).

Background

During migration to a new environment, some paths might have changed or became obsolete, eventually causing hard-coded values in Windows Installer databases (for example, in the Registry or IniFile tables) to no longer be valid. This could result in installation failures or functionality issues.

Note • *ICE48 checks for directories that are hard-coded to local paths.*

Resolution

The following resolutions are available.

Manual Fix

For each hard-coded path, a property that contains that value should be created. That property should replace the hard-coded path.

Basic Auto Fix

Hard-coded paths are replaced with properties that contain the original paths in a Windows Installer transform. The newly created properties are named **ASFIX_PATH_#** (where # is an enumerator to uniquely name the properties).

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Hard-Coded Paths in Script-Based Custom Actions

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for hard-coded paths inside script-based custom actions.

Test Group/Test Category

- **0326**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0426**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2826**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5226**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5426**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5626**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5826**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7026**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2926**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5326**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5526**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5726**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5926**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7126**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6726**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0526**—Operating System Compatibility/Windows Server 2012
- **0626**—Operating System Compatibility/Windows Server 2016
- **1026**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains a hard-coded path [PATH] in a script-based custom action [CUSTOM_ACTION_KEY] stored in property [PROPERTY] (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]; Table: Property, Key: [PROPERTY]).`
- `This Windows Installer database contains a hard-coded path [PATH] in a script-based custom action [CUSTOM_ACTION_KEY] (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`

- `This Windows Installer database contains a hard-coded path [PATH] in a script-based custom action [CUSTOM_ACTION_KEY] stored in a binary stream [STREAM] (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]; Table: Binary, Key: [BINARY_KEY]).`
- `This Windows Installer database contains a hard-coded path [PATH] in a script-based custom action [CUSTOM_ACTION_KEY] installed within this product (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]; Table: File, Key: [FILE_KEY]).`

Background

During migration to a new environment, some paths might have changed or became obsolete, eventually causing hard-coded values in script-based custom actions to no longer be valid. This could result in installation failures or functionality issues.

Note • *ICE48 checks for directories that are hard-coded to local paths.*

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

All hard-coded paths in script-based custom actions should be replaced with Windows Installer properties or environment variables.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

IIS VBScripting Configuration

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of custom actions and scripts that are used to configure an Internet Information Services (IIS) server. Additionally, the database is scanned for the presence of deprecated IIS libraries. The extensions that are scanned are .vbs and .cmd.

Test Group/Test Category

- **0534**—Operating System Compatibility/Windows Server 2012
- **0634**—Operating System Compatibility/Windows Server 2016
- **1034**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains custom action [CUSTOM_ACTION_KEY] that configures Internet Information Services (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`
- `This Windows Installer database contains custom action [CUSTOM_ACTION_KEY] that configures Internet Information Services via script [FILE_KEY] (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).`
- `This Windows Installer database contains script [FILE_NAME] that configures Internet Information Services via [FILE_KEY] (Table: File, Key: [FILE_KEY]).`
- `This Windows Installer database contains script [FILE_NAME] that uses deprecated Internet Information Services interface (Table: File, Key: [FILE_KEY]).`

Background

IIS 6 had several interfaces for automated management via scripts. Windows Server 2008 and later systems (with IIS 7 and later) do not support this functionality. Many applications that use VBScript code to manipulate IIS configuration might not function as expected.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

A Windows Server 2008 R2-compatible application should be delivered by its manufacturer. Self-developed applications should be re-engineered to use IIS 7 interfaces. If this is not feasible, the IIS 6 Management Compatibility role should be installed using Server Manager.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Immediate Execution System-Context Custom Actions

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of any non-impersonated custom actions that are not scheduled to run in the script (deferred execution).

Test Group/Test Category

- **0304**—Operating System Compatibility/Windows 8.1 (32-Bit)

- **0404**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2804**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5204**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5404**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5604**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5804**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7004**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2904**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5304**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5504**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5704**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5904**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7104**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6704**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0504**—Operating System Compatibility/Windows Server 2012
- **0604**—Operating System Compatibility/Windows Server 2016
- **1004**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains non-impersonated immediate custom action [CUSTOM_ACTION_NAME]
    (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).
```

Background

The purpose of a deferred execution custom action is to delay a system change until the installation script runs. This differs from a regular custom action or a standard action, where the installation executes the action immediately. Immediate custom actions are not elevated; therefore, actions that make system changes should be deferred in execution until in-script is generated.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

All custom actions that make system changes should be deferred in execution to run in the script.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Non-impersonated immediate custom actions are marked to run as deferred actions in a Windows Installer transform.

Installation to Secure Location

For this operating system compatibility test, the Windows Installer database is scanned for the presence of files that are installed to the `Program Files\WindowsApps` folder.

Test Group/Test Category

- **0342**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0442**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2842**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5242**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5442**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5642**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5842**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7042**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2942**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5342**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5542**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5742**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5942**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7142**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6742**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0542**—Operating System Compatibility/Windows Server 2012
- **0642**—Operating System Compatibility/Windows Server 2016
- **1042**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains file FILE_NAME being installed to restricted location PATH
    (Table: File, Key: FILE_KEY).
```

Background

When a Windows Store application is added to a Windows 8/Windows Server 2012 or later system, it is installed in the `Program Files\WindowsApps` directory. If desktop applications are installed to this location, they may cause collisions with existing configurations. In addition, some antivirus/anti-malware detectors identify this location as a potential cause for concern.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should not install to the restricted location.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Installers with Known Windows OS Compatibility Issues

For this operating system compatibility test, the application is scanned for known installer compatibility issues against the Microsoft Application Compatibility Database.

Test Group/Test Category

- **0658**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0758**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2858**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5258**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5458**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5658**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5858**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7058**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2958**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5358**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5558**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5758**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5958**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7158**—Operating System Compatibility/Windows 10-21H2 (64-Bit)

- **6758**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0558**—Operating System Compatibility/Windows Server 2012
- **0958**—Operating System Compatibility/Windows Server 2016
- **1058**—Operating System Compatibility/Windows Server 2019

Severity

Error or Warning: This test will generate an Error if Microsoft has determined that the detected issue would prevent the installer from installing on this operating system, or it will generate a Warning if Microsoft has provided a workaround for the detected issue that will enable the installer to install on this operating system.

Message

```
[PACKAGE_NAME] AND [PRODUCT_NAME] AND [PRODUCT_VERSION] has a known compatibility issue with this
    version of Windows and might not run as expected. For more information, contact [VENDOR_NAME].
```

Resolution

Because Microsoft has detected a compatibility issue with this installer, use a more recent version of this installer, if possible.

Interactive Services in Session 0

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of services that are being installed or configured and that require interaction with the user's environment.

Test Group/Test Category

- **0307**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0407**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2807**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5207**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5407**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5607**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5807**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7007**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2907**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5307**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5507**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5707**—Operating System Compatibility/Windows 10-20H2 (64-Bit)

- **5907**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7107**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6707**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0507**—Operating System Compatibility/Windows Server 2012
- **0607**—Operating System Compatibility/Windows Server 2016
- **1007**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains interactive service [SERVICE_NAME] ([FILE_NAME]) (Table:
    ServiceInstall, Key: [SERVICEINSTALL_KEY]).
```

Background

Windows system processes and services run in Session 0. On systems earlier than Windows Vista, the first user who logged on to the console also used Session 0. Running services and user applications together in Session 0 poses a security risk because services run at elevated privileges and therefore are targets for malicious agents trying to elevate their own privilege levels. To eliminate the security risk, Session 0 is non-interactive on Windows Vista and later systems; that is, the first user logs on to Session 1. However, services still run in Session 0. This means that services that need to display user interface dialog boxes or communicate with user applications must properly secure a communication channel. If this is not done, the services fail to work properly on Windows 7 and later systems.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to establish correct communications with required services that are running in Session 0.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Invalid Component Identifiers

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of components with null, invalid, or duplicated component identifiers.

Test Group/Test Category

- **0344**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0444**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2844**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5244**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5444**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5644**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5844**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7044**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2944**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5344**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5544**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5744**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5944**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7144**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6744**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0544**—Operating System Compatibility/Windows Server 2012
- **0644**—Operating System Compatibility/Windows Server 2016
- **1044**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains component [COMPONENT_NAME] with null Globally Unique Identifier (GUID) (Table: Component, Key: [COMPONENT_NAME]).`
- `This Windows Installer database contains component [COMPONENT_NAME] with invalid Globally Unique Identifier (GUID) [COMPONENT_ID] (Table: Component, Key: [COMPONENT_NAME]).`
- `This Windows Installer database contains duplicated component Globally Unique Identifier (GUID) [COMPONENT_ID] (Table: Component, Keys: [COMPONENT_NAME]).`

Background

Windows Installer tracks every component by its component GUID, which is specified in the Component table. It is essential for the operation of the Windows Installer reference-counting mechanism that the component GUID is set and its value is correct. The `ComponentID` property takes a string that is formatted as a GUID, using the format {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}, where X is a hexadecimal digit (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F). The braces are required.

Resolution

The following resolutions are available.

Manual Fix

Each component should receive a non-null, valid GUID in the ComponentId field.

Basic Auto Fix

Valid GUIDs are generated for each component that has a null or invalid GUID; this is done through a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Junction Points

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the usage of changed or obsolete junction points in the following tables: CustomAction, IniFile, Registry, RemoveIniFile, ServiceControl, ServiceInstall, Shortcut, and Environment.

Test Group/Test Category

- **0312**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0412**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2812**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5212**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5412**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5612**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5812**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7012**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2912**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5312**—Operating System Compatibility/Windows 10-1909 (64-Bit)

- **5512**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5712**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5912**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7112**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6712**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0512**—Operating System Compatibility/Windows Server 2012
- **0612**—Operating System Compatibility/Windows Server 2016
- **1012**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- This Windows Installer database contains a custom action [CUSTOM_ACTION_NAME] with a hard-coded path "[CUSTOM_ACTION_TARGET]" (Table: CustomAction, Key: [CUSTOM_ACTION_KEY]).
- This Windows Installer database contains a hard-coded path "[INI_FILE_VALUE]" in INI entry (Table: IniFile, Key: [INIFILE_KEY]).
- This Windows Installer database contains a hard-coded path "[REGISTRY_VALUE]" in registry value "[REGISTRY_KEY]" (Table: Registry, Key: [REGISTRY_VALUE]).
- This Windows Installer database contains a hard-coded path "[REMOVEINIFILE_VALUE]" in table RemoveIniFile (Table: RemoveIniFile, Key: [REMOVEINIFILE_KEY]).
- This Windows Installer database contains a hard-coded path "[SERVICEINSTALL_ARGUMENTS]" in arguments of installed service [SERVICE_DISPLAY_NAME] (Table: ServiceInstall, Key: [SERVICEINSTALL_KEY]).
- This Windows Installer database contains a hard-coded path "[SERVICECONTROL_ARGUMENTS]" in arguments of controlled service [SERVICE_CONTROL_DISPLAY_NAME] (Table: ServiceControl, Key: [SERVICECONTROL_KEY]).
- This Windows Installer database contains a hard-coded path "[ENVIRONMENT_VALUE]" in environment variable [ENVIRONMENT_NAME] (Table: Environment, Key: [ENVIRONMENT_KEY]).
- This Windows Installer database contains a hard-coded path [SHORTCUT_ARGUMENTS] in arguments of shortcut [SHORTCUT_NAME] (Table: Shortcut, Key: [SHORTCUT_KEY]).

Background

Beginning with Windows Vista, the default locations for user and system data have changed. For example, user data that was previously stored in the `%SystemDrive%\Documents and Settings` directory is now stored in the `%SystemDrive%\Users` directory. For backward compatibility, the old locations have junction points that point to the new locations. For example, `C:\Documents and Settings` is now a junction point that refers to `C:\Users`.

Resolution

The following resolutions are available.

Manual Fix

Changed or obsolete junction points should be replaced with the appropriate Windows Installer property.

Basic Auto Fix

Changed or obsolete junction points are replaced with the appropriate Windows Installer properties in a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Manifest Files Using Operating System Identifier

For this operating system compatibility test, the Windows Installer database is scanned for manifest files that contain a compatibility section without a <supportedOS> tag that refers to Windows 8.

Test Group/Test Category

- **0340**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0440**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2840**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5240**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5440**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5640**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5840**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7040**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2940**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5340**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5540**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5740**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5940**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7140**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6740**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0540**—Operating System Compatibility/Windows Server 2012
- **0640**—Operating System Compatibility/Windows Server 2016
- **1040**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a manifest file [FILE_NAME] with a compatibility section that
    has no <supportedOS> tag that refers to Windows 8 (Table: File, Key: [FILE_KEY]).
```

Background

On Windows 7 and later systems, applications can specify supported operating system identifiers through their manifest files. A manifest file is a simple .xml file that contains settings that inform the operating system how to handle the program when it is launched. On Windows 7 and later systems, compatibility information for operating system support is read from the <supportedOS> tags in the compatibility section of the manifest file. The operating system chooses the highest version identifier in the manifest up to the running Windows version and gives the application support at that level. Applications without a compatibility section in their manifest file have Windows Vista behavior by default on Windows 7 and later systems. This might break visual appearance or functionality (for example, the client area of applications might be rendered without a theme in high contrast mode).

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered by updating the application manifests with the latest compatibility information for operating system support.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Maximum Version of the OS Where This App Was Tested by the Developer

For this operating system compatibility test, the application is scanned to determine the maximum version of the Windows OS where this app was tested by the developer and known to be in a working state.

Test Group/Test Category

- **3004**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **3104**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **4102**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **4502**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **4702**—Operating System Compatibility/Windows 10-2004 (32-Bit)

- **6002**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **6202**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **8002**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **4202**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **4602**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **4802**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **6102**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **6302**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **8102**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6502**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Resolution

Application should only be installed on a device with an OS version with which it has been tested.

Microsoft Management Console (MMC) Snap-ins Data Execution Prevention

For this operating system compatibility test, the Windows Installer database is scanned for the presence of Microsoft Management Console snap-in (.msc) files that are not Data Execution Prevention-aware (DEP-aware).

Test Group/Test Category

- **0309**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0409**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2809**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5209**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5409**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5609**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5809**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7009**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2909**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5309**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5509**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5709**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5909**—Operating System Compatibility/Windows 10-21H1 (64-Bit)

- **7109**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6709**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0509**—Operating System Compatibility/Windows Server 2012
- **0609**—Operating System Compatibility/Windows Server 2016
- **1009**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains Microsoft Management Console snap-in [FILE_NAME2] referencing a
    non DEP-aware library [FILE_NAME2] (Table: File, Keys: [FILE_KEY1], [FILE_KEY2]).
```

Background

On Windows XP SP2 and later systems, the DEP security feature prevents an application or a service from executing code from a non-executable memory region. Microsoft Management Console (MMC) is a common presentation service for management applications, hosting snap-ins provided by Microsoft and third-party software manufacturers. `MMC.exe` always runs with DEP enabled: the feature cannot be turned off, and no compatibility mode exists. Snap-ins that are not DEP-aware might fail to load within the MMC or might not work properly.

Resolution

The following resolutions are available.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered so that all Microsoft Management Console snap-ins (.msc) are DEP-aware.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Microsoft Management Console snap-ins (.msc) that are not DEP-aware are removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *Caution: Removed snap-ins might cause (part of) the application to not function or to function incorrectly.*

Mixed Per-User and Per-Machine Data

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of components that contain mixed per-user and per-machine content.

Test Group/Test Category

- **0345**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0445**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2845**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5245**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5445**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5645**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5845**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7045**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2945**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5345**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5545**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5745**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5945**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7145**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6745**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0545**—Operating System Compatibility/Windows Server 2012
- **0645**—Operating System Compatibility/Windows Server 2016
- **1045**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine file keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).`
- `This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user file keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).`

- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine registry keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user registry keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine directory keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user directory keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]). Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine file keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user file keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine registry keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user registry keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user content ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-machine directory keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).

- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) with per-user directory keypath [COMPONENT_KEY_PATH] (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with mixed per-machine ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) and per-user file keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) and per-user registry keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]) and per-user directory keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-user data ([PER_USER_FILES], [PER_USER_REGISTRY_ENTRIES]). Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-machine data ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-machine file keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-machine data ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-machine registry keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-machine data ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]) and per-machine directory keypath [COMPONENT_KEY_PATH]. Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).
- This Windows Installer Database contains component [COMPONENT_NAME] with per-machine data ([PER_MACHINE_FILES], [PER_MACHINE_REGISTRY_ENTRIES]). Additionally, a part of the data ([UNKNOWN_REGISTRY_ENTRIES]) may be either per-user or per-machine, depending on installation context (Table: Component, Key: [COMPONENT_NAME]).

Background

Mixing per-user and per-machine data in the same component could result in only partial installation of the component for some users in a multiuser environment.

Resolution

The following resolutions are available.

Manual Fix

Per-user and per-machine data should be moved to separate components.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Per-user content and per-machine content are moved to separate components in a Windows Installer transform.

This fix is enabled by default.

Nested SendTo Menus

For this operating system compatibility test, the Windows Installer database is scanned for the creation of shortcuts in subfolders of the SendTo folder.

Test Group/Test Category

- **0324**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0424**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2824**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5224**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5424**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5624**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5824**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7024**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2924**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5324**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5524**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5724**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5924**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7124**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6724**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

- **0524**—Operating System Compatibility/Windows Server 2012
- **0624**—Operating System Compatibility/Windows Server 2016
- **1024**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a shortcut [SHORTCUT_NAME] created in a subfolder of the SendTo
    folder (Table: Shortcut, Key: [SHORTCUT_KEY]; Table: Directory, Key: [DIRECTORY_KEY]).
```

Background

The SendTo folder contains shortcuts for possible destinations to which files and folders can be sent. Since Windows Vista, shortcuts that are placed in subfolders of the SendTo folder are not shown. Only shortcuts that are placed directly in the SendTo folder are visible in the context menu.

Resolution

The following resolutions are available.

Manual Fix

Shortcuts in subfolders of the SendTo folder should be moved directly into the SendTo folder. Empty subfolders should be removed.

Basic Auto Fix

Shortcuts in subfolders of the SendTo folder are moved directly into the SendTo folder in a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Obsolete API Calls

For this operating system compatibility test, the Windows Installer database is scanned for references to obsolete API calls. The extensions of the files that are scanned are `.exe`, `.dll`, `.sys`, `.src`, `.drv`, `.cpl`, and `.ocx`.

Test Group/Test Category

- **0323**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0423**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2823**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5223**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5423**—Operating System Compatibility/Windows 10-2004 (32-Bit)

- **5623**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5823**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7023**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2923**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5323**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5523**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5723**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5923**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7123**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6723**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0523**—Operating System Compatibility/Windows Server 2012
- **0623**—Operating System Compatibility/Windows Server 2016
- **1023**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains a reference to an obsolete API call [CALL_NAME] in file
    [FILE_NAME] (Table: File, Key: [FILE_KEY]).
```

Background

A number of API calls that were previously used in Microsoft Windows are no longer supported on Windows 7 and later systems. These functions may have been removed from corresponding DLLs, or those DLL are not available on Windows 7 and later systems. Obsolete functions cannot be called, and programs that attempt to use them may fail to launch or may not function as expected.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to avoid calling obsolete API calls. Obsolete API calls are documented in Microsoft SDK updates.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Obsolete File Associations

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the presence of obsolete file associations.

Test Group/Test Category

- **0355**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0455**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **2855**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5255**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5455**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5655**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5855**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7055**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2955**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5355**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5555**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5755**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5955**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7155**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6755**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0555**—Operating System Compatibility/Windows Server 2012
- **0655**—Operating System Compatibility/Windows Server 2016
- **1055**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains file [FILE_NAME] with not supported extension in Windows 7
    (Table: File, Key: [FILE_NAME])
```

Background

In Windows 7 and later, some file associations have been deprecated or disabled. When attempting to open a file with these extensions, users will be prompted to select another application that is installed or will be pointed to a web page that offers solutions.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should not contain files with obsolete extensions in their installers.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Operating System Version Conditions

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the usage of conditions that evaluate to false in the Windows operating system in the tables InstallExecuteSequence, MsiLockPermissionsEx, Condition, and Component.

Test Group/Test Category

- **0313**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0413**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2813**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5213**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5413**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5613**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5813**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7013**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2913**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5313**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5513**—Operating System Compatibility/Windows 10-2004 (64-Bit)

- **5713**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5913**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7113**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6713**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0513**—Operating System Compatibility/Windows Server 2012
- **0613**—Operating System Compatibility/Windows Server 2016
- **1013**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains component [COMPONENT_NAME] using condition "[COMPONENT_CONDITION]" that evaluates to false for Windows [VERSION] (Table: Component, Key: [COMPONENT_NAME]).`
- `This Windows Installer database contains custom action [InstallExecuteSequence_ACTION] using condition "[InstallExecuteSequence_CONDITION]" that evaluates to false for Windows [VERSION] (Table: CustomAction, Key: [InstallExecuteSequence_ACTION]).`
- `This Windows Installer database contains security entry [MsiLockPermissionsEx_ENTRY] in MsiLockPermissionsEx using condition [MsiLockPermissionsEx_CONDITION] that evaluates to false for Windows [VERSION] (Table: MsiLockPermissionsEx, key: [MsiLockPermissionsEx_ENTRY]).`
- `This Windows Installer database contains feature [CONDITION_FEATUREKEY] using conditional Install Level with condition "[CONDITION]" that evaluates to false for Windows [VERSION] (Table: Feature, Key: [CONDITION_FEATUREKEY]; Table: Condition, Key: [CONDITION_FEATUREKEY], [CONDITION_LEVEL]).`

Background

Windows Installer provides the built-in properties `VersionNT` and `WindowsBuild`, which can be used in conditions to determine, for a given version of the operating system, which Windows Installer features/components should be installed, which custom actions should be executed, and/or what security should be applied.

Resolution

The following resolutions are available.

Manual Fix

The tables InstallExecuteSequence, MsiLockPermissionsEx, Condition, and Component should be scanned for conditions that evaluate to false for the specified Windows operating system. If the component, feature, custom action, or security is needed, the condition should be modified so that it evaluates to true for the specified Windows operating system.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Conditions in the Windows Installer tables InstallExecuteSequence, MsiLockPermissionsEx, Condition, and Component that evaluate to false for the specified Windows operating system are replaced with a condition that evaluates to true in a Windows Installer transform.

This fix is enabled by default.

Caution • *This workaround enables the entire Windows Installer database, including anything that was not intended for the specified Windows operating system.*

Operating System Version Launch Conditions

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for launch conditions that prevent the installation from running on the specified Windows operating system.

Test Group/Test Category

- **0314**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0414**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2814**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5214**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5414**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5614**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5814**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7014**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2914**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5314**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5514**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5714**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5914**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7114**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6714**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0514**—Operating System Compatibility/Windows Server 2012
- **0614**—Operating System Compatibility/Windows Server 2016
- **1014**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains Launch Condition "[LAUNCHCONDITION]" that prevents the software
    from being installed in Windows [VERSION] (Table: LaunchConditions, Key: [LAUNCHCONDITION_KEY]).
```

Background

Launch conditions are usually evaluated in the very beginning of the installation process for a Windows Installer package. If any of these conditions evaluates to false, the installation does not take place. Launch conditions often rely on the value of the `VersionNT` or `VersionBuild` properties, which depend on the operating system version.

Resolution

The following resolutions are available.

Manual Fix

Unless the application is known to cause problems on the specified Windows operating system, launch conditions that might prevent the installation from taking place on the specified Windows operating system should be removed.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Launch conditions that prevent the installation from taking place on the specified Windows operating system are removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *This workaround enables the installation of a Windows Installer package, even if it was not intended for the specified Windows operating system.*

Quick Launch Bar

For this operating system compatibility test, the Windows Installer database is scanned for shortcuts that will be installed on the Quick Launch bar.

Test Group/Test Category

- **0325**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0425**—Operating System Compatibility/Windows 8.1(64-Bit)
- **2825**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5225**—Operating System Compatibility/Windows 10-1909 (32-Bit)

- **5425**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5625**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5825**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7025**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2925**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5325**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5525**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5725**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5925**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7125**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6725**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0525**—Operating System Compatibility/Windows Server 2012
- **0625**—Operating System Compatibility/Windows Server 2016
- **1025**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains shortcut [SHORTCUT_NAME] installed in the Quick Launch bar
    (Table: Shortcut, Key: [SHORTCUT_KEY]).
```

Background

The Quick Launch bar is a dockable toolbar that contains user-defined shortcuts to applications. Icons in this area respond to a single click. Since Windows 7, this functionality is included on the taskbar buttons (shortcuts can be pinned to the taskbar) and the Quick Launch bar is by default not available. It can be enabled, but it has compatibility issues with the Language bar. If both are enabled, the Quick Launch bar is removed after the machine is restarted.

Resolution

The following resolutions are available.

Manual Fix

Quick Launch shortcuts should be moved to another location or pinned to the taskbar. Alternatively, the Quick Launch bar can be enabled.

Caution • *Enabling the Quick Launch bar is not recommended because of possible conflicts with the Language bar.*

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Quick Launch shortcuts are removed in a Windows Installer transform. This fix is enabled by default.

Reboot Pending Launch Condition

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the absence of launch conditions that prevent the installation from continuing when a restart is pending.

Test Group/Test Category

- **0348**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0448**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2848**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5248**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5448**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5648**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5848**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7048**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2948**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5248**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5548**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5748**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5948**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7148**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6748**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0548**—Operating System Compatibility/Windows Server 2012
- **0648**—Operating System Compatibility/Windows Server 2016
- **1048**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database does not contain any LaunchCondition that prevent the installation when
    system reboot is pending (Table: LaunchCondition).
```

Background

The installation sets the value of the `MsiSystemRebootPending` property to 1 if there is an operation pending to rename a file. Package authors can base a condition in the LaunchCondition table on this property to prevent the installation of their package in cases where there is an operation pending to rename a file. This may prevent a restart of the operating system caused by the renaming of the file. Any installation that explicitly uses the `MsiSystemRebootPending` property in the LaunchCondition table may not continue when there are pending operations that require a system reboot.

Resolution

The following resolutions are available.

Manual Fix

The following condition should be added to the LaunchCondition table to prevent the installation of the package if a system reboot is pending and required.

```
NOT MsiSystemRebootPending
```

Basic Auto Fix

The following condition is added through a Windows Installer transform.

```
MsiSystemRebootPending <> 1
```

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Reorganized Start Screen

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for shortcuts to non-executable files in the Start Menu folder. Additionally, the database is scanned for shortcuts that are located in a subfolder of the Start Menu.

Test Group/Test Category

- **0343**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0443**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2843**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5243**—Operating System Compatibility/Windows 10-1909 (32-Bit)

- **5443**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5643**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5843**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7043**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2943**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5343**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5543**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5743**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5943**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7143**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6743**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0543**—Operating System Compatibility/Windows Server 2012
- **0643**—Operating System Compatibility/Windows Server 2016
- **1043**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database contains a shortcut [SHORTCUT_NAME] to a non-executable file [FILE_NAME] which might not be pinned to the Start screen (Table: Shortcut, Key: [SHORTCUT_KEY]).`
- `This Windows Installer database contains a shortcut [SHORTCUT_NAME] in a subfolder of the "Start Menu\Programs" folder which might not be displayed correctly in the All Apps view (Table: Shortcut, Key: [SHORTCUT_KEY]).`

Background

On Windows 8 and later systems, the Start Menu is no longer available, and its functionality has been replaced with a new Start screen. The appearance and functionality of the new solution might result in an ambiguous shortcut structure. Windows 8 and later automatically pins shortcuts to executable files to the new Start screen. However, it does not pin shortcuts for other file types (for example, text files, help files, and command files [`.bat`, `.cmd`]). Note that the shortcuts are still visible when the user browses to the All Apps applet; this is the equivalent of the “All Applications” on the old Start Menu.

Furthermore, on Windows 8 and later systems, the tree hierarchy from the old Start Menu is no longer available. Shortcuts in the All Apps applet are grouped by the root subfolders in the Start Menu folder. Shortcuts from subfolders are displayed in one group with no visual clue to which group it belongs. Hence, if two shortcuts with the same name exist in different subfolders, users might be unable to distinguish between them. This behavior might limit productivity.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

Shortcuts should be renamed to unequivocally express their behavior and affiliation. For example, instead of using generic names like **Readme** or **Help documentation**, a more specific name like **Readme for <application name>** or **Help documentation for <application name>** should be used. Shortcuts to non-executable files should be manually pinned by the logged-on user.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Restart Manager FilesInUse Dialog

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for the absence of the Restart Manager FilesInUse dialog (MsiRMFilesInUse) and the `MSIRESTARTMANAGERCONTROL` property value that disables Restart Manager.

Test Group/Test Category

- **0346**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0446**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2846**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5246**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5446**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5646**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5846**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7046**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2946**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5346**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5546**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5746**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5946**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7146**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6746**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0546**—Operating System Compatibility/Windows Server 2012

- **0646**—Operating System Compatibility/Windows Server 2016
- **1046**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Messages

- `This Windows Installer database does not contain definition of Restart Manager Files in Use (MsiRMFilesInUse) dialog to handle Restart Manager on Windows 7 (Table: Dialog).`
- `This Windows Installer database contains Restart Manager Files in Use (MsiRMFilesInUse) dialog suppressed by property MSIRESTARTMANAGERCONTROL (the current value is [PROPERTY_VALUE]) (Table: Property, Key: MSIRESTARTMANAGERCONTROL).`

Background

The MsiRMFilesInUse dialog can be authored to display a list of processes that are currently running files that need to be overwritten or deleted by the installation. The dialog contains two options to allow end users to specify how to proceed:

- Users can choose to have the installation close the applications that are using those files and then attempt to restart the applications after the installation is complete.
- Users can avoid closing the applications. A reboot will be required at the end of the installation.

If the user selects the first option, a push button control on this dialog can be authored to publish the RMShutdownAndRestart control event, and the Restart Manager can close the applications and restart them at the end of the installation. This can eliminate or reduce the need to restart the computer. The MsiRMFilesInUse dialog is displayed during an installation only if installation is running with full user interface and the MsiRMFilesInUse dialog is present in Dialog table. Additionally, the public property `MSIRESTARTMANAGERCONTROL` can be used to disable Restart Manager.

Resolution

The following resolutions are available.

Manual Fix

The MsiRMFilesInUse dialog should be added to the Dialog table of the Windows Installer database.

Basic Auto Fix

The MsiRMFilesInUse dialog is enabled through a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

rundll32 Calls (User Account Control)

For this operating system compatibility test, the Windows Installer database is scanned for references to `rundll32.exe`.

Test Group/Test Category

- **0311**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0411**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2811**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5211**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5411**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5611**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5811**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7011**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2911**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5311**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5511**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5711**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5911**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7111**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6711**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0511**—Operating System Compatibility/Windows Server 2012
- **0611**—Operating System Compatibility/Windows Server 2016
- **1011**—Operating System Compatibility/Windows Server 2019

Severity

Error

Messages

- `This Windows Installer database contains a custom action calling rundll32.exe (Table:CustomAction, Key: [CUSTOM_ACTION_KEY]).`
- `This Windows Installer database contains a system startup registry entry calling rundll32.exe (Table: Registry, Key: [REGISTRY_ENTRY]).`
- `This Windows Installer database contains a shortcut to rundll32.exe (Table:Shortcut, Key: [SHORTCUT_KEY]).`
- `This Windows Installer database contains a reference to rundll32.exe in Windows Installer property (Table:Property, Key: [PROPERTY]).`

Background

The Windows tool `rundll32.exe` is used to run executable code in DLL files as if it were called by an application. On Windows Vista and later systems, User Account Control (UAC) can cause problems for solutions that use `rundll32.exe`. When an application that relies on `rundll32.exe` needs elevated privileges to perform some global tasks, it is `rundll32.exe` (rather than the application) that requests the UAC elevation prompt. As a result, the user sees a request from `Windows host process (rundll32)`. Without a clear description and icon for the application that is requesting elevation, users have no way to identify the application and determine whether it is safe to elevate it. Any DLL that runs under `rundll32.exe` and that needs elevation should be modified into an executable file that has its elevation level set in its manifest.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

A "Run DLL as an app" DLL call should be wrapped in a separate executable file. Additionally, a manifest file for this executable file should be included with the required elevated privileges setting.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Self-Update Functionality (User Account Control)

For this operating system compatibility test, the Windows Installer database is scanned for the presence of unmanifested executable files that are recognized as installations, upgrades, or patches. Heuristic analysis scans files that match any of the following criteria: `*update*.exe`, `*setup*.exe`,`*install*.exe`, `*unins*.exe`, or `*patch*.exe` for their User Account Control (UAC) awareness.

Test Group/Test Category

- **0319**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0419**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2819**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5219**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5419**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5619**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5819**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7019**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2919**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5319**—Operating System Compatibility/Windows 10-1909 (64-Bit)

- **5519**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5719**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5919**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7119**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6719**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0519**—Operating System Compatibility/Windows Server 2012
- **0619**—Operating System Compatibility/Windows Server 2016
- **1019**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains self-update functionality in file [FILE_NAME] (Table: File,
    Key: [FILE_KEY]).
```

Background

Some software has a built-in mechanism to automatically update itself. Self-updating should be avoided in a managed environment due to lack of control over managed software and privilege issues. Furthermore, the self-update functionality might leave old files behind or cause issues when the software is being removed. Since Windows Vista, installation and update programs are recognized and, if UAC is enabled, cause prompts for credentials. This is done to prevent installations without the user's knowledge and approval.

Resolution

The following resolutions are available.

Manual Fix

Self-update functionality of the software should be disabled.

Basic Auto Fix

A manifest file is added for each unmanifested installation or upgrade in a Windows Installer transform. The content of the manifest depends on whether the executable file is UAC aware. If the executable file is UAC aware, the manifest file sets the privilege level to requireAdministrator. If the executable is not UAC aware, the manifest file sets the privilege level to `asInvoker`.

This fix is enabled by default.

Advanced Auto Fix

Unmanifested executable files that matching the following patterns are removed in a Windows Installer transform: `*update*.exe`, `*setup*.exe`,`*install*.exe`, `*unins*.exe` or `*patch*.exe`.

Caution • *This might have a high negative impact on application functionality.*

Standard User Changes (User Account Control)

For this operating system compatibility test, the Windows Installer database is scanned for the presence of `.exe` files (other than installations and upgrades) that cause the User Account Control (UAC) prompt to be displayed.

Test Group/Test Category

- **0320**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0420**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2820**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5220**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5420**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5620**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5820**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7020**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2920**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5320**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5520**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5720**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5920**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7120**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6720**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0520**—Operating System Compatibility/Windows Server 2012
- **0620**—Operating System Compatibility/Windows Server 2016
- **1020**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains an unmanifested file [FILE_NAME] (Table: File, Key:
    [FILE_KEY]).
```

Background

On Windows Vista and later systems, all applications are run by default with standard user privileges, even when the logged-on user is a member of the Administrators group. If an application requires elevated privileges, an accompanying manifest file can indicate this. At run time, Windows confirms that the privilege elevation that is declared in the manifest aligns with the user's intention by displaying a UAC prompt for consent or credentials. Unmanifested executable files do not trigger a UAC prompt, and any actions that require elevated privileges—including any changes to system or global settings—silently fail. Therefore, unmanifested applications that require elevated privileges might not function properly.

Resolution

The following resolutions are available.

Manual Fix

For each unmanifested executable file, create a manifest file that sets the required privilege level. For applications that do not require administrative privileges, the required privilege level should be set to asInvoker. (That is, the UAC prompt is not shown, and permissions are not elevated.) Otherwise, the required privilege level should be set to either requireAdministrator or highestAvailable. If an application seeks an unsuited privilege level (and the manifest cannot be corrected), you can create a shim database specify the desired privilege level.

Basic Auto Fix

A manifest file is added in a Windows Installer transform to each application that requires administrative privileges but does not already have an embedded or associated manifest file. The content of the manifest depends on whether a given executable file is UAC aware. If the executable file is UAC aware, the manifest file sets the privilege level to requireAdministrator. If the executable is not UAC aware, the manifest file sets the privilege level to `asInvoker`.

This fix is enabled by default.

Advanced Auto Fix

No resolution is available.

Unmanifested Control Panel (.cpl) Files (User Account Control)

For this operating system compatibility test, the Windows Installer database is scanned for the presence of unmanifested Control Panel (`.cpl`) files.

Test Group/Test Category

- **0302**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0402**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2802**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5202**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5402**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5602**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5802**—Operating System Compatibility/Windows 10-21H1 (32-Bit)

- **7002**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2902**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5302**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5502**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5702**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5902**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7102**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6702**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0502**—Operating System Compatibility/Windows Server 2012
- **0602**—Operating System Compatibility/Windows Server 2016
- **1002**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains unmanifested Control Panel file [FILE_NAME] (Table: File, Key:
    [FILE_KEY]).
```

Background

On Windows Vista and later systems, all applications are run by default with standard user privileges (even when the logged-on user is a member of the Administrators group). As a result, unmanifested Control Panel (.cpl) files might fail. A manifest file is a simple .xml file that contains settings that inform the operating system how to handle the program when it is launched.

Resolution

The following resolutions are available.

Manual Fix

Control Panel (`.cpl`) files should be embedded in .exe files that include a manifest that specifies the privilege level that is required to execute the application. Where this is not feasible, an external manifest file can be created. In the latter case, the manifest file must be located in the same folder with the `.cpl` file and named the same as the full file name of the `.cpl` file, with a `.manifest` extension (for example, `<application_name>.cpl.manifest`).

Basic Auto Fix

For each unmanifested Control Panel (`.cpl`) file, a manifest file that specifies privilege level `highestAvailable` is added in a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

For each unmanifested Control Panel (`.cpl`) file, a manifest file that specifies privilege `requireAdministrator` is added in a Windows Installer transform.

Unmanifested Control Panel Applications (User Account Control)

For this operating system compatibility test, the Windows Installer database is scanned for the presence of unmanifested Control Panel Applications. The file extensions that are scanned are `.exe` and `.dll`.

Test Group/Test Category

- **0303**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0403**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **0503**—Operating System Compatibility/Windows Server 2012
- **0603**—Operating System Compatibility/Windows Server 2016
- **1003**—Operating System Compatibility/Windows Server 2019
- **2803**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5203**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5403**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5603**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5803**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7003**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2903**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5303**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5503**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5703**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5903**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7103**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6703**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Warning

Message

```
This Windows Installer database contains unmanifested Control Panel Application [FILE_NAME] (Table:
    File, Key: [FILE_KEY]).
```

Background

On Windows Vista and later systems, all applications are run by default with standard user privileges, even when the logged-on user is a member of the Administrators group. As a result, unmanifested Control Panel (`.cpl`) files might fail. A manifest file is a simple `.xml` file that contains settings that inform the operating system how to handle the program when it is launched.

Resolution

The following resolutions are available.

Manual Fix

An external manifest file should be created and included in the same folder with the Control Panel Application file and named the same as the full file name of the executable with a `.manifest` extension (for example `<application_name.extension>.manifest`).

Basic Auto Fix

For each unmanifested Control Panel Application file, a manifest file that specifies privilege level `highestAvailable` is added in a Windows Installer transform.

This fix is enabled by default.

Advanced Auto Fix

For each unmanifested Control Panel Application file, a manifest file that specifies privilege level `requireAdministrator` is added in a Windows Installer transform.

Unsigned Drivers

For this operating system compatibility test, the Windows Installer database is scanned for the presence of unsigned drivers.

Test Group/Test Category

- **0321**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0421**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2821**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5221**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5421**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5621**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5821**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7021**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2921**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5321**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5521**—Operating System Compatibility/Windows 10-2004 (64-Bit)

- **5721**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5921**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7121**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6721**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0521**—Operating System Compatibility/Windows Server 2012
- **0621**—Operating System Compatibility/Windows Server 2016
- **1021**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains a driver ([FILE_NAME]) which is not correctly signed (Table:
    File, Key: [FILE_KEY]).
```

Background

A signed device driver includes a digital signature (an electronic security mark that can indicate the manufacturer of the software, as well as validate the original contents of the driver package). If a driver has been signed by a manufacturer that has verified its identity with a certification authority, it is confirmed that the driver actually comes from that publisher and has not been altered. If a user tries to install an unsigned driver, Windows displays a warning and prompts the user.

Resolution

The following resolutions are available.

Manual Fix

A signed version of the driver should be delivered by its manufacturer. Alternatively, Windows Driver Kit (WDK) from Microsoft can be used to sign the driver with a trusted certificate.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Unsigned drivers are removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *This might have a high negative impact on application functionality.*

Unsigned Executables

For this operating system compatibility test, the Windows Installer database is scanned for the presence of unsigned executables. Scanned file extensions are: `.exe`, `.dll`, `.ocx`, and `.cab`.

Test Group/Test Category

- **0352**—Operating System Compatibility/Windows 8.1 (32-bit)
- **0452**—Operating System Compatibility/Windows 8.1 (64-bit)
- **2852**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5252**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5452**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5652**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5852**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7052**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2952**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5352**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5552**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5752**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5952**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7152**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6752**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0552**—Operating System Compatibility/Windows Server 2012
- **0652**—Operating System Compatibility/Windows Server 2016
- **1052**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains unsigned executable [EXECUTABLE_NAME] (Table: File,
    Key: [FILE_NAME]).
```

Background

According to Microsoft best practices, all binaries should be digitally signed with a certificate issued by a Trusted Publisher. The Trusted Publisher's certificate store contains information about the Authenticode (signing) certificates of Trusted Publishers that are installed on a computer. Unsigned executables will prompt the user for authorization.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

The application with executables signed by a Trusted Publisher should be delivered by its manufacturer.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Unsigned Windows Installer Database

Note • *This test is not applicable to App-V packages.*

For this operating system compatibility test, the Windows Installer database is scanned for signing with trusted certificate.

Test Group/Test Category

- **0353**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0453**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2853**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5253**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5453**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5653**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5853**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7053**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2953**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5353**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5553**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5753**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5953**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7153**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6753**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0553**—Operating System Compatibility/Windows Server 2012
- **0653**—Operating System Compatibility/Windows Server 2016

- **1053**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

`This Windows Installer database is not signed with certificate from a trusted Certificate Authority.`

Background

All Windows Installer databases should be digitally signed with a certificate issued by a Trusted Publisher. Unsigned Windows Installer databases will prompt the user for authorization.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

The Windows Installer database should be digitally signed with a certified issues by a Trusted Publisher.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Unsupported .NET Framework 1.0/1.1 Applications

For this operating system compatibility test, the Windows Installer database is scanned for the presence of files that contain references to .NET Framework 1.0 or 1.1 in the header. The extensions of files that are scanned are `.exe`, `.dll`, `.sys`, `.src`, `.drv`, `.cpl`, and `.ocx`.

Test Group/Test Category

- **0335**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0435**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2835**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5235**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5435**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5635**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5835**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7035**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2935**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)

- **5335**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5535**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5735**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5935**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7135**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6735**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0535**—Operating System Compatibility/Windows Server 2012
- **0635**—Operating System Compatibility/Windows Server 2016
- **1035**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains application [FILE_NAME] dependent on .NET Framework Version
    [VERSION] (Table: File, Key: [FILE_KEY]).
```

Background

Microsoft .NET Framework 1.0 and 1.1 are not supported on Windows 7 and later systems. Although it may be possible to install .NET Framework 1.0 or 1.1 components on Windows 7, Microsoft provides no level of support for these configurations.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to use a more recent version of .NET Framework.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Unsupported 16-Bit Files

For this operating system compatibility test, the Windows Installer database is scanned for the presence of components that contain 16-bit files without conditions that disable them for 64-bit Windows systems. The file extensions that are scanned are `.exe`, `.dll`, `.sys`, `.drv`, `.ocx`, `.cpl`, and `.src`.

Test Group/Test Category

- **0417**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **0517**—Operating System Compatibility/Windows Server 2012
- **0618**—Operating System Compatibility/Windows Server 2016
- **1018**—Operating System Compatibility/Windows Server 2019
- **0617**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **2817**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5217**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5417**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5617**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5817**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7017**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2917**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5317**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5517**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5717**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5917**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7117**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6717**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Error

Message

```
This Windows Installer database contains 16-bit file [FILE_NAME] which might be installed in 64-bit
    systems (Table: File, key: [FILE_KEY]).
```

Background

Since the introduction of 64-bit Windows systems, 16-bit code is no longer supported. If the launch conditions are missing or incorrect, 16-bit files might be installed on 64-bit Windows Server 2008 R2 systems. If a user attempts to launch such a file, they encounter an error message stating that the file is not a valid Win32 application.

Resolution

The following resolutions are available.

Manual Fix

A Windows Server 2008 R2-compatible application should be delivered by its manufacturer. Self-developed applications should be re-engineered by replacing 16-bit code with the appropriate 32- or 64-bit code.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

The 16-bit files that are configured to be installed on 64-bit systems are moved to separate components with conditions that enable them only for 32-bit systems; this is done in a Windows Installer transform.

This fix is enabled by default.

Caution • *On 64-bit systems, those files will not be installed.*

Unsupported 32-Bit Windows Help Files

For this operating system compatibility test, the Windows Installer database is scanned for the presence of 32-bit Windows Help files (`.hlp`).

Test Group/Test Category

- **0301**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0401**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2801**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5201**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5401**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5601**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5801**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7001**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2901**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5301**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5501**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5701**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5901**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7101**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6701**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0501**—Operating System Compatibility/Windows Server 2012

- **0601**—Operating System Compatibility/Windows Server 2016
- **1001**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

`[PACKAGE_NAME] (Table: File, Key: [FILE_KEY]).`

Background

Windows Vista and later do not support 32-bit Windows Help files (`.hlp`), now superseded by HTML Help (`.chm`). Users who try to open `.hlp` files see an error message instead of the expected Help. Note that support for viewing 16-bit `.hlp` files remains available in Windows 7.

Resolution

The following resolutions are available.

Manual Fix

Windows Help (`.hlp`) files should be converted to the Microsoft HTML Help (`.chm`) format. Where the conversion is not feasible, Microsoft supplies a downloadable version of the executable for 32-bit `.hlp` files, available from update KB917607.

Basic Auto Fix

The Windows Help browser (`WinHlp32.exe`) and necessary file associations are added in a Windows Installer transform via a Merge Module.

This fix is enabled by default.

Advanced Auto Fix

The Windows Help browser (`WinHlp32.exe`) for Windows 7 (update KB917607) is added in a Windows Installer transform via a Merge Module.

Unsupported DHTML Editing Control

For this operating system compatibility test, the Windows Installer database is scanned for the use of DHTML Editing Control functionality. The extensions of the files that are scanned are `.exe`, `.dll`, and `.ocx`.

Test Group/Test Category

- **0308**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0408**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2808**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5208**—Operating System Compatibility/Windows 10-1909 (32-Bit)

- **5408**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5608**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5808**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7008**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2908**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5308**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5508**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5708**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5908**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7108**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6708**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0508**—Operating System Compatibility/Windows Server 2012
- **0608**—Operating System Compatibility/Windows Server 2016
- **1008**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains executable [FILE_NAME] requiring the DHTML Editing Control for
    Applications (Table: File, Key: [FILE_KEY]).
```

Background

The DHTML Editing Control, which Microsoft originally released in 1998, is an ActiveX control designed for WYSIWYG HTML editing in Web pages and Windows-based applications. Windows Vista and later systems do not support this control because of security reasons. Software that incorporates the DHTML Editing Control for Applications no longer functions as intended on Windows 7 systems. For example, Delphi applications may cause unhandled exceptions and Visual Basic applications may display the following message when they are opened or when the form that contains the control is instantiated: `"Component 'dhtmled.ocx' or one of its dependencies is not registered correctly: a file is missing or invalid."`

Resolution

The following resolutions are available.

Manual Fix

Applications that require the DHTML Editing Control should use a different WYSIWYG HTML editor. Where this is not feasible, Microsoft provides a downloadable original control in a signed Windows Installer package called `DHTMLEd.msi`.

Caution • *Since this workaround leaves applications unchanged, they remain exposed to the same security risks that Microsoft originally identified.*

Basic Auto Fix

The contents of the `DHTMLEd.msi` from Microsoft are added in a Windows Installer transform via a Merge Module.

This fix is enabled by default.

Caution • *Since this workaround leaves applications unchanged, they remain exposed to the same security risks that Microsoft originally identified.*

Advanced Auto Fix

No resolution is available.

Unsupported GINA Functionality

For this operating system compatibility test, the Windows Installer database is scanned for the presence of any custom GINA DLL references.

Test Group/Test Category

- **0330**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0430**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2830**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5230**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5430**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5630**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5830**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7030**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2930**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5330**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5530**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5730**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5930**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7130**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6730**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0530**—Operating System Compatibility/Windows Server 2012

- **0630**—Operating System Compatibility/Windows Server 2016
- **1030**—Operating System Compatibility/Windows Server 2019

Severity

Error

Message

```
This Windows Installer database contains unsupported customized GINA functionality (Table: Registry,
    Key: [REGISTRY_KEY]).
```

Background

Microsoft changed the interactive logon process in Windows Vista. On earlier systems, where software required a logon to a third-party server or a logon using a third-party device, the supplier had to replace the built-in Windows library `MSGina.dll` with a custom DLL. The new authentication model on Windows Vista and later systems removes GINA functionality (including customization). Software that uses the original or customized GINA functionality does not work on Windows Vista or later systems.

Resolution

The following resolutions are available.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to support the Credential Providers model as described by Microsoft.

Basic Auto Fix

No basic fix is available.

Advanced Auto Fix

Registry value `HKLM\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\GinaDLL` is removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *If this workaround is applied to software using a customized logon through a modified GINA module, it is possible that the installation might fail, and highly likely that users might not be able to log on.*

Windows Desktop Gadgets

For this operating system compatibility test, the Windows Installer database is scanned for the presence of Windows Desktop Gadgets.

Test Group/Test Category

- **0354**—Operating System Compatibility/Windows 8.1 (32-Bit)

- **0454**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2854**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5254**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5454**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5654**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5854**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7054**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2954**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5354**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5554**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5754**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5954**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7154**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6754**—Operating System Compatibility/Windows 11-21H2 (64-Bit)

Severity

Warning

Message

```
This Windows Installer database contains a Windows Desktop Gadget.
```

Background

Since Windows 8, Microsoft has deprecated Desktop Gadgets and outclassed them by new live tiles and apps. The main reasons for the deprecation are security risks and the outdated look of Desktop Gadgets.

Resolution

The following resolutions are available. Note that this issue is not resolved automatically by default.

Manual Fix

Windows 8 compatible application should be delivered by its manufacturer. Self-developed applications should not contain Desktop Gadgets in their installers.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

No resolution is available.

Windows Internet Explorer Protected Mode

For this operating system compatibility test, the Windows Installer database is scanned for the presence of registry entries that turn off Protected Mode.

Test Group/Test Category

- **0310**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0410**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2810**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5210**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5410**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5610**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5810**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7010**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2910**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5310**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5510**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5710**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5910**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7110**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6710**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0510**—Operating System Compatibility/Windows Server 2012
- **0610**—Operating System Compatibility/Windows Server 2016
- **1010**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database attempts to turn off Windows Internet Explorer Protected Mode (Table:
    Registry, Key: [REGISTRY_ENTRY]).
```

Background

On Windows Vista and later systems, Internet Explorer runs by default in Protected Mode. By preventing unauthorized access to sensitive areas of a user's system, Protected Mode limits the amount of damage that a compromised Internet Explorer process or malware can cause. As a result, applications that use Internet Explorer cannot write directly to the disk while in the Internet or Intranet zones. In addition to displaying a warning message when web pages try to write to protected areas, Internet Explorer also informs the user when web pages try to run certain software programs.

Resolution

The following resolutions are available.

Manual Fix

An application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to correctly handle Internet Explorer Protected Mode. When this is not feasible, the needed web sites should be added to the list of trusted sites.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

Registry entries that are responsible for turning off the Internet Explorer Protected Mode are removed in a Windows Installer transform.

This fix is enabled by default.

Note • *An additional manual action is needed: the web sites should be added to the list of trusted sites.*

Windows Resource Protection Files

For this operating system compatibility test, the Windows Installer database is scanned for the presence of files that are subject to Windows Resource Protection (WRP).

Test Group/Test Category

- **0315**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0415**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2815**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5215**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5415**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5615**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5815**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7015**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2915**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5315**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5515**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5715**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5915**—Operating System Compatibility/Windows 10-21H1 (64-Bit)

- **7115**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6715**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0515**—Operating System Compatibility/Windows Server 2012
- **0516**—Operating System Compatibility/Windows Server 2016
- **1015**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains Windows Resource Protection file [FILE_NAME] (Table: File, Key:
    [FILE_KEY]).
```

Background

On Windows Vista and later systems, WRP prevents the modification of essential system files, folders, and registry keys that are installed as part of the operating system. Protecting these resources prevents application and operating system failures. Accordingly, Windows Installer automatically and silently ignores attempts to write or modify a protected resource. If the application was installed with Windows Installer and logging was enabled, a warning might be logged for each file operation that was ignored because of WRP. WRP files can be installed or updated only using Microsoft-provided redistributable packages that are designed for Windows 7.

Resolution

The following resolutions are available.

Manual Fix

Affected files should be assessed whether they are required for the application to run successfully on Windows 7 systems. If the file is required, an application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to respect WRP restrictions.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

WRP files are removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *If the application relies on particular settings in the now protected area, this workaround might still result in run-time errors.)*

Windows Resource Protection Registry Keys

For this operating system compatibility test, the Windows Installer database is scanned for the presence of registry entries that are subject to Windows Resource Protection (WRP).

Test Group/Test Category

- **0316**—Operating System Compatibility/Windows 8.1 (32-Bit)
- **0416**—Operating System Compatibility/Windows 8.1 (64-Bit)
- **2816**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (32-Bit)
- **5216**—Operating System Compatibility/Windows 10-1909 (32-Bit)
- **5416**—Operating System Compatibility/Windows 10-2004 (32-Bit)
- **5616**—Operating System Compatibility/Windows 10-20H2 (32-Bit)
- **5816**—Operating System Compatibility/Windows 10-21H1 (32-Bit)
- **7016**—Operating System Compatibility/Windows 10-21H2 (32-Bit)
- **2916**—Operating System Compatibility/Windows 10-1809 (and 2019 LTSC) (64-Bit)
- **5316**—Operating System Compatibility/Windows 10-1909 (64-Bit)
- **5516**—Operating System Compatibility/Windows 10-2004 (64-Bit)
- **5716**—Operating System Compatibility/Windows 10-20H2 (64-Bit)
- **5916**—Operating System Compatibility/Windows 10-21H1 (64-Bit)
- **7116**—Operating System Compatibility/Windows 10-21H2 (64-Bit)
- **6716**—Operating System Compatibility/Windows 11-21H2 (64-Bit)
- **0516**—Operating System Compatibility/Windows Server 2012
- **0616**—Operating System Compatibility/Windows Server 2016
- **1016**—Operating System Compatibility/Windows Server 2019

Severity

Warning

Message

```
This Windows Installer database contains Windows Resource Protection registry entry [REGISTRY_KEY]
    (Table: Registry, Key: [REGISTRY_ENTRY]).
```

Background

On Windows Vista and later systems, WRP prevents the modification of essential system files, folders, and registry keys that are installed as part of the operating system. Protecting these resources prevents application and operating system failures. Accordingly, Windows Installer automatically and silently ignores attempts to write or modify a protected resource. If the application was installed with Windows Installer and logging was enabled, a warning might be logged for each registry key write operation that was ignored because of WRP. Starting with Windows 7, several additional registry

keys are protected via Windows WRP. To preserve the installation process, Windows Installer might report success in changing these keys, even though the operation failed. If the application relies on particular settings in the now protected area, this strategy might result in run-time errors. WRP registry entries can be written only using Microsoft-provided redistributable packages that are designed for Windows 7 or later.

Resolution

The following resolutions are available.

Manual Fix

Affected registry entries should be assessed whether they are required for the application to run successfully on the specified Windows systems. If the registry entry is required, an application compatible with the specified Windows operating system should be delivered by its manufacturer. Self-developed applications should be re-engineered to respect WRP restrictions.

Basic Auto Fix

No resolution is available.

Advanced Auto Fix

WRP registry keys are removed in a Windows Installer transform.

This fix is enabled by default.

Caution • *If the application relies on particular settings in the now protected area, this workaround might still result in run-time errors.*

Windows Phone Tests

The Windows Phone category consists of the following Operating System Compatibility tests:

- Application Requires Specific Minimum OS Version (M3001 / M3003 / M3101)

Application Requires Specific Minimum OS Version (M3001 / M3003 / M3101)

For this operating system compatibility test, the mobile app is scanned to determine if it requires a specific minimum OS version.

Test Group/Test Category

- **M3001**—Operating System Compatibility/Windows Phone 8
- **M3003**—Operating System Compatibility/Windows Phone 8.1
- **M3101**—Operating System Compatibility/Windows Phone 10

Severity

Error

Resolution

Application should only be installed on a device with a compatible OS version.

Application Requires VCLibs 11.0 (M3005)

For this operating system compatibility test, the mobile application is scanned to determine if it requires version VCLibs 11.0 or higher installed on Windows Phone 8.

Test Group/Test Category

- **M3005**—Operating System Compatibility/Windows Phone 8

Severity

Error

Resolution

Application should only be installed on a device where VCLibs 11.0 is installed.

Application Requires VCLibs 12.0 (M3007, M3107)

For this operating system compatibility test, the mobile application is scanned to determine if it requires version VCLibs 12.0 or higher installed on Windows Phone 8.1 or Windows Phone 10.

Test Group/Test Category

- **M3007**—Operating System Compatibility/Windows Phone 8
- **M3107**—Operating System Compatibility/Windows Phone 10

Severity

Error

Resolution

Application should only be installed on a device where VCLibs 12.0 is installed.

Application Requires WinJS 1.0 (M3006)

For this operating system compatibility test, the mobile application is scanned to determine if it requires version WinJS 1.0 or higher installed on Windows Phone 8.

Test Group/Test Category

- **M3006**—Operating System Compatibility/Windows Phone 8

Severity

Error

Resolution

Application should only be installed on a device where WinJS 1.0 is installed.

Application Requires WinJS 2.0 or Higher (M3008 / M3108)

For this operating system compatibility test, the mobile application is scanned to determine if it requires version WinJS 2.0 or higher installed on Windows Phone 8.1 or Windows Phone 10.

Test Group/Test Category

- **M3008**—Operating System Compatibility/Windows Phone 8
- **M3108**—Operating System Compatibility/Windows Phone 10

Severity

Error

Resolution

Application should only be installed on a device where WinJS 2.0 or higher is installed.

Maximum Version of the OS Where This App Was Tested by the Developer (M3002 / M3004 / M3102)

For this operating system compatibility test, the application is scanned to determine the maximum version of the operating system where this app was tested by the developer and known to be in a working state.

Test Group/Test Category

- **M3002**—Operating System Compatibility/Windows Phone 8
- **M3004**—Operating System Compatibility/Windows Phone 8.1
- **M3102**—Operating System Compatibility/Windows Phone 10

Severity

Error

Resolution

Application should only be installed on a device with an OS version with which it has been tested.

Apple iOS Tests

The Apple iOS categories consist of the following Operating System Compatibility tests:

- Application Requires Specific Minimum OS Version (M301 / M401 / M501 / M1001 / M1101 / M1301 / M1401 / M1501 / M1601)

Application Requires Specific Minimum OS Version (M301 / M401 / M501 / M1001 / M1101 / M1301 / M1401 / M1501 / M1601)

For this operating system compatibility test, the mobile app is scanned to determine if it requires a specific minimum OS version.

Test Group/Test Category

- **M301**—Operating System Compatibility/Apple/Mobile/32-bit/iOS 6 (32-bit)
- **M401**—Operating System Compatibility/Apple/Mobile/32-bit/iOS 7 (32-bit)
- **M501**—Operating System Compatibility/Apple/Mobile/64-bit/iOS 7 (64-bit)
- **M1001**—Operating System Compatibility/Apple/Mobile/32-bit/iOS 8 (32-bit)
- **M1101**—Operating System Compatibility/Apple/Mobile/64-bit/iOS 8 (64-bit)
- **M1301**—Operating System Compatibility/Apple/Mobile/32-bit/iOS 9 (32-bit)
- **M1401**—Operating System Compatibility/Apple/Mobile/64-bit/iOS 9 (64-bit)
- **M1501**—Operating System Compatibility/Apple/Mobile/32-bit/iOS 10 (32-bit)
- **M1601**—Operating System Compatibility/Apple/Mobile/64-bit/iOS 10 (64-bit)

Severity

Error

Resolution

Application should only be installed on a device with a compatible OS version.

Apple OS X Tests

The Apple OS X categories consist of the following Operating System Compatibility tests:

- Application Requires Specific Minimum OS Version (MAC003 / MAC103)
- Application Requires 64-bit Processor (MAC005 / MAC105)
- Deprecated APIs (MAC004 / MAC104)
- Deprecated Frameworks (MAC002 /MAC102)
- Deprecated Property List Keys (MAC001 / MAC101)

- Removed APIs (MAC007 / MAC107)
- Removed Frameworks (MAC006 / MAC106)

Application Requires Specific Minimum OS Version (MAC003 / MAC103)

For this operating system compatibility test, the macOS application is scanned to determine if it requires a specific minimum OS version to run.

Test Group/Test Category

- **MAC003**—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- **MAC103**—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Error

Resolution

Application should only be installed on a device with a compatible OS version.

Application Requires 64-bit Processor (MAC005 / MAC105)

For this operating system compatibility test, the macOS application is scanned to determine if it requires 64-bit architecture.

Test Group/Test Category

- **MAC005**—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- **MAC105**—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Warning

Deprecated APIs (MAC004 / MAC104)

For this operating system compatibility test, the macOS application's `.plist` file is scanned for references to deprecated functions.

Apple publishes a list of deprecated and removed frameworks for each major OS update. The highest priority would be finding applications that rely on a "removed" framework. In OS X, when an API is deprecated, a warning is generated to inform the developer that the API will be removed in a future update, but the application will generally still work in the current OS X release. Applications that use deprecated frameworks should be identified so that you can look for an update from the application's developer in the near future.

Test Group/Test Category

- MAC004—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- MAC104—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Deprecated APIs generate a Warning.

Obsolete APIs generate an Error.

Deprecated Frameworks (MAC002 /MAC102)

For this operating system compatibility test, the macOS application is scanned to determine if it contains references to a deprecated framework.

Similar to the Windows “shim” database, macOS has a list of applications that are known to have compatibility issues. OS X will either prevent them from installing, or prevent them from running if they were installed prior to the OS upgrade. When you upgrade your Mac to OS X, or when you migrate your content to a new Mac, software that is known to be incompatible with the new version of OS X is set aside and won’t run on your updated system.

***Note** • During the upgrade or migration process, OS X creates an Incompatible Software folder at the top level startup drive of your Mac. Software known to be incompatible with the new version of OS X is placed in this folder. You can look in this folder to see the applications that were set aside.*

Test Group/Test Category

- MAC002—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- MAC102—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Warning

Resolution

If you want to use one of these incompatible applications, you need to get an updated version that is compatible with your new OS. Applications in the Mac App Store list their compatibility and system requirements on their product pages. You can also check with the application developer to find out if they have a new, compatible version or plan to release one.

Deprecated Property List Keys (MAC001 / MAC101)

For the MAC001 Operating System Compatibility test, the macOS application is scanned to determine if the package has references to deprecated property list keys, such as legacy Java applications.

Applications with the Java dictionary listed in their `.plist` file will trigger OS X to prompt the user to install Java 6, even if the application relies on newer versions of Java. This is a legacy function and triggers OS X to expect a legacy dependency (Java 5 or 6). Applications with the Java dictionary in the `.plist` file are highlighted as a compatibility error.

Test Group/Test Category

- **MAC001**—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- **MAC101**—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Warning

Removed APIs (MAC007 / MAC107)

For the MAC007 Operating System Compatibility test, the macOS application is scanned to determine if it contains a reference to a removed API call.

Test Group/Test Category

- **MAC007**—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- **MAC107**—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Error

Removed Frameworks (MAC006 / MAC106)

For this operating system compatibility test, the macOS application is scanned to determine if it contains a reference to a removed framework.

Test Group/Test Category

- **MAC006**—Operating System Compatibility/Apple/Desktop/macOS 10.11 El Capitan
- **MAC106**—Operating System Compatibility/Apple/Desktop/macOS 10.12 Sierra

Severity

Error

Google Android Tests

The Google Android categories consist of the following Operating System Compatibility tests:

- Application Requires Specific Minimum OS Version (M601 / M701 / M801 / M901 / M1201 / M1701 / M1801)

Application Requires Specific Minimum OS Version (M601 / M701 / M801 / M901 / M1201 / M1701 / M1801)

For this operating system compatibility test, the mobile app is scanned to determine if it requires a specific minimum OS version.

Test Group/Test Category

- **M601**—Operating System Compatibility/Google /Mobile/Android 4.1 Jelly Bean
- **M701**—Operating System Compatibility/Google /Mobile/Android 4.2 Jelly Bean
- **M801**—Operating System Compatibility/Google /Mobile/Android 4.3 Jelly Bean
- **M901**—Operating System Compatibility/Google /Mobile/Android 4.4 KitKat
- **M1201**—Operating System Compatibility/Google /Mobile/Android 5.0 Lollipop
- **M1701**—Operating System Compatibility/Google /Mobile/Android 6.0 Marshmallow
- **M1801**—Operating System Compatibility/Google /Mobile/Android 7.0 Nougat

Severity

Error

Resolution

Application should only be installed on a device with a compatible OS version.

MSIX Conversion Compatibility Tests

Use the MSIX Conversion Compatibility tests to check the compatibility of msi packages to be converted to msix.

- Device Driver (MSIX001)
- Windows/NT Service (MSIX002)
- Shortcuts (MSIX003)
- Custom Action (MSIX004)
- Conditionalized Components (MSIX005)
- Unsupported .Net Framework Version (MSIX006)
- Requires Elevated Privileges (MSIX007)

Device Driver (MSIX001)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether or not the package installs the device driver.

Test Group/Test Category

- **MSIX001**—MSIX Conversion Compatibility/Desktop

Severity

Error

Message

`The package is scanned to check whether it installs Device Drivers.`

Background

MSIX package format does not support Device Drivers. Applications which contain Device Drivers may not work as expected when converted to MSIX package format. It is recommended not to convert such applications to MSIX.

Windows/NT Service (MSIX002)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether or not the package installs the Windows/NT Service.

Test Group/Test Category

- **MSIX002**—MSIX Conversion Compatibility/Desktop

Severity

Warning

Message

`The package is scanned to check whether it installs Windows/NT Services.`

Background

MSIX package format does not support Windows/NT Services. Applications which contains Windows/NT Services may not work as expected when converted to MSIX package format. It is recommended not to convert such applications to MSIX.

Shortcuts (MSIX003)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether it has shortcuts or not.

Test Group/Test Category

- **MSIX003**—MSIX Conversion Compatibility/Desktop

Severity

Error

Message

`The package is scanned to check whether it has shortcuts.`

Background

Shortcuts are the entry point for the MSIX packages. Applications with no shortcuts are not recommended to be converted to MSIX.

Custom Action (MSIX004)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether it has custom actions.

Test Group/Test Category

- **MSIX004**—MSIX Conversion Compatibility/Desktop

Severity

Warning

Message

`The package is scanned to check whether it has custom actions.`

Background

Applications having custom actions are not recommended to be converted to MSIX.

Conditionalized Components (MSIX005)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether it has conditionalized components.

Test Group/Test Category

- **MSIX005**—MSIX Conversion Compatibility/Desktop

Severity

Warning

Message

`The package is scanned to check whether it has conditionalized components.`

Background

Applications having conditionalized components are not recommended to be converted to MSIX.

Unsupported .Net Framework Version (MSIX006)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether or not the package installs the unsupported .net framework version.

Test Group/Test Category

- **MSIX006**—MSIX Conversion Compatibility/Desktop

Severity

Warning

Message

```
The package is scanned to check whether it installs unsupported .net framework version.
```

Background

MSIX package format does not support Unsupported .Net Framework Version. Applications having unsupported .net framework version below 4.6.2 are not recommended to be converted to MSIX.

Requires Elevated Privileges (MSIX007)

For this MSIX Conversion Compatibility test, the msi packages are scanned to check whether it requires elevated privileges or not.

Test Group/Test Category

- **MSIX007**—MSIX Conversion Compatibility/Desktop

Severity

Warning

Message

```
The package is scanned to check whether it requires elevated privileges.
```

Background

MSIX package format does not requires elevated privileges. Applications having shortcut exe which requires elevated privileges are not recommended to be converted to MSIX.

Application Virtualization Compatibility Tests

AdminStudio's Analyze offers tests for the compatibility of Windows Installer packages to be converted to virtual formats.

- Installer Analysis Tests

Installer Analysis Tests

AdminStudio uses the application virtualization compatibility installer analysis tests to determine if a Windows Installer package is a suitable candidate for virtualization to Microsoft App-V, VMware ThinApp, Citrix XenApp.

You can choose to customize your test results so that only the virtualization formats that you are interested in are displayed.

- Application Virtualization Compatibility Installer Analysis Tests
- Choosing the Virtual Formats to Display in Test Results

Application Virtualization Compatibility Installer Analysis Tests

AdminStudio uses the following tests to determine if a Windows Installer package is a suitable candidate for virtualization to Microsoft App-V, VMware ThinApp, Citrix XenApp.

- Tests That Return Errors or Warnings
- Tests That Indicate Repackaging is Required

Tests That Return Errors or Warnings

The following tests return errors or warnings for positive test results. An icon indicating the severity of a positive test result is listed in the column of each of the virtual technologies that the test applies to. If the test does not apply to a specific virtualization technology, a gray bar is displayed in that technology's column.

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
Invalid Windows Installer Package Package is not a valid Windows Installer package. Usually this issue is found in legacy installations that require repackaging. In rare cases, this issue could also be found in a Windows Installer package that has become corrupted.					

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests (cont.)

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
No Shortcut This package contains no shortcuts. Shortcuts are necessary to define the entry point into the virtual application. • **App-V suitability**—For conversion to App-V packages, this issue is acceptable in some scenarios, such as packages that provide dependencies to others that dynamically suite it. However, if this package merely provides a plug-in to another application, it must contain a shortcut to launch that application in this package's virtual context. • **ThinApp and XenApp suitability**—For conversion to ThinApp 4.x and XenApp formats, shortcuts are necessary to define the entry point into the virtual application. One potential resolution to this issue is to use InstallShield Editor to add one or more shortcuts to the Windows Installer package.					
ClickOnce Package contains a ClickOnce application. ClickOnce is a per-user installation format that is often incompatible with the per-machine nature of virtual package deployment. A ClickOnce application also may try to automatically update itself, which results in invalid versioning in the application virtualization client.					

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests (cont.)

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
Shell Extension Package contains a shell extension. Shell extensions extend Windows Explorer and cannot be loaded from a virtual package. This extension may be critical to the use of this application, and, if so, this application will not function when virtualized. However if this extension is non-critical, the application may function when virtualized.		—			
OS Integrated Package contains files that are closely integrated with the operating system. The files that make up applications like Internet Explorer or Windows Media Player, or frameworks like the .NET Framework, do not make good candidates for virtualization. These files should instead be installed locally on the machine.					
Boot Service Package contains a service that starts at boot-time. Virtualized services are limited to the lifetime of the virtual application, so services that must start at boot-time do not make good candidates for virtualization to App-V or XenApp formats. It may be possible to extract this service such that it can be installed locally on the machine and allow the rest of the package to be virtualized.				—	—

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests (cont.)

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
Too Large Package contains more than 4 GB of files. Since App-V 4.x and XenApp do not support packages that contain more than 4 GB of files, this application cannot be successfully virtualized to App-V 4.x or XenApp as an uncompressed package. However, if the compressed size of the package is less than 4 GB, then this application can be virtualized to these formats as a compressed package.		—		—	—
COM Surrogate DLLs Package contains a COM DLL that uses surrogate virtualization. App-V, XenApp, and ThinApp do not support COM DLL surrogate virtualization, so this package may not work correctly if virtualized.					
COM Plus Package contains a COM Plus component. App-V, XenApp, and ThinApp do not support COM+ components, so this package may not work correctly if virtualized.					
Device Driver Package contains a device driver. System-level drivers such as print drivers or USB device drivers do not work from a virtualized environment. It may be possible to extract this driver such that it can be installed locally on the machine and allow the rest of the package to be virtualized.					
64-Bit Package Package is a 64-bit package. XenApp and ThinApp 4.x do not support virtualization of 64-bit packages.	—	—			—

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests (cont.)

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
ASP.NET/IIS Application Package contains an ASP.NET or IIS application component, which is not supported by App-V 4.x, App-V 5.x, XenApp, and ThinApp. If the ASP.NET or IIS application component is not an important part of the application, or if it can be separately installed from the package, this error can be suppressed and ignored.					
WMI Provider Package contains a WMI provider component, which is not supported by App-V 4.x, App-V 5.x, XenApp, and ThinApp. If the WMI Provider component is not an important part of the application, or if it can be separately installed from the App-V package, this error can be suppressed and ignored.					
J2EE Application Server Package contains a J2EE application server, which is not supported by App-V, XenApp, or ThinApp. If the J2EE application is not an important part of the application, or if it can be separately installed from the package, this error can be suppressed and ignored.					
Unsupported Application (Error) This package contains an application known to not be a good candidate for virtualization.					

Table 17-2 • Application Virtualization Compatibility / Installer Analysis Tests (cont.)

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
Unsupported Application (Warning) This package contains some files that indicate the presence of unsupported applications such as antivirus software or various server software such as Exchange Server or SQL Server. If these unsupported application components are not an important part of the application, or if they can be separately installed from the package, this error can be suppressed and ignored.	⚠	⚠	⚠	⚠	⚠
URL Protocol Package registers an URL protocol.	⚠	—	⚠	—	—
Default Program Package registers its capabilities in the Default Programs list.	⚠	—	⚠	⚠	⚠

Tests That Indicate Repackaging is Required

When a positive test result is returned for any of the following tests, neither an error nor a warning is generated because these issues can be resolved by repackaging the package. For these tests, if a positive test result is generated, an informational icon is displayed on the **Application Virtualization Compatibility** tab of the **Analyze Deployment Type View**.

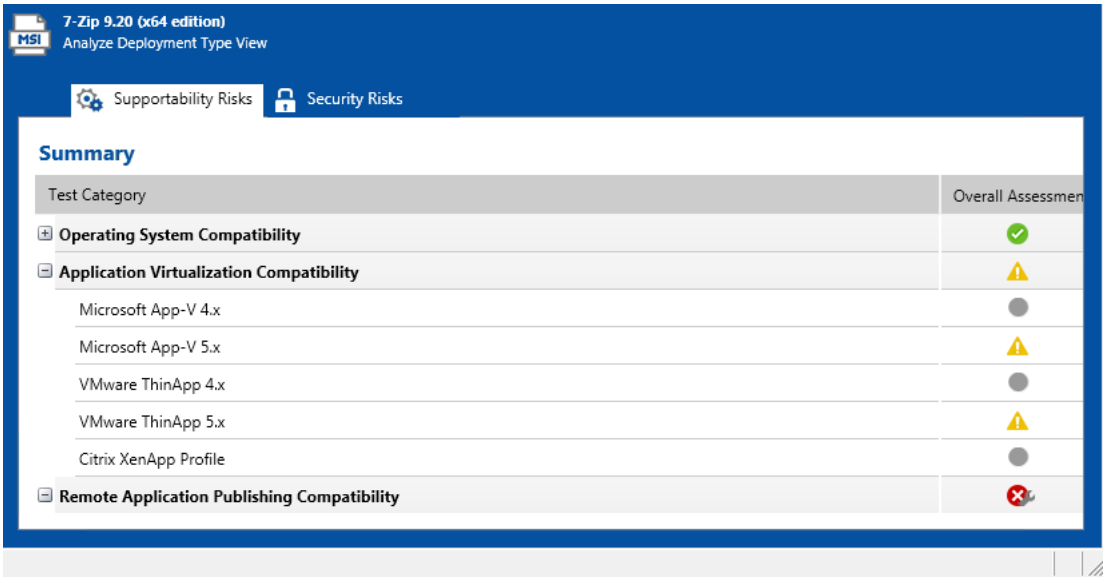


Figure 17-1: Informational Messages on Application Virtualization Compatibility Tab

If a package has a positive test result in this category, Automated Application Converter will automatically repackage it on a clean VM before converting it to a virtual package.

Because a positive test result to one of these tests does not have an impact on a package's overall compatibility to be virtualized, a **Ready** icon is displayed for the package on summary views if these are the only issues that the package has generated. For more information, see Hierarchical Level of Status Icons.

Table 17-3 • Application Virtualization Compatibility / Installer Analysis Tests That Indicate Repackaging is Required

Description	App-V 4.x	App-V 5.x	XenApp	ThinApp 4.x	ThinApp 5.x
Conditionalized Component This package contains one or more components which are only installed in under certain conditions. Since some components may be excluded in certain environments, the exact set of files and registry to convert must be determined by repackaging this application.	✓	✓	✓	✓	✓
Unsupported Table This package contains one or more tables that are not supported by direct conversion. Since these tables may result in the addition or removal of files or registry, the exact set to convert must be determined by repackaging this application.	✓	✓	✓	✓	✓
Custom Action This package contains one or more unknown custom actions. Since these actions may result in the addition or removal of files or registry, the exact set to convert must be determined by repackaging this application.	✓	✓	✓	✓	✓

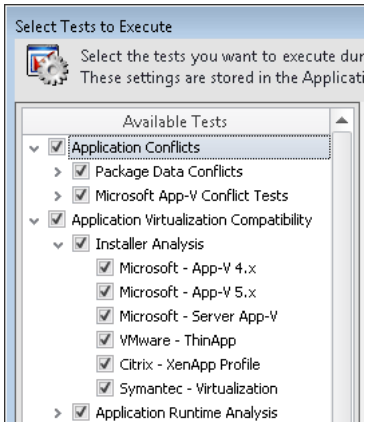
Choosing the Virtual Formats to Display in Test Results

All of the tests in the **Installer Analysis** subcategory of the **Application Virtualization Compatibility** test category are always run each time that you run Application Virtualization Compatibility tests in Analyze.

However, if you do not want to display test results for a specific virtual technology, perform the following steps:

Task ***To choose the virtual formats to display in test results:***

1. On the Application Catalog **Analyze** tab, click the **Select Tests to Execute** button in the ribbon. The **Select Tests to Execute** dialog box opens.
2. Expand the subcategories of the **Application Virtualization Compatibility > Installer Analysis** category:



3. Under **Installer Analysis**, clear the selection of the virtual formats that you do not want to display in test results.
4. Click **OK**.

Best Practices Tests

Edition • *The following tests are included in AdminStudio Professional and Enterprise Editions:*

- *Windows Installer Internal Consistency Evaluators*
- *Windows Installer Best Practices ACE tests*
- *Remote Desktop Services Tests*

The following subcategories of best practice tests are available:

- Windows Installer Internal Consistency Evaluators
- Windows Installer Best Practices Tests
- Microsoft App-V Best Practices Tests
- Apple Best Practices Tests

Windows Installer Internal Consistency Evaluators

Edition • *The Windows Installer Internal Consistency Evaluators are included in AdminStudio Professional and Enterprise Editions.*

The internal consistency evaluators (ICEs) are tests that you can run to check whether Windows Installer packages are valid databases that perform as expected. These tests validate the data in each table of a package, as well as the data among tables.

Examples of ICEs are:

- ICE08: Each component has a unique component code.
- ICE09: Any component that is being installed to the Windows System folder is marked as permanent.
- ICE12, ICE75, and others: custom actions are scheduled at valid times in the installation sequences.

Analyze includes more than 100 ICE tests. ICE tests are stored in .cub files, which are Windows Installer-format databases that perform custom actions that validate data in Windows Installer databases.

About ICE43, ICE50, and ICE57 Tests for Shortcuts

Edition • *These tests are included in AdminStudio Professional and Enterprise Editions.*

Each entry in the CreateLink section of the `.inc` file is converted into an entry in the Shortcut table. The exact properties of the shortcut depends on the information in the CreateLink line as well as the nature of the target file itself.

ICE43, ICE50, and ICE57 are the most common validation tests for shortcuts.

Shortcut Types

The primary distinction between shortcuts is advertised vs. non-advertised. Here are two reasons why it is preferable to create advertised shortcuts:

- Advertised shortcuts are triggers for MSI's self-repair mechanism.
- Non-advertised shortcuts are intended for a per-user context only:
  - The target file must be a file installed in a user-specific directory.
  - The key path of the component that contains the target file must be a user-specific registry value.

Conversion from CreateLink Entries to Shortcut Table Entries

The .inc converter always tries to create advertised shortcuts for every CreateLink line that is found in the .inc file. However, not every CreateLink line can be converted into an advertised shortcut.

To create an advertised shortcut, the information in the CreateLink line must meet all of the following requirements:

- The target file must be the key path of its component. This means that the target file must be listed in the .inc file list. The converter will create a new component for a non-PE (portable executable) target files, so that it is guaranteed to become the key path of the component. (Normally, for each target directory, non-PE files are grouped together into one component).
- The target file must contain an icon.

In general, this means that as long as the target file contains an icon, the converter is able to create an advertised shortcut for it. However, whenever a shortcut cannot be advertised, the converter does the following:

- It creates a “catch all” component (if not yet created) named `ShortcutsComponent`. It also creates an HKCU registry entry in the Registry table, and that entry is used as the key path for `ShortcutsComponent`. This is done to avoid ICE43.
- A new shortcut entry is created in the Shortcut table, and it is associated with `ShortcutsComponent`.

Windows Installer Best Practices Tests

Edition • *The Windows Installer Best Practices ACE tests are included in AdminStudio Professional and Enterprise Editions.*

The following Windows Installer best practice tests are described in this section:

Table 17-4 • Windows Installer Best Practice Tests

Category	Test
Components	• ACE04: Components Without Files or Key Paths • ACE05: More Than One Executable File Per Component • ACE06: Executable File Not Marked as Key File of Component
Merge Module Integrity	• ACE26: Merge Modules That Are Missing from the Application Catalog • ACE36: Merge Module Dependencies That Are Missing from the Application Catalog
Recommended Tests	• ACE25: Hard-Coded Paths for Custom Action Targets • ACE27: Duplicate File Data Without the Required Standard Actions • ACE28: Hard-Coded Paths for Environment Variable Values • ACE29: Hard-Coded Paths for INI File Changes • ACE31: MoveFile Data Without the Required Standard Actions • ACE32: Hard-Coded Paths in Registry Entries • ACE33: RemoveFile Data Without the Required Standard Actions • ACE34: RemoveIniFile Data Without the Required Standard Actions • ACE35: RemoveRegistry Data Without the Required Standard Actions

ACE04: Components Without Files or Key Paths

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE04 verifies that components with no files and no key paths have an associated entry in the CreateFolder table according to Windows Installer best practices.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Components

Severity

Error

Message

```
The component [COMPONENT1] in the package [PACKAGE1] does not have a key file, a key path, or an
    associated entry in the CreateFolder table.
```

Background

ACE04 is designed to identify and fix the same issue that ICE18 detects. It flags components that have an empty KeyPath column, that do not have any files, and that also do not have associated entries in the RemoveFile, DuplicateFile, and MoveFile tables.

Resolution

Automatic Fix (CARD04)

CARD04 creates a CreateFolder table entry for the component by executing the following query:

```
INSERT INTO CreateFolder ( `Directory_`, `Component_` )
     VALUES ('Source Directory','Source Component')
```

ACE05: More Than One Executable File Per Component

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE05 checks for the existence of more than one executable file (.exe, .dll, .ocx, .hlp, .chm, .tlb, .sys, .drv) per component in a Windows Installer package.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Components

Severity

Error

Message

```
The component [COMPONENT1] in the package [PACKAGE1] has more than one executable module (*.chm, *.dll,
    *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file [EXECUTABLE_MODULE_FILENAME] must be included
    as a separate component.
```

Background

If more than one executable file (.exe, .dll, .ocx, .hlp, .chm, .tlb, .sys, .drv) exists in a component, ACE05 fails.

Resolution

Automatic Fix (CARD05)

CARD05 automatically modifies the component so that only one .exe or .dll exists, and it adds new components for remaining .exe, .dll, .ocx, .hlp, .chm, .tlb, .sys, and .drv files. To do this, CARD05 generates a new component name and ComponentId and inserts a record in the Component table and in the FeatureComponents table. The relevant entry in the File table is then updated to effectively move the file into the new component.

ACE06: Executable File Not Marked as Key File of Component

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE06 checks whether the executable file (.exe, .dll, .ocx, .hlp, .chm, .tlb, .sys, .drv) within the component is the key file.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Components

Severity

Error

Message

```
The component [COMPONENT1] in the package [PACKAGE1] does not have an executable module (*.chm, *.dll,
    *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb) as the key file. The current key file [NON-
    EXECUTABLE_FILENAME] can be replaced with [EXECUTABLE_MODULE_FILENAME].
```

Background

If an executable file is not the key file of its component, ACE06 fails.

Resolution

Automatic Fix (CARD06)

CARD06 automatically makes the executable file the key file of its component. To do this, CARD06 runs the following query and replaces the key path with a file entry from the File column of the File table; the file entry is associated with this component and of the correct type.

```
SELECT `KeyPath` FROM `Component` WHERE `
     ComponentId` = 'Source ComponentId'
```

ACE25: Hard-Coded Paths for Custom Action Targets

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE25 checks the entries in the CustomAction table to identify any hard-coded paths.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The [CUSTOM_ACTION_NAME] custom action has a hard coded directory path of [TARGET_PATH] in its Target
    field.
```

Background

If a package has a hard-coded path in the Target column of the CustomAction table, ACE25 fails.

Resolution

Manual Fix

Open the package file in InstallShield Editor, and in the Target column of the CustomAction table, change any hard-coded paths to relative paths.

ACE26: Merge Modules That Are Missing from the Application Catalog

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE26 checks whether the Merge Modules in a package are present in the Application Catalog.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Merge Module Integrity

Severity

Warning

Message

```
The [MERGE_MODULE_NAME], version [VERSION] Merge Module is included with this package and yet not
    imported into the Application Catalog. It is recommended that all Merge Modules be imported into the
    Application Catalog.
```

Background

If a package refers to a Merge Module that does not exist in the Application Catalog, ACE26 fails.

Note • *ACE26 and ACE36, optional Best Practice ACEs, both check for conflicts with Merge Modules. ACE26 checks merge modules that are listed in the ModuleSignature table, while ACE36 checks the ModuleDependency table.*

These ACEs are provided to encourage you to import Merge Modules into the Application Catalog and, by doing so, improve the effectiveness of ACE12, which checks for components that contain files that could be replaced by one of the imported Merge Modules.

Resolution

Manual Fix

To resolve this issue, open the package file in Application Catalog and import the identified Merge Module into the Application Catalog.

ACE27: Duplicate File Data Without the Required Standard Actions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE27 checks whether data in the DuplicateFile table is executed with an associated DuplicateFiles standard action.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The package contains data ([FileKey]) in the 'DuplicateFile' table but not the necessary actions to use
    this data. You should consider whether a 'DuplicateFiles' or 'RemoveDuplicateFiles' action is needed
    for your 'InstallExecuteSequence' table.
```

Background

If a package contains data in the DuplicateFile table but not the necessary actions to use this data, ACE27 fails.

Resolution

Manual Fix

To resolve this issue, determine whether a DuplicateFiles action or a RemoveDuplicateFiles action is needed for your InstallExecuteSequence table. If so, either open the MSI file in InstallShield Editor and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the DuplicateFile table.

ACE28: Hard-Coded Paths for Environment Variable Values

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE28 checks the entries of the Environment table to identify hard-coded paths.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The [ENVIRONMENT_TABLE_NAME] Environment table entry has a hard coded directory path of [DIRECTORY_PATH]
    in its Value field.
```

Background

If the Value column of the Environment table in a package contains any hard-coded paths, ACE28 fails.

Resolution

Manual Fix

To resolve this issue, open the package file in InstallShield Editor and change any hard-coded paths in the Value column of the Environment table to relative paths.

ACE29: Hard-Coded Paths for INI File Changes

ACE29 checks the entries in the IniFile table to identify hard-coded paths.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The [PATH_NAME]\[INI_FILE_NAME] INI file has a hard coded directory path of [HARD_CODED_PATH] in its
    Value field.
```

Background

If the Value column of the IniFile table of a package contains any hard-coded paths, ACE29 fails.

Resolution

Manual Fix

To resolve this issue, open the package file in InstallShield Editor and change any hard-coded paths in the Value column of the IniFile table to relative paths.

ACE31: MoveFile Data Without the Required Standard Actions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE31 checks whether data in the MoveFile table is being executed with an associated MoveFiles standard action.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The package contains data ([FileKey]) in the 'MoveFile' table but not the necessary actions to use this
    data. You should consider whether a 'MoveFiles' action is needed for your 'InstallExecuteSequence'
    table.
```

Background

If a package contains data in the MoveFile table but not the necessary actions to use this data, ACE31 fails.

Resolution

Manual Fix

To resolve this issue, determine whether a MoveFiles action is needed for your InstallExecuteSequence table. If so, either open the MSI file in InstallShield Editor and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the MoveFile table.

ACE32: Hard-Coded Paths in Registry Entries

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE32 checks the entries of the registry table to identify hard-coded paths.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The [REGISTRY_TABLE_ENTRY_NAME] Registry table entry has a hard coded directory path of
    [DIRECTORY_PATH_NAME] in its Value field.
```

Background

If the Value column of the Registry table of a package contains any hard-coded paths, ACE32 fails.

Resolution

Manual Fix

To resolve this issue, open the package file in InstallShield Editor and change any hard-coded paths in the Value column of the Registry table to relative paths.

ACE33: RemoveFile Data Without the Required Standard Actions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE33 checks whether data in the RemoveFile table is being executed with an associated RemoveFiles action.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The package contains data ([FileKey]) in the 'RemoveFile' table but not the necessary actions to use this
    data. You should consider whether a 'RemoveFiles' action is needed for your 'InstallExecuteSequence'
    table.
```

Background

If a package contains data in the RemoveFile table but not the necessary actions to use this data, ACE33 fails.

Resolution

Manual Fix

To resolve this issue, determine whether a RemoveFiles action is needed for your InstallExecuteSequence table. If so, either open the MSI file in InstallShield Editor and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveFile table.

ACE34: RemoveIniFile Data Without the Required Standard Actions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE34 checks whether data in the RemoveIniFile table is executed with an associated RemoveIniFiles action.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The package contains data ([RemoveIniFile]) in the 'RemoveIniFile' table but not the necessary actions
    to use this data. You should consider whether a 'RemoveIniFiles' action is needed for your
    'InstallExecuteSequence' table.
```

Background

If a package contains data in the RemoveIniFile table but not the necessary actions to use this data, ACE34 fails.

Resolution

Manual Fix

To resolve this issue, determine whether a RemoveIniFiles action is needed for your InstallExecuteSequence table. If so, either open the MSI file in InstallShield Editor and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveIniFile table.

ACE35: RemoveRegistry Data Without the Required Standard Actions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE35 checks whether data in the RemoveRegistry table is executed with an associated RemoveRegistryValues action.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Recommended Rules

Severity

Warning

Message

```
The package contains data ([RemoveRegistry]) in the 'RemoveRegistry' table but not the necessary actions
    to use this data. You should consider whether a 'RemoveRegistryValues' action is needed for your
    'InstallExecuteSequence' table.
```

Background

If a package contains data in the RemoveRegistry table but not the necessary actions to use this data, ACE35 fails.

Resolution

Manual Fix

To resolve this issue, determine whether a RemoveRegistryValues action is needed for your InstallExecuteSequence table. If so, either open the MSI file in InstallShield Editor and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveRegistry table.

ACE36: Merge Module Dependencies That Are Missing from the Application Catalog

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

ACE36 checks whether a package's Merge Module dependencies are present in the Application Catalog.

Test Group/Test Category/Test Subcategory

Best Practices/Windows Installer Best Practices/Merge Module Integrity

Severity

Warning

Message

```
The required [MERGE_MODULE_NAME], version [REQUIRED_VERSION] Merge Module is included with this package
    and yet not imported into the Application Catalog. It is recommended that all Merge Modules be
    imported into the Application Catalog.
```

Background

If any of a package's Merge Module dependencies do not exist in the Application Catalog, ACE36 fails.

Note • *ACE26 and ACE36, optional Best Practice ACEs, both check for conflicts with Merge Modules. ACE26 checks merge modules that are listed in the* ModuleSignature *table, while ACE36 checks the* ModuleDependency *table.*

These ACEs are provided to encourage you to import Merge Modules into the Application Catalog and, by doing so, improve the effectiveness of ACE12, which checks for components that contain files that could be replaced by one of the imported Merge Modules.

Resolution

Manual Fix

Import the package's missing Merge Modules into the Application Catalog.

Microsoft App-V Best Practices Tests

The following Microsoft App-V best practices ACE tests are described in this section:

- ACE201: Shortcuts with Hard-Coded Paths for Targets
- ACE202: Shortcuts with Hard-Coded Paths in Command-Line Arguments
- ACE203: Shortcut Targets with Hard-Coded Paths for the Working Directory
- ACE208: App-V Packages Without at Least One Shortcut
- ACE209: App-V Packages with Shell Extensions
- ACE210: App-V Packages with ClickOnce Support
- ACE211: App-V Package with DLL Surrogates
- ACE212: App-V Packages with Boot Services
- ACE213: App-V Packages with OS Integrated Files
- ACE214: App-V Packages with Drivers
- ACE216: App-V Package with Long .sft File Names
- ACE217: App-V Packages with WMI Providers
- ACE218: App-V Package with a J2EE Application Server
- ACE219: App-V Packages with ASP.NET or IIS Components
- ACE220: App-V Packages with Unsupported Applications

ACE201: Shortcuts with Hard-Coded Paths for Targets

ACE201 checks whether a target in the package has a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

`Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded Target of [TARGET].`

Background

If a shortcut in package has a hard-coded path, ACE201 fails.

Resolution

Manual Fix

To resolve this ACE in an App-V package, change the path of the target to use a variable instead of a hard-coded path.

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.
4. In the **Target** setting, replace the existing hard-coded path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`.

Note • *If there is no appropriate* `CSIDL` *or* `SFT` *constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE202: Shortcuts with Hard-Coded Paths in Command-Line Arguments

ACE202 checks whether a command-line argument for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

```
Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded argument of [ARGUMENTS].
```

Background

If a command-line argument for a target in the package includes a hard-coded path, ACE202 fails.

Resolution

Manual Fix

To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path.

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.
4. In the **Arguments** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path.

Note • *If there is no appropriate* `CSIDL` *or* `SFT` *constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE203: Shortcut Targets with Hard-Coded Paths for the Working Directory

ACE203 checks whether a working directory for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

```
Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded working directory of
    [DIRECTORY_NAME].
```

Background

If the package contains a shortcut target whose working directory is a hard-coded path, ACE203 fails.

Resolution

Manual Fix

To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path.

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.

4. In the **Working Directory** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path.

Note • *If there is no appropriate* CSIDL *or* SFT *constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE208: App-V Packages Without at Least One Shortcut

ACE208 checks whether an App-V package contains at least one shortcut.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

```
Package [PACKAGE_NAME] has no shortcuts.
```

Background

If the App-V package does not contain any shortcuts, ICE208 fails.

Resolution

Manual Fix

You can ignore this ACE if one of the following is true:

- This package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package in the Virtual Package Editor and select this App-V package as a dependency in the Dependencies view.
- This package is intended to be used as a plug-in. In this case, you need to create a shortcut to the application for which this is a plug-in. Some common examples include Office and Internet Explorer.

If end users need to be able to launch this App-V package independently, consider opening the package in the Virtual Package Editor and adding a target to the App-V package if necessary (through the Shortcuts view), and then adding a shortcut to the target.

ACE209: App-V Packages with Shell Extensions

ACE209 checks App-V packages for shell extensions.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

`Package [PACKAGE_NAME] has shell extensions.`

Background

If an App-V package has a shell extension, ACE209 fails.

Resolution

Manual Fix

You need to assess how important the shell extension is to the application so that you can determine if it matters whether the shell extension behaves as intended. You then need to do one of the following:

- **If the shell extension is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the shell extension is important**, this package will not function well, and it should not be deployed.

ACE210: App-V Packages with ClickOnce Support

ACE210 checks App-V packages for ClickOnce installations.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning

Message

`Package [PACKAGE_NAME] has ClickOnce.`

Background

If an App-V package has a ClickOnce installation, ACE210 fails.

Resolution

Manual Fix

You need to assess how important the ClickOnce installation is to the application so that you can determine if it matters whether the ClickOnce installation behaves as intended. You then need to do one of the following:

- **If the ClickOnce installation is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the ClickOnce installation is important**, this package will not function well, and it should not be deployed.

ACE211: App-V Package with DLL Surrogates

ACE211 checks App-V packages for DLL surrogates.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

`Package [PACKAGE_NAME] has dll surrogates.`

Background

If an App-V package has a DLL surrogate, ACE211 fails.

Resolution

Manual Fix

You need to assess how important the DLL surrogate is to the application so that you can determine if it matters whether the DLL surrogate behaves as intended. You then need to do one of the following:

- **If the DLL surrogate is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the DLL surrogate is important**, this package will not function well, and it should not be deployed.

ACE212: App-V Packages with Boot Services

ACE212 checks App-V packages for boot services.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

`Package [PACKAGE_NAME] has boot services.`

Background

If an App-V package has boot services, ACE212 fails.

Resolution

Manual Fix

You need to assess how important the boot service is to the application so that you can determine if it matters whether the boot service behaves as intended. You then need to do one of the following:

- **If the boot service is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the boot service is important but separable**, install the boot service on the main machine, and virtualize the rest of the application. A modified virtual package can then be deployed successfully.
- **If the boot service is important but not separable**, this package will not function well, and it should not be deployed.

ACE213: App-V Packages with OS Integrated Files

ACE213 checks App-V packages for OS integrated files.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

`Package [PACKAGE_NAME] has OS integrated files.`

Background

If an App-V package has an OS integrated file, ICE212 fails.

Resolution

It is likely that the OS integrated file is central to the virtual package. Therefore, it is recommended that this App-V package not be deployed.

ACE214: App-V Packages with Drivers

ACE214 checks App-V packages for drivers.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

```
Package [PACKAGE_NAME] has drivers.
```

Background

If an App-V package has a driver, ACE214 fails.

Resolution

Manual Fix

Separate the drivers from the rest of the application so that the drivers can be installed on the main machine. Then, virtualize the rest of the application.

ACE216: App-V Package with Long .sft File Names

ACE216 checks whether an App-V package's .sft file name contains more than 56 characters.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

```
Package [PACKAGE_NAME] has a SFT filename that is too long (> 56 characters).
```

Background

If the .sft file for the App-V package contains more than 56 characters, ACE216 fails.

Resolution

Manual Fix

To resolve this ACE in an App-V package, rename the `.sft` file with a name that contains fewer than 56 characters.

ACE217: App-V Packages with WMI Providers

ACE217 checks whether the App-V package contains a WMI Provider component.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

```
WMI provider [PROVIDER_NAME] was found in package [PACKAGE_NAME].
```

Background

If an App-V package includes a WMI Provider component, ACE217 fails.

Resolution

Manual Fix

If the WMI Provider is not an important part of the application, or if it can be separately installed from the App-V package, this error can be suppressed and ignored.

ACE218: App-V Package with a J2EE Application Server

ACE218 checks whether the App-V package contains a J2EE application server.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

```
Files such as [FILE_NAME] indicate that a J2EE application server is in package [PACKAGE_NAME].
```

Background

If an App-V package includes a J2EE application server, ACE218 fails.

Resolution

Manual Fix

If the J2EE application is not an important part of the application, or if it can be separately installed from the App-V package, this error can be suppressed and ignored.

ACE219: App-V Packages with ASP.NET or IIS Components

ACE219 checks whether the App-V package contains an ASP.NET or IIS application component.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Error

Message

```
Files such as [FILE_NAME] indicate that an ASP.NET application is in package [PACKAGE_NAME].
```

Background

If an App-V package includes an ASP.NET or IIS application component, ACE219 fails.

Resolution

Manual Fix

If the ASP.NET/IIS application is not an important part of the application, or if it can be separately installed from the App-V package, this error can be suppressed and ignored.

ACE220: App-V Packages with Unsupported Applications

ACE220 checks whether an App-V package contains files that indicate that the package includes unsupported applications such as antivirus software or server software such as Exchange Server or SQL Server.

Test Group/Test Category

Best Practices/Microsoft App-V Best Practices

Severity

Warning or error

Message

```
Files such as [FILE_NAME] indicate that the unsupported application [APPLICATION_NAME] is in package
    [PACKAGE_NAME].
```

Background

If an App-V package contains files that indicate that the package includes unsupported applications such as antivirus software or server software such as Exchange Server or SQL Server, ACE220 fails.

Resolution

Manual Fix

If these unsupported application components are not an important part of the application, or if they can be separately installed from the App-V package, this error can be suppressed and ignored.

Apple Best Practices Tests

The following Apple Best Practices tests are described in this section:

- M001: Recommended Policy Keys are Specified to Ensure Proper Classification of the Application (Info.plist)
- M002: Default Policy Keys are Defined When Device-Specific Versions are Present
- M003: Localization Resources are Present and Contain All Required Information
- M004: Localization Resources are Present and Contain the Recommended Keys
- MAC701: Recommended Property List Keys
- MAC702: Code Signature
- MAC703: Volume Purchase Program
- MAC704: Allows In-app Purchases

M001: Recommended Policy Keys are Specified to Ensure Proper Classification of the Application (Info.plist)

M001 verifies that policy keys are specified in the application profile.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

M002: Default Policy Keys are Defined When Device-Specific Versions are Present

M002 verifies that default policy keys are defined.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

M003: Localization Resources are Present and Contain All Required Information

M003 verifies that location resources are complete.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

M004: Localization Resources are Present and Contain the Recommended Keys

M004 verifies that the recommended keys for localization are present.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

MAC701: Recommended Property List Keys

For the MAC701 Apple Best Practices test, the macOS application is scanned to determine to find out if it contains the recommended property list keys. If it does not contain the recommended property list keys, a warning is generated.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

MAC702: Code Signature

For the MAC702 Apple Best Practices test, the macOS application is scanned to determine to find out if it contains a digital signature. If it does not contain a digital signature, a warning is generated.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

MAC703: Volume Purchase Program

For the MAC703 Apple Best Practices test, the macOS application is scanned to determine it is enabled for the Volume Purchase Program.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

MAC704: Allows In-app Purchases

For the MAC704 Apple Best Practices test, the macOS application is scanned to determine it allows in-app purchases.

Test Group/Test Category

Best Practices/Apple Best Practices

Severity

Warning

Risk Assessment Tests

The following subcategories of risk assessment tests are available:

- Mobile Risk Assessment Tests
- Desktop Risk Assessment Tests

Mobile Risk Assessment Tests

The following categories of mobile risk assessment tests are described in this section:

- Windows Mobile Risk Assessment Tests

- Android Mobile Risk Assessment Tests
- Apple Mobile Risk Assessment Tests

Windows Mobile Risk Assessment Tests

The following Windows mobile risk assessment tests are described in this section:

- M4001: Application Requires Telephony
- M4002: Application Requires Wi-Fi
- M4004: Application Uses a Camera
- M4006: Application Uses a Front-Facing Camera
- M4008: Application Uses a Video Camera
- M4010: Application Uses a Gyroscope
- M4011: Application Uses Location Services
- M4013: Application Uses a Magnetometer
- M4015: Application Uses the Microphone
- M4020: Application Uses Peer-to-Peer via Bluetooth
- M4021: Application Uses Bluetooth LE
- M4030: Application Accesses the Address Book
- M4031: Application Supports In-App Purchases
- M4037: Application Uses the NFC Card Emulation Feature in the Device
- M4040: Application Uses the Device Proximity Sensor
- M4045: Application Uses USB Feature
- M4046: Application Accesses the Calendar
- M4050: Application Uses Internet Access
- M4052: Application Uses External Storage
- M4053: Application Uses HID
- M4054: Application Uses POS
- M4055: Application Accesses the Local Document Library
- M4056: Application Accesses Local Picture Library
- M4057: Application Accesses Local Video Library
- M4058: Application Accesses Local Music Library
- M4059: Application Supports Enterprise Authentication
- M4060: Application Uses Shared User Certificates

- M4061: Application Supports Private Network Access
- M4062: Application Uses Web Camera
- M4063: Application Uses Web Browser
- M4064: Application Uses DirectX 11
- M4065: Application Uses Digital Compass
- M4067: Application Uses Push Notification Service
- M4068: Application Uses Speech Recognition
- M4069: Application Uses Local Ring Tones
- M4070: Other App Management
- M4071: Wallet
- M4072: AllJoyn
- M4073: Supports User Profiles
- M4074: VOIP Service
- M4075: Screen Projection
- M4076: Application Uses Device Unlock
- M4077: VPN Features

M4001: Application Requires Telephony

M4004 scans the mobile app for the presence of telephony usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4002: Application Requires Wi-Fi

M4002 scans the mobile app for the presence of wireless networking usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4003: SMS Scheme

M4003 scans the mobile app for the presence of SMS scheme usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4004: Application Uses a Camera

M4004 scans the mobile app to determine if it uses the device camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4006: Application Uses a Front-Facing Camera

M206 scans the mobile app to determine if it uses a front-facing camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4008: Application Uses a Video Camera

M4008 scans the mobile app to determine if it uses the video camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4010: Application Uses a Gyroscope

M210 scans the mobile app to determine if it uses the device gyroscope.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4011: Application Uses Location Services

M4011 scans the mobile app to determine if it uses location services.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4013: Application Uses a Magnetometer

M213 scans the mobile app to determine if it uses the device magnetometer.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4015: Application Uses the Microphone

M215 scans the mobile app to determine if it uses the device microphone.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4020: Application Uses Peer-to-Peer via Bluetooth

M220 scans the mobile app to determine if it uses peer-to-peer connectivity via Bluetooth.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4021: Application Uses Bluetooth LE

M221 scans the mobile app to determine if it uses Bluetooth LE.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4030: Application Accesses the Address Book

M230 scans the mobile app to determine if it accesses the user's address book.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4031: Application Supports In-App Purchases

M4031 scans the mobile app to determine if it supports in-app purchases.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4037: Application Uses the NFC Card Emulation Feature in the Device

M237 scans the mobile app to determine if it uses the NFC card emulation feature in the device.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4040: Application Uses the Device Proximity Sensor

M240 scans the mobile app to determine if it uses the device proximity sensor.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4045: Application Uses USB Feature

M245 scans the mobile app to determine if it uses USB features.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4046: Application Accesses the Calendar

M4046 scans the mobile app to determine if it accesses the calendar.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4050: Application Uses Internet Access

M250 scans the mobile app to determine if it uses Internet access.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4052: Application Uses External Storage

M4052 scans the mobile app to determine if it uses external storage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4053: Application Uses HID

M4053 scans the mobile app to determine if it allows access to devices that support the Human Interface Device (HID) protocol.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4054: Application Uses POS

M4054 scans the mobile app to determine if it accesses the point of service (POS) API which lets your Windows Store app access bar code scanners and magnetic stripe readers.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4055: Application Accesses the Local Document Library

M4054 scans the mobile app to determine if it allows programmatic access to the user's Document library, which is restricted to the file type associations declared in the package manifest.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4056: Application Accesses Local Picture Library

M4056 scans the mobile app to determine if it provides programmatic access to the user's Picture library, allowing the app to enumerate and access all files in the library without user interaction. This capability is usually used in photo apps that make sue of the entire Pictures library.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4057: Application Accesses Local Video Library

M4057 scans the mobile app to determine if it allows programmatic access to the user's Videos, allowing the app to enumerate and access all files in the library without user interaction. This capability is typically used in movie-playback apps that make use of the entire Video library.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4058: Application Accesses Local Music Library

M4058 scans the mobile app to determine if it allows programmatic access to the user's Music, allowing the app to enumerate and access all files in the library without user interaction. This capability is typically used in jukebox apps that make use of the entire Music library.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4059: Application Supports Enterprise Authentication

M4059 scans the mobile app to determine if it supports enterprise authentication, which is typically used in line-of-business applications that connect to servers within an enterprise.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4060: Application Uses Shared User Certificates

M4060 scans the mobile app to determine if it uses shared user certificates, which enable an application to access software and hardware certificates, such as certificates stored on a Smart card. This capability is typically used for financial or enterprise apps that require a Smart card for authentication.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4061: Application Supports Private Network Access

M4061 scans the mobile app to determine if it supports private network access.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4062: Application Uses Web Camera

M4062 scans the mobile app to determine if it provides access to the video feed of a built-in camera or external web cam, which allows the app to capture photos and videos. The web cam capability only grants access to the video stream and not the audio stream.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4063: Application Uses Web Browser

M4063 scans the mobile app to determine if it uses the a web browser.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4064: Application Uses DirectX 11

M4064 scans the mobile app to determine if it uses DirectX 11.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4065: Application Uses Digital Compass

M4065 scans the mobile app to determine if it uses Digital Compass.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4066: Application Uses Xbox Service

M4066 scans the mobile app to determine if it uses the Xbox service.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4067: Application Uses Push Notification Service

M4067 scans the mobile app to determine if it uses the Push notification service.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4068: Application Uses Speech Recognition

M4068 scans the mobile app to determine if it uses speech recognition.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4069: Application Uses Local Ring Tones

M4069 scans the mobile app to determine if it uses local ring tones.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M4070: Other App Management

M4070 scans the application to determine if interacts with other applications.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses interact with other apps.
```

M4071: Wallet

M4071 scans the application to determine if it uses stored wallet cards.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses Wallet.
```

M4072: AllJoyn

M4072 scans the application to determine if it uses AllJoyn capability to discover and interact with devices on network.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses Wallet.
```

M4073: Supports User Profiles

M4073 scans the mobile app to determine if it uses Profile.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses Profile.
```

M4074: VOIP Service

M4074 scans the mobile app to determine if it uses VOIP service.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses VOIP.
```

M4075: Screen Projection

M4075 scans the mobile app to determine if it uses screen projection. This capability is required to use APIs in the DirectX namespace.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses Screen Projection.
```

M4076: Application Uses Device Unlock

M4076 scans the mobile app to determine if it uses device unlock capability for developer and enterprise sideloading scenarios.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses Device Unlock.
```

M4077: VPN Features

M4077 scans the mobile app to determine if it uses virtual private network (VPN) features.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Windows Mobile Risk Assessment

Severity

Error or Warning: If application requires to interact then it will be an error else if the application uses to interact then it will be a warning.

Message

```
The application requires/uses VPN features.
```

Android Mobile Risk Assessment Tests

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

The following Android mobile risk assessment tests are described in this section:

- M201: Application Requires Telephony
- M202: Application Requires Wi-Fi
- M203: Application Requires SMS Scheme
- M204: Application Uses a Camera
- M205: Application Uses an Auto-Focus Camera
- M206: Application Uses a Front-Facing Camera
- M207: Application Uses a Camera Flash
- M208: Application Uses a Video Camera
- M209: Application Uses an Accelerometer
- M210: Application Uses a Gyroscope
- M211: Application Uses Location Services
- M212: Application Uses GPS
- M213: Application Uses a Magnetometer
- M215: Application Uses the Microphone
- M220: Application Uses Peer-to-Peer via Bluetooth
- M221: Application Uses Bluetooth LE
- M230: Application Accesses the Address Book
- M231: Application Supports In-App Purchases

- M232: Application Supports Social Networking
- M235: Application Uses a Low-Latency Audio Pipeline
- M236: Application Uses the Consumer IR Capabilities on the Device
- M237: Application Uses the NFC Card Emulation Feature in the Device
- M238: Application Uses the Barometer in the Device
- M239: Application Uses the Device Light Sensor
- M240: Application Uses the Device Proximity Sensor
- M241: Application Uses the Step Device Detector
- M242: Application Requires Landscape Orientation
- M243: Application Requires Portrait Orientation
- M244: Application is Designed for a Television User Experience
- M245: Application Uses USB Feature
- M246: Application Accesses the Calendar
- M247: Application Uses Device Admin
- M248: Application Uses Heart Rate Sensor
- M249: Application Uses Relative Humidity Sensor
- M250: Application Uses Internet Access
- M251: Application Accesses Bookmarks
- M252: Application Uses External Storage
- M253: Uses Account Manager
- M254: Application Uses Kill Background Processes
- M255: Application Uses Profile
- M256: Application Uses Manage Documents
- M257: Application Uses IRTransmitter
- M258: Application Uses Body Sensors
- M259: Application Accesses Voice Mail
- M260: Android Pay
- M261: Fingerprint Scanner
- M262: Direct Share
- M263: VR Mode

M201: Application Requires Telephony

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M201 scans a mobile app for the presence of telephony usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M202: Application Requires Wi-Fi

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M202 scans the mobile app for the presence of wireless networking usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M203: Application Requires SMS Scheme

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M203 scans the mobile app for the presence of SMS scheme usage

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M204: Application Uses a Camera

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M204 scans the mobile app to determine if it uses the device camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M205: Application Uses an Auto-Focus Camera

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M205 scans the mobile app to determine if it uses the auto-focus camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M206: Application Uses a Front-Facing Camera

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M206 scans the mobile app to determine if it uses a front-facing camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M207: Application Uses a Camera Flash

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M207 scans the mobile app to determine if it uses the camera flash.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M208: Application Uses a Video Camera

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M208 scans the mobile app to determine if it uses the video camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M209: Application Uses an Accelerometer

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M209 scans the mobile app to determine if it uses the device accelerometer.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M210: Application Uses a Gyroscope

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M210 scans the mobile app to determine if it uses the device gyroscope.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M211: Application Uses Location Services

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M211 scans the mobile app to determine if it uses location services.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M212: Application Uses GPS

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M212 scans the mobile app to determine if it uses GPS.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M213: Application Uses a Magnetometer

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M213 scans the mobile app to determine if it uses the device magnetometer.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M215: Application Uses the Microphone

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M215 scans the mobile app to determine if it uses the device microphone.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M220: Application Uses Peer-to-Peer via Bluetooth

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M220 scans the mobile app to determine if it uses peer-to-peer connectivity via Bluetooth.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M221: Application Uses Bluetooth LE

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M221 scans the mobile app to determine if it uses Bluetooth LE.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M230: Application Accesses the Address Book

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M230 scans the mobile app to determine if it accesses the user's address book.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M231: Application Supports In-App Purchases

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M231 scans the mobile app to determine if it supports in-app purchases.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M232: Application Supports Social Networking

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M232 scans the mobile app to determine if it supports social networking.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M235: Application Uses a Low-Latency Audio Pipeline

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M235 scans the mobile app to determine if it uses a low-latency audio pipeline on the device and is sensitive to delays or lag in sound input or output.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M236: Application Uses the Consumer IR Capabilities on the Device

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M236 scans the mobile app to determine if it uses the consumer IR capabilities on the device.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M237: Application Uses the NFC Card Emulation Feature in the Device

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M237 scans the mobile app to determine if it uses the NFC card emulation feature in the device.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M238: Application Uses the Barometer in the Device

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M238 scans the mobile app to determine if it uses the barometer in the device.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M239: Application Uses the Device Light Sensor

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M239 scans the mobile app to determine if it uses the device light sensor.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M240: Application Uses the Device Proximity Sensor

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M240 scans the mobile app to determine if it uses the device proximity sensor.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M241: Application Uses the Step Device Detector

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M241 scans the mobile app to determine if it uses the device step detector.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M242: Application Requires Landscape Orientation

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M242 scans the mobile app to determine if it requires landscape orientation.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M243: Application Requires Portrait Orientation

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M243 scans the mobile app to determine if it requires portrait orientation.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M244: Application is Designed for a Television User Experience

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M244 scans the mobile app to determine if it is designed for a television user experience.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M245: Application Uses USB Feature

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M245 scans the mobile app to determine if it uses USB features.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M246: Application Accesses the Calendar

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M246 scans the mobile app to determine if it accesses the calendar.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M247: Application Uses Device Admin

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M247 scans the mobile app to determine if it uses the device administration feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M248: Application Uses Heart Rate Sensor

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M248 scans the mobile app to determine if it uses the heart rate sensor feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M249: Application Uses Relative Humidity Sensor

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M249 scans the mobile app to determine if it uses the relative humidity sensor feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M250: Application Uses Internet Access

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M250scans the mobile app to determine if it uses Internet access.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M251: Application Accesses Bookmarks

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M251 scans the mobile app to determine if it accesses bookmarks.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M252: Application Uses External Storage

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M252 scans the mobile app to determine if it uses external storage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M253: Uses Account Manager

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M253 scans the mobile app to determine if it uses the Account Manager feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M254: Application Uses Kill Background Processes

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M254 scans the mobile app to determine if it uses “kill background” processes.

AdminStudio examines the application’s metadata to determine if the feature is part of the application’s primary functionality, and whether it calls the feature’s APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application’s primary functionality, an Error is generated.
- If the application calls the feature’s APIs, a Warning is generated.

M255: Application Uses Profile

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M255 scans the mobile app to determine if it uses the profile feature.

AdminStudio examines the application’s metadata to determine if the feature is part of the application’s primary functionality, and whether it calls the feature’s APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application’s primary functionality, an Error is generated.
- If the application calls the feature’s APIs, a Warning is generated.

M256: Application Uses Manage Documents

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M256 scans the mobile app to determine if it uses the manage documents feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M257: Application Uses IRTransmitter

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M257 scans the mobile app to determine if it uses the IRTransmitter feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M258: Application Uses Body Sensors

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M258 scans the mobile app to determine if it uses body sensors.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M259: Application Accesses Voice Mail

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M259 scans the mobile app to determine if it accesses voice mail.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M260: Android Pay

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M260 scans the mobile app to determine if it uses Android Pay.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M261: Fingerprint Scanner

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M261 scans the mobile app to determine if it uses the fingerprint scanner.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M262: Direct Share

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M262 scans the mobile app to determine if it implements Direct Share, which enables someone to share something from just about any app on the mobile phone directly to one of their frequent contacts.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M263: VR Mode

Edition • *The Android Mobile Risk Assessment tests are included in AdminStudio Professional Edition Mobile and AdminStudio Enterprise Edition Mobile.*

M263 scans the mobile app to determine if it uses VR (virtual reality) mode.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Android Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

Apple Mobile Risk Assessment Tests

The following Apple mobile risk assessment tests are described in this section:

- M101: Application Requires Telephony
- M102: Application Requires Wi-Fi
- M103: Application Requires SMS Scheme
- M104: Application Uses a Camera
- M105: Application Uses an Auto-Focus Camera
- M106: Application Uses a Front-Facing Camera
- M107: Application Uses a Camera Flash
- M108: Application Uses a Video Camera
- M109: Application Uses an Accelerometer
- M110: Application Uses a Gyroscope
- M111: Application Uses Location Services
- M112: Application Uses GPS
- M113: Application Uses a Magnetometer
- M114: Application Uses Gamekit
- M115: Application Uses the Microphone
- M116: Application Uses OpenGL ES 1.1
- M117: Application Uses OpenGL ES 2.0
- M118: Application Uses ARMv6
- M119: Application Uses ARMv7
- M120: Application Uses Peer-to-Peer via Bluetooth
- M121: Application Uses Bluetooth LE
- M122: Application Uses Safari
- M123: Application Runs Only on an iPad
- M124: Application Uses Persistent Wi-Fi
- M125: Application Runs Only on an iPhone or iPod
- M126: Application Can Share Files Through iTunes
- M127: Application Can Interface Enumerated External Devices
- M128: Application Can Open a Specific File Type
- M129: Application Can Save a Specific File Type
- M130: Application Can Copy/Paste a Specific File Type
- M131: Application Supports Location Tracking
- M132: Application Supports Ad Networks
- M133: Application Accesses the Address Book

- M134: Application Supports In-App Purchases
- M135: Application Supports Social Networking
- M136: Application Supports User Identity
- M137: Application Accesses Local Pictures
- M138: Application Accesses the Calendar
- M139: Application Uses OpenGL ES 3.0
- M140: Application Accesses HealthKit
- M141: Application Uses Metal
- M142: Application Uses Local Authentication (Touch ID)
- M143: Application Uses HomeKit
- M144: Application Uses CloudKit
- M145: Application Uses Barometer
- M146: Application Uses PassKit (ApplePay)
- M147: Application Uses App-Extension Custom Keyboard
- M148: Application Uses App-Extension Document Picker
- M149: Application Uses App-Extension File Provider
- M150: Application Uses App-Extension Photo Editing
- M151: Application Uses App-Extension Share
- M152: Application Uses App-Extension Today
- M153: 3D Touch
- M154: ARM 64
- M155: Siri Voice Assistance (SiriKit)

M101: Application Requires Telephony

M101 scans the mobile app for the presence of telephony usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.

- If the application calls the feature's APIs, a Warning is generated.

M102: Application Requires Wi-Fi

M102 scans the mobile app for the presence of wireless networking usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M103: Application Requires SMS Scheme

M103 scans the mobile app for the presence of SMS scheme usage.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M104: Application Uses a Camera

M104 scans the mobile app to determine if it uses the device camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M105: Application Uses an Auto-Focus Camera

M105 scans the mobile app to determine if it uses the auto-focus camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M106: Application Uses a Front-Facing Camera

M106 scans the mobile app to determine if it uses a front-facing camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M107: Application Uses a Camera Flash

M107 scans the mobile app to determine if it uses the camera flash.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M108: Application Uses a Video Camera

M108 scans the mobile app to determine if it uses the video camera.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M109: Application Uses an Accelerometer

M109 scans the mobile app to determine if it uses the device accelerometer.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M110: Application Uses a Gyroscope

M110 scans the mobile app to determine if it uses the device gyroscope.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M111: Application Uses Location Services

M111 scans the mobile app to determine if it uses location services.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M112: Application Uses GPS

M112 scans the mobile app to determine if it uses GPS.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M113: Application Uses a Magnetometer

M113 scans the mobile app to determine if it uses the device magnetometer.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M114: Application Uses Gamekit

M114 scans the mobile app to determine if it uses the game center.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M115: Application Uses the Microphone

M115 scans the mobile app to determine if it uses a microphone.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.

- If the application calls the feature's APIs, a Warning is generated.

M116: Application Uses OpenGL ES 1.1

M116 scans the mobile app to determine if it uses accelerated 3D graphics (OpenGL ES 1.1).

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M117: Application Uses OpenGL ES 2.0

M117 scans the mobile app to determine if it uses accelerated 3D graphics (OpenGL ES 2.0).

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M118: Application Uses ARMv6

M118 scans the mobile app to determine if it uses the ARMv6 instruction set.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M119: Application Uses ARMv7

M119 scans the mobile app to determine if it uses the ARMv7 instruction set.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M120: Application Uses Peer-to-Peer via Bluetooth

M120 scans the mobile app to determine if it uses peer-to-peer connectivity via Bluetooth.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M121: Application Uses Bluetooth LE

M121 scans the mobile app to determine if it uses Bluetooth LE.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M122: Application Uses Safari

M122 scans the mobile app to determine if it uses the Safari web browser.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M123: Application Runs Only on an iPad

M123 scans the mobile app to determine if it only can be installed on an iPad.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M124: Application Uses Persistent Wi-Fi

M124 scans the mobile app to determine if it uses persistent Wi-Fi.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M125: Application Runs Only on an iPhone or iPod

M125 scans the mobile app to determine if it only can be installed on an iPhone/iPod.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M126: Application Can Share Files Through iTunes

M126 scans the mobile app to determine if it can share files through iTunes.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M127: Application Can Interface Enumerated External Devices

M127 scans the mobile app to determine if it can interface with external devices.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M128: Application Can Open a Specific File Type

M128 scans the mobile app to determine if it opens a specific file type.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M129: Application Can Save a Specific File Type

M129 scans the mobile app to determine if it can save a specific file type.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.

- If the application calls the feature's APIs, a Warning is generated.

M130: Application Can Copy/Paste a Specific File Type

M130 scans the mobile app to determine if it can copy/paste a specific file type.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M131: Application Supports Location Tracking

M131 scans the mobile app to determine if it uses location tracking.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M132: Application Supports Ad Networks

M132 scans the mobile app to determine if it supports ad networks.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M133: Application Accesses the Address Book

M133 scans the mobile app to determine if it accesses the user's address book.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M134: Application Supports In-App Purchases

M134 scans the mobile app to determine if it supports in-app purchases.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M135: Application Supports Social Networking

M135 scans the mobile app to determine if it supports social networking.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M136: Application Supports User Identity

M136 scans the mobile app to determine if it supports user identification.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M137: Application Accesses Local Pictures

M137 scans the mobile app to determine if it accesses the user's local pictures.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M138: Application Accesses the Calendar

M138 scans the mobile app to determine if it accesses the calendar.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M139: Application Uses OpenGL ES 3.0

M139 scans the mobile app to determine if it uses accelerated 3D graphics (OpenGL ES 3.0).

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M140: Application Accesses HealthKit

M140 scans the mobile app to determine if it uses HealthKit, which allows apps that provide health and fitness services to share their data with the iOS Health app and with each other.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M141: Application Uses Metal

M141 scans the mobile app to determine if it uses Metal, a low-level, low-overhead hardware-accelerated graphics API.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M142: Application Uses Local Authentication (Touch ID)

M142 scans the mobile app to determine if it uses Touch ID, which enables users to unlock their phone using their fingerprint.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M143: Application Uses HomeKit

M143 scans the mobile app to determine whether it uses HomeKit, which is a framework for communicating with and controlling connected accessories in a user's home.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M144: Application Uses CloudKit

M144 scans the mobile app to determine whether it uses CloudKit, which is a way to move structured data between an app and iCloud.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M145: Application Uses Barometer

M145 scans the mobile app to determine whether it uses the iOS Barometer feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M146: Application Uses PassKit (ApplePay)

M146 scans the mobile app to determine whether it uses ApplePay payment platform.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M147: Application Uses App-Extension Custom Keyboard

M147 scans the mobile app to determine if it uses the iOS custom keyboard feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M148: Application Uses App-Extension Document Picker

M148 scans the mobile app to determine if it uses the Document Picker view controller.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M149: Application Uses App-Extension File Provider

M140 scans the mobile app to determine if it uses the iOS File Provider feature, which allows other apps to access the documents managed by this app.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M150: Application Uses App-Extension Photo Editing

M150 scans the mobile app to determine whether it uses the iOS photo editing feature.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M151: Application Uses App-Extension Share

M151 scans the mobile app to determine whether it uses the iOS Share feature, which give users a convenient way to share content with other entities, such as social sharing websites or upload services.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M152: Application Uses App-Extension Today

M152 scans the mobile app to determine whether it uses the iOS Today feature, which enables the app to provide widgets to give users quick access to information that's important at that moment, such as stock prices or weather conditions.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M153: 3D Touch

M153 scans the mobile app to determine whether it uses 3D Touch, a new dimension to the Multi-Touch. 3D Touch senses how deeply users press the display, letting them perform additional functions with apps and games.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M154: ARM 64

M154 scans the mobile app to determine if it uses the ARM64 instruction set.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

M155: Siri Voice Assistance (SiriKit)

M155 scans the mobile app to determine if it uses the SiriKit, a framework that allows the app to communicate with the Siri service using voice commands.

Test Group/Test Category

Best Practices/Mobile Risk Assessment/Apple Mobile Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

Desktop Risk Assessment Tests

The following categories of desktop risk assessment tests are described in this section:

- Mac OS Risk Assessment
- Windows Risk Assessment

Mac OS Risk Assessment

The following Mac OS risk assessment tests are described in this section:

- D001: App Transport Security (ATS)
- D002: ATS Exemptions
- D003: Insecure HTTP Connections
- D004: Forward Secrecy Exemptions
- EOLMAC001: Java Versions Required
- EOLMAC002: Java Installer Versions

D001: App Transport Security (ATS)

D001 scans the app to determine if App Transport Security (ATS) is disabled.

Starting in OS X v10.11, a new security feature called App Transport Security (ATS) is available to apps and is enabled by default. It improves the privacy and data integrity of connections between an app and web services by enforcing additional security requirements for HTTP-based networking requests. Specifically, with ATS enabled, HTTP connections must use HTTPS (RFC 2818). Attempts to connect using insecure HTTP fail. Furthermore, HTTPS requests must use best practices for secure communications.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

D002: ATS Exemptions

D001 scans the app to determine if App Transport Security (ATS) is disabled.

Starting in OS X v10.11, a new security feature called App Transport Security (ATS) is available to apps and is enabled by default. It improves the privacy and data integrity of connections between an app and web services by enforcing additional security requirements for HTTP-based networking requests. Specifically, with ATS enabled, HTTP connections must use HTTPS (RFC 2818). Attempts to connect using insecure HTTP fail. Furthermore, HTTPS requests must use best practices for secure communications.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

D003: Insecure HTTP Connections

D003 scans the app to determine if the app allows insecure HTTP connections to any of the domains.

Starting in OS X v10.11, a new security feature called App Transport Security (ATS) is available to apps and is enabled by default. It improves the privacy and data integrity of connections between an app and web services by enforcing additional security requirements for HTTP-based networking requests. Specifically, with ATS enabled, HTTP connections must use HTTPS (RFC 2818). Attempts to connect using insecure HTTP fail. Furthermore, HTTPS requests must use best practices for secure communications.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.

- If the application calls the feature's APIs, a Warning is generated.

D004: Forward Secrecy Exemptions

D004 scans the app to determine if Forward Secrecy is exempted for any of the domains.

The security of communications transmitted across the Internet can be improved by using public key cryptography. However, if the public and private keys used in those communicators are compromised, it can reveal the data exchanged in that session as well as the data exchanged in previous sessions.

The concept of Forward Secrecy (FS) is the property that ensures that a session key derived from a set of long-term public and private keys will not be compromised if one of the (long-term) private keys is compromised in the future.

AdminStudio examines the application's metadata to determine if the feature is part of the application's primary functionality, and whether it calls the feature's APIs.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- If the application requires the feature as part of the application's primary functionality, an Error is generated.
- If the application calls the feature's APIs, a Warning is generated.

EOLMAC001: Java Versions Required

EOLMAC001 scans the application to determine whether it has a dependency on a Java version (required for the package after installation) that has been designated as end of life/end of support. If the application does have such a dependency, an error is generated. Also, a warning is generated if the application depends on a version of Java (required for the package after installation) that is not the most recent version.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- **Error**—Generated if application is dependent upon a Java version (required for the package after installation) that has been designated as end-of-life/end-of-support.
- **Warning**—Generated if application depends upon a version of Java (required for the package after installation) that is not the most recent version.

Rule Background

An application which requires an EOL Java version needs to be handled before deploying, for instance, by updating the application to it's higher/latest version. The application is tested to check if it requires a Java version which is EOL.

Note • *Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the ASCMPackageJavaSummary table in the* ***Tables View****.*

EOLMAC002: Java Installer Versions

EOLMAC002 scans the application to determine whether it has a dependency on a Java version (required during package installation) that has been designated as end of life/end of support. If the application does have such a dependency, an error is generated. Also, a warning is generated if the application depends on a version of Java (required during package installation) that is not the most recent version.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/macOS Risk Assessment

Severity

- **Error**—Generated if application is dependent upon a Java version (required during package installation) that has been designated as end-of-life/end-of-support.
- **Warning**—Generated if application depends upon a version of Java (required during package installation) that is not the most recent version.

Rule Background

An application which installs an EOL Java version is risky. The application is tested to check if it installs a Java version which is EOL.

Note • *Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the ASCMPackageJavaSummary table in the* ***Tables View****.*

Windows Risk Assessment

The following Windows risk assessment tests are described in this section:

- EOLWIN001: Java Versions Required
- EOLWIN002: Java Installer Versions
- VRAWIN001: Checks for Log4j jar files

EOLWIN001: Java Versions Required

EOLWIN001 scans the application to determine whether it has a dependency on a Java version (required for the package after installation) that has been designated as end of life/end of support. If the application does have such a dependency, an error is generated. Also, a warning is generated if the application depends on a version of Java (required for the package after installation) that is not the most recent version.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/Windows Risk Assessment

Severity

- **Error**—Generated if application is dependent upon a Java version (required for the package after installation) that has been designated as end-of-life/end-of-support.
- **Warning**—Generated if application depends upon a version of Java (required for the package after installation) that is not the most recent version.

Note • *Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the ASCMPackageJavaSummary table in the* ***Tables View****.*

EOLWIN002: Java Installer Versions

EOLWIN002 scans the application to determine whether it has a dependency on a Java version (required during package installation) that has been designated as end of life/end of support. If the application does have such a dependency, an error is generated. Also, a warning is generated if the application depends on a version of Java (required during package installation) that is not the most recent version.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/Windows Risk Assessment

Severity

- **Error**—Generated if application is dependent upon a Java version (required during package installation) that has been designated as end-of-life/end-of-support.
- **Warning**—Generated if application depends upon a version of Java (required during package installation) that is not the most recent version.

Note • *Java dependency information is extracted during package import of Windows and macOS desktop applications, and can be viewed by opening the ASCMPackageJavaSummary table in the* ***Tables View****.*

VRAWIN001: Checks for Log4j jar files

VRAWIN001 scans the package for the presence of log4j jar files. When a log4j jar file is found in a package, the file name, its version and path to the file in the installation directory of the application is displayed in the warning message.

Test Group/Test Category

Risk Assessment/Desktop Risk Assessment/Windows Risk Assessment

Severity

- **Warning**—Generated if Log4j jar files detected in a package.

Note • *Log4j jar files extracted during package import can be viewed by opening the cstblPEApplicationData and csmsiFile tables in the* ***Tables View****.*

Application Conflicts Tests

Edition • *The Package Data Conflicts tests are included in the AdminStudio Professional and Enterprise Editions.*

The following subcategories of application conflicts tests are available:

- Package Data Conflicts Tests
- Microsoft App-V Conflict Tests

Package Data Conflicts Tests

Edition • *These tests are included in the AdminStudio Professional and Enterprise Editions.*

The following package data conflict tests are described in this section:

Table 17-5 • Package Data Conflict Tests

Category	Tests
Components	• ACE02: Identical Components with Different Destinations • ACE09: Identical Merge Modules • ACE30: Different Components that Install the Same Key File
File Extensions	• ACE17: Duplicate File Extension-Verb Combinations in Different Components
Files	• ACE03: New or Missing Files in Identical Components • ACE07: Same File in Different Components • ACE08: Identical Components with Different Versions of a File • ACE12: Files from Merge Modules • ACE23: Duplicate Files with Different Sizes, Versions, or Languages

Table 17-5 • Package Data Conflict Tests

Category	Tests
INI Files	• ACE14: Duplicate INI File in Different Components • ACE21: Conflicts Between Entries in the IniFile and File Tables • ACE22: IniFile and File Table Entries for the Same File
NT Services	• ACE16: Duplicate Services in Different Components
ODBC Resources	• ACE15: Duplicate ODBC Entries in Different Components
Product Properties	• ACE18: Identical Package Codes for Different Packages • ACE19: Identical Product Codes for Different Packages • ACE20: Identical Upgrade Codes for Different Packages
Registry	• ACE10: Conflicts in Registry Root, Key, and Name Combinations • ACE24: Duplicate Registry Entries with Different Data Types or Values
Shortcuts	• ACE13: Shortcut Conflicts

ACE02: Identical Components with Different Destinations

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE02 checks whether components in different packages that have matching ComponentId values also have identical destination paths.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Components

Severity

Warning

Message

```
The destination [PATH1] for the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    destination for the component [COMPONENT1] in the package [PACKAGE2]. The correct destination should
    be [PATH2].
```

Background

If components with the same ComponentId have different destination paths, ACE02 fails.

Resolution

Automatic Fix (CARD02)

CARD02 automatically sets the destination path of the component in the source package to match that of the component in the target package. To do this, CARD02 runs the following query and replaces the Directory_ column with a run-time translation of the target's component (csFullPath) path:

```
MsiDBUtils::GetDirectoryTargetPathKey.
SELECT `Directory_` FROM `Component`
     WHERE ComponentId='Source ComponentId'
```

ACE03: New or Missing Files in Identical Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE03 checks whether components in different packages that have matching ComponentId values also contain the same files.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Files

Severity

Warning

Message

```
The file(s) [FILENAME] in the component [COMPONENT1] in the package [PACKAGE1] are either new to or
    missing from the component [COMPONENT1] in the package [PACKAGE2].
```

Background

If components with the same ComponentId do not contain the same files (either files are missing or they are different versions), ACE03 fails.

Resolution

Manual Fix

Use InstallShield Editor to create a Windows Installer Transform (.mst) file for the source package MSI so that the component in the source package contains the same files as the component in the target package.

Task ***To resolve this issue:***

1. Open the MSI package in InstallShield Editor.
2. Add or modify the components as necessary so that the file(s) is present in both components and the versions match in both MSI packages. To accomplish this:

a. In the View List under **Organization**, click **Components**.

b. In the **Components** explorer, find the component that needs to be modified, expand the list under the component, and select **Files**. A list of the files included with that component is displayed.

c. To delete a file that is not present in the other component, right-click the file and then click **Delete**.

   To add a new file to match the other component, right-click anywhere in the **Files** list, click **Add**, and add the correct file.

3. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.

4. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE03 again.

ACE07: Same File in Different Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE07 checks for the existence of the same file in components with different ComponentIds.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Files

Severity and Messages

ACE07 reports four types of errors, depending on the various combinations of source and target files and whether they originated in a merge module:

Table 17-6 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Message
Not Merge Module	**Merge Module**	Warning	No	The file [FILENAME] in the component [COMPONENTNAME] in the package [PACKAGENAME] is identical to the merge module installed file in the component [COMPONENTNAME] in the package [PACKAGENAME]. Confirm this error by running this package against ACE12.
Not Merge Module	**Not Merge Module**	Error	Yes	The file [FILENAME] is identical in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs.

Table 17-6 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Message
Merge Module	**Merge Module**	Error	No	The file [FILENAME] is identical and installed by merge modules in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs. Run ACE12 to determine which merge module is most appropriate to use.
Merge Module	**Not Merge Module**	Error	No	The file [FILE_NAME] in the merge module installed component [COMPONENT_NAME] in the package [PACKAGE_NAME] is identical to the file in the component [COMPONENT_NAME] in the package [PACKAGE_NAME]. Confirm this error by running the [PACKAGE_NAME] package against the ACE12.

Background

ACE07 reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module.

Resolutions

Manual Fix

If both the source and target files did not originate in a Merge Module, confirm the error by running the package against ACE12 and then, based upon the results, decide how to proceed.

Automatic Fix (CARD07)

If both the source and target files originated in a Merge Module, you can use CARD07 to resolve the issue. CARD07 changes the ComponentId value of the source package to match that of the target package. To do this, CARD07 runs the following query against the source package and then updates the ComponentId value with the ComponentId value from the target package:

```
SELECT `ComponentId` FROM `Component` WHERE
     Component` = 'Source Package ComponentId'
```

ACE08: Identical Components with Different Versions of a File

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE08 identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Files

Severity

Warning

Message

```
The file version [VERSION_NUMBER] of file [FILENAME] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same file in the component [COMPONENT2] in the package [PACKAGE2].
```

Background

If components with identical ComponentIds contain different versions of a file, ACE08 fails.

Resolution

Manual Fix

Change the file versions to match those of the Target component, or change the file versions in the Source component.

Task ***To resolve this issue:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, there are two options that can be used to resolve the issue:

   Option #1: Change the Component Code. To accomplish this:

   a. In the View List under **Organization**, click **Components**.

   b. In the **Components** explorer, click the component that needs to be modified.

   c. In the **Component Code**, enter the appropriate value.

   Option #2: Replace the files so that the versions match in both MSI packages. To accomplish this:

   a. In the View List under **Organization**, click **Components**.

   b. In the **Components** explorer, find the component that needs to be modified, expand the listing under the component, and select **Files**. The list of files included with that component is displayed.

   c. To delete a file with the wrong version, right-click the file, and then click **Delete**.

      To add the correct version of a file, right-click anywhere in the **Files** list, click **Add**, and add the new file with the correct version.
3. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
4. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE08 again.

ACE09: Identical Merge Modules

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE09 checks whether merge modules with identical ComponentId values are identical.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Components

Severity

Warning

Message

```
The merge module, [MERGE_MODULE_NAME], version [VERSION_NUMBER1], conflicts with the same merge module,
    version [VERSION_NUMBER2], in another package.
```

Background

If merge modules with identical ComponentIds are different, a warning is generated.

Resolution

Manual Fix

Obtain the latest version of the merge module and rebuild the MSI.

Task ***To resolve this issue:***

1. Obtain the latest version of the merge module that is displayed in the ACE message. The merge module may be available for download at the vendor's web site.
2. Open the MSI package in **InstallShield Editor**.
3. Place the new merge module in one of the Merge Module Locations that are specified on the **Merge Modules** tab of the Application Catalog **Options** dialog box. This allows the merge module to be displayed in the **Redistributables** view of the InstallShield Editor.
4. In the View List under **Application Data**, click **Redistributables**.
5. Clear the check box of the old merge module that is causing the conflict, and select the check mark of the new merge module in its place.
6. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
7. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE09 again.

ACE10: Conflicts in Registry Root, Key, and Name Combinations

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE10 checks for the existence of identical root/key/name registry combinations in components with different ComponentId values.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Registry

Severity

Warning

Message

```
The registry entry [REGISTRY_ENTRY] [REGISTRY_KEY] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same registry entry in the component [COMPONENT1] in the package
    [PACKAGE2].
```

Background

If the same root/key/name registry combination is in more than one component, a warning is generated.

Conditions When an ACE10 Error Can Be Ignored

ACE10 uses data from the Registry table in the .msi package to check for identical registry root/key/name combinations in different components. However, there may be situations in which ACE10 reports an error unnecessarily. Windows Installer supports a grammar for the Registry table in which every time that the value in the Value field is preceded or terminated by the sequence tilde '[~]', the registry value is appended or prefixed, respectively, to the existing registry value. This sort of operation may be perfectly acceptable if the applications in question are modifying a common registry key in a manner consistent with its purpose.

Decide individually if an ACE10 error is valid, but consider checking the Registry Value field, since its contents may prove useful in helping you decide.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open the MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.
3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   If it is unclear what value you should use, do the following:

   a. In InstallShield Editor, open the package that will not be edited.

   b. In the View List under **Additional Tools**, click **Direct Editor**.

c. In the **Tables** explorer, click the **Component** table.

d. Search for the component name that is included in the message, and note the value that is listed in the **ComponentId** column for that component. This is the component code that you should use for the component in the other package.

5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.

6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE10 again.

ACE12: Files from Merge Modules

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE12 checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. (Before running ACE12, you should import all merge modules that you are likely to use at your organization into the Application Catalog database.)

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Files

Severity and Messages

ACE12 reports an error if the file in question originated from a merge module and the merge module was found in the Application Catalog. If the merge module was not found in the Application Catalog, ACE12 reports a warning:

Table 17-7 • ACE12 Output Summary

Package File	Severity	Message
Not from Merge Module	Error	The file [FILENAME] in the component [COMPONENT1] in the package [PACKAGE1] should be replaced with the merge module [MERGE_MODULE_NAME].
From Merge Module	Warning	The [FILENAME] file originating from the [ModuleID] Merge Module in package [PACKAGE1] is a candidate to be replaced with the [MERGE_MODULE_NAME]. However, the [ModuleID] Merge Module is not in the Application Catalog which makes proper evaluation impossible.

Background

ACE12 reports a warning if the file in question originated from a merge module and the merge module was found in the Application Catalog. If the merge module was not found in the Application Catalog, ACE12 reports an error.

Using merge modules is always preferable as a way to install files in a consistent fashion.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open the MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Setup Design**.
3. In the **Setup Design** explorer, find the component that is listed in the message, expand the list under the component, and select **Files**. A list of the files included with that component is displayed.
4. Right-click the file that was displayed in the error message, and then click **Delete**.
5. Take note of the feature that contains the component.
6. In the View List under **Application Data**, click **Redistributables**.
7. Select the check box of the appropriate merge module.
8. In the **Conditional Installation** pane, select check box of the feature that contains the component.
9. Rebuild the package.
10. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE12 again.

ACE13: Shortcut Conflicts

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE13 checks for the existence of the same shortcut within different packages in components with different ComponentIds.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Shortcuts

Severity

Warning

Message

```
The shortcut [SHORTCUT_NAME] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    same shortcut in the component [COMPONENT2] in the package [PACKAGE2].
```

Background

If the same shortcut exists in more than one component, ACE13 fails.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open the MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.
3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   If it is unclear what value you should use, do the following:

   a. In InstallShield Editor, open the package that will not be edited.

   b. In the View List under **Additional Tools**, click **Direct Editor**.

   c. In the **Tables** explorer, click the **Component** table.

   d. Search for the component name that is included in the message, and note the value that is listed in the **ComponentId** column for that component. This is the component code that you should use for the component in the other package.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE13 again.

ACE14: Duplicate INI File in Different Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE14 checks for the existence of components with different ComponentIds that modify the same INI file entry, such as the [File Name/Section/Key/Value] entry.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/INI Files

Severity

Warning

Message

```
The INI file entry [INI_FILE_ENTRY] in the file [INI_FILE_NAME] in the component [COMPONENT1] in the
    package [PACKAGE1] conflicts with the same INI file entry in the component [COMPONENT1] in the
    package [PACKAGE2].
```

Background

If the same INI file entry is modified by different components, ACE14 fails.

Resolution

Manual Fix

To resolve ACE14, change the ComponentId of the Source component to match the ComponentId of the Target component.

Task ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.
3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   If it is unclear what value you should use, do the following:

   a. In InstallShield Editor, open the package that will not be edited.

   b. In the View List under **Additional Tools**, click **Direct Editor**.

   c. In the **Tables** explorer, click the **Component** table.

   d. Search for the component name that is included in the message, and note the value that is listed in the **ComponentId** column for that component. This is the component code that you should use for the component in the other package.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE14 again.

ACE15: Duplicate ODBC Entries in Different Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE15 checks for the existence of identical ODBC entries in components with different ComponentId values.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/ODBC Resources

Severity

Error

Message

```
The ODBC entry [ODBC_ENTRY] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    same ODBC entry in the component [COMPONENT1] in the package [PACKAGE2].
```

Background

If identical ODBC entries exist in components with different ComponentId values, ACE15 fails.

Resolution

Automatic Fix (CARD15)

CARD15 changes the ComponentId value of the source package to match that of the target package. To do this, CARD15 runs the following query against the source package and then updates the ComponentId value with the ComponentId value from the target package:

```
SELECT `ComponentId` FROM
Component` WHERE `Component` = 'Source Package ComponentId'
```

ACE16: Duplicate Services in Different Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE16 checks for the existence of identical Windows services in components with different ComponentIds.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/NT Services

Severity

Warning

Message

```
The NT service [SERVICE1] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    same NT Service in the component [COMPONENT1] in the package [PACKAGE2].
```

Background

If identical services are present within different components, ACE16 fails.

Resolution

Manual Fix

Change the ComponentId of the Source component to match the ComponentId of the Target component.

Task ***To resolve this issue:***

1. Open the package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.
3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   If it is unclear what value you should use, do the following:

   a. In InstallShield Editor, open the package that will not be edited.

   b. In the View List under **Additional Tools**, click **Direct Editor**.

   c. In the **Tables** explorer, click the **Component** table.

   d. Search for the component name that is included in the message, and note the value that is listed in the **ComponentId** column for that component. This is the component code that you should use for the component in the other package.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE16 again.

ACE17: Duplicate File Extension-Verb Combinations in Different Components

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE17 checks for identical file extension/verb combinations in components with different ComponentIds.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/File Extensions

Severity

Warning

Message

```
The verb [VERB_NAME] in extension [EXTENSION_NAME] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same verb & extension in the component [COMPONENT1] in the package
    [PACKAGE2].
```

Background

If identical file Extension/Verb combinations exist in components with different ComponentIds, ACE17 fails.

Resolution

Manual Fix

To resolve ACE17, change the ComponentId of the Source component to match the ComponentId of the Target component.

Task ***To resolve this issue:***

1. Open the transform file or MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.
3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   If it is unclear what value you should use, do the following:

   a. In InstallShield Editor, open the package that will not be edited.

   b. In the View List under **Additional Tools**, click **Direct Editor**.

   c. In the **Tables** explorer, click the **Component** table.

   d. Search for the component name that is included in the message, and note the value that is listed in the **ComponentId** column for that component. This is the component code that you should use for the component in the other package.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE17 again.

ACE18: Identical Package Codes for Different Packages

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE18 checks the package code to see if it is unique.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Product Properties

Severity

Error

Message

```
The package code [PACKAGE_CODE] in the package [PACKAGE1] is the same as the package code in the package
    [PACKAGE2].
```

Background

If the package code is identical to any other package code in the Application Catalog, ACE18 fails.

Resolution

Manual Fix

Manually change the package code in one of the packages.

Automatic Fix

None.

ACE19: Identical Product Codes for Different Packages

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE19 checks the product code to see if it is unique.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Product Properties

Severity

Error

Message

```
The product code [PRODUCT_CODE] in the package [PACKAGE1] is the same as the product code in the package
    [PACKAGE2].
```

Background

If the product code is identical to any other product code in the Application Catalog, ACE19 fails.

Resolution

Manual Fix

Manually change the package code in one of the packages.

Automatic Fix

None.

ACE20: Identical Upgrade Codes for Different Packages

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE20 checks the upgrade code to see if it is unique.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Product Properties

Severity

Error

Message

```
The upgrade code [UPGRADE_CODE] in the package [PACKAGE1] is the same as the upgrade code in the package
    [PACKAGE2].
```

Background

If the upgrade code is not unique, ACE20 fails.

Resolution

Manual Fix

Change the upgrade code in one of the packages.

Automatic Fix

None.

ACE21: Conflicts Between Entries in the IniFile and File Tables

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE21 checks entries in the IniFile table to see if they conflict with similar entries in the File table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/INI Files

Severity

Warning

Message

```
The INI file [INI_FILENAME] installed using the IniFile table to the destination [PATH_NAME] in the
    package [PACKAGE1.MSI] conflicts with the same file in the File table in the package [PACKAGE2].
```

Background

If an entry in the IniFile table of the first MSI package duplicates a file name in a component of the File table of the second MSI package, and both are set to be installed to the same destination, ACE21 fails.

Resolution

Manual Fix

To resolve ACE21, ensure that identically named INI files with identical destinations are identical in both the File and IniFile tables. This may involve adding or deleting sections or changing values in the INI files.

Task ***To resolve this issue:***

1. In InstallShield Editor, open a transform file or MSI package that has an entry in the IniFile table.
2. In the View List under **System Configuration**, click **INI Files Changes**.
3. In the **INI Files** explorer, find the INI file entry that is listed in the message.
4. Analyze the sections, keywords, and values in this INI file and compare them to the INI file in the other MSI package. Edit the sections, keywords, and values in this INI file so that they match those of the second INI file. This may require adding and/or deleting data in the **INI File Changes** view.

   To retrieve the INI file from the other MSI package, you may need to extract it from the MSI file itself or a cabinet file if the files are compressed on the source media.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE21 again.

ACE22: IniFile and File Table Entries for the Same File

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE22 checks file name/target directory pairs in the File table to see if they conflict with similar entries in the IniFile table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/INI Files

Severity

Warning

Message

```
The INI file [INI_FILENAME] installed using the File table to the destination [PATH_NAME] in the package
    [PACKAGE3.MSI] conflicts with the same file in the IniFile table in the package [PACKAGE4].
```

Background

If a file name/target directory pair in a component in the File table is also listed as an IniFile table entry, ACE22 fails.

Resolution

Manual Fix

To resolve ACE22, ensure that identically named INI files with identical destinations are identical in both the File and IniFile tables. This may involve adding or deleting sections or changing values in the INI files.

Task ***To resolve this issue:***

1. In InstallShield Editor, open a transform file or MSI package that has an entry in the IniFile table.
2. In the View List under **System Configuration**, click **INI Files Changes**.
3. In the **INI Files** explorer, find the INI file entry that is listed in the message.
4. Analyze the sections, keywords, and values in this INI file and compare them to the INI file in the other MSI package. Edit the sections, keywords, and values in this INI file so that they match those of the second INI file. This may require adding and/or deleting data in the **INI File Changes** view.

   To retrieve the INI file from the other MSI package, you may need to extract it from the MSI file itself or a cabinet file if the files are compressed on the source media.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE22 again.

ACE23: Duplicate Files with Different Sizes, Versions, or Languages

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE23 identifies file duplication between source and target packages. ACE23 checks whether files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Files

Severity

Warning

Message

```
The file [EXECUTABLE_FILENAME, SIZE, VERSION, LANGUAGE] installed to location [PATHNAME] by component
    [COMPONENT1] in Package [PACKAGE1] conflicts with the same file in component [COMPONENT2] in Package
    [PACKAGE2].
```

Background

If a file with the same name and destination directory is found in both the source and target package (and, in the case of an MSI package comparison, the packages have different ComponentId values), but the file has a different size, version, or language, ACE23 fails. If the source and target packages are MSI packages and they have the same ComponentId value, no error is reported.

Resolution

This issue requires a manual resolution. Investigate the issue and decide which file has precedence. If the source file has precedence over the target file, remove the target file. Use one of the following solutions.

Manual Fix: Solution 1

This fix involves replacing the file in the source package with the same file that is in the target package.

Task ***To replace the file in the source package with the same file that is in the target package:***

1. Retrieve a copy of the file from the operating system in which the target package was taken.
2. Open a transform file or MSI package in InstallShield Editor.
3. In the View List under **Application Data**, click **Files and Folders**.
4. Find the file that is listed in the message, and replace it with the one that you retrieved from the operating system.
5. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
6. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE23 again.

Manual Fix Solution 2

This fix involves changing the destination of the component that contains the file in the target package.

Task ***To change the destination of the component that contains the file in the target package:***

1. Open a transform file or MSI package in InstallShield Editor.
2. In the View List under **Organization**, click **Components**.

3. In the **Components** explorer, find the component that is listed in the message.
4. In the **Component Code** setting, change the value to match the component code (ComponentId) of the component in the other project.

   To quickly find the component name, open the File table in the Direct Editor view and search for the file name. Then, check the **Component** column for the component name.
5. In the **Destination** setting, change the value to a new destination.

   Changing the destination may cause the application to break. Before changing the destination, verify that the application will still work with the new destination.
6. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
7. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE23 again.

ACE24: Duplicate Registry Entries with Different Data Types or Values

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

ACE 24 checks whether registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target packages, but the registry entry has a different data type or value, ACE24 fails.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Registry

Severity

Warning

Message

```
The registry entry [REGISTRY_ENTRY] in [PACKAGE1] conflicts with the same registry entry in the
    [COMPONENT] in [PACKAGE2].
```

Background

If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target packages, but the registry entry has a different data type or value, ACE24 fails.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open a transform file or MSI package in InstallShield Editor.
2. In the View List under **System Configuration**, click **Registry**.
3. Find the registry value that is listed in the message, and replace that value with the same registry value and data type that are in the target package.
4. On the **File** menu, click **Save As**, and save the changes as a Windows Installer Transform (.mst) file.
5. Open **Application Catalog** and reimport this package with its transform file into your **Application Catalog**, and then use the **Conflict Wizard** to check it against ACE24 again.

ACE30: Different Components that Install the Same Key File

Edition • *This test is included in the AdminStudio Professional and Enterprise Editions.*

The ACE30 is a check for key path conflicts across components. It identifies components that have different ComponentID values but that install the same key file to the same directory.

Test Group/Test Category

Application Conflicts/Package Data Conflicts/Components

Severity

Warning

Message

```
The [SOURCE_COMPONENT_NAME] component in [SOURCE_PRODUCT_NAME] is installing the
    [SOURCE_PATH_NAME]\[SOURCE_KEYPATH_FILE] file which is also being installed by [TARGET_PRODUCT_NAME]
    with a different ComponentId.
```

Background

ACE30 checks to determine if different components are installing identically named key files to the same directory.

The ACE executes the following query to check to see if the same key file is being installed with different ComponentId values:

```
Source.ComponentId <> Target.ComponentId AND Source.csFullPath = Target.csFullPath AND Source.KeyPath =
    Target.KeyPath
```

If the source and target products are installed in a particular order, one of the products may not work as expected. For example, if the files have the same name and location but one of them is an earlier version, if the product that contains the earlier version is installed after the product that installs the latest version, the product that requires the latest version of the file may not work. It is also possible that uninstalling one of the products may make the other product stop functioning.

If you encounter an ACE30 failure, test the installation and uninstallation of the source and target products, and ensure that both behave as expected in all expected scenarios.

Microsoft App-V Conflict Tests

The following are the Microsoft App-V conflict tests:

- ACE200: Shortcut Location Conflicts
- ACE204: App-V Package ID Conflicts
- ACE205: Package Name Conflicts
- ACE206: File Extension and ProgID Conflicts
- ACE207: App-V Conflicts in Root Folder Names
- ACE215: App-V Shortcut Name and Version Conflicts

ACE200: Shortcut Location Conflicts

ACE200 checks whether two or more packages contain a shortcut with the same display name and location. It can be run against an App-V source package and against either an App-V or MSI target package. ACE200 identifies the shortcut name and location for the source package, then compares it to the target package's name and location.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Shortcut [SHORTCUT_NAME] in package [PACKAGE_NAME] has a target that conflicts with shortcut
    [SHORTCUT_NAME] in package [PACKAGE_NAME]
```

Background

If two or more packages contain a shortcut with the same display name and location, an error is generated.

Resolution

To resolve this issue, use the Virtual Package Editor.

Manual Fix

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. Open the **Shortcuts** view, and do one of the following:
   - Select the shortcut, and then modify the value in the **Display Name** setting or the **Location** setting.
   - Remove the shortcut from the App-V package.

ACE204: App-V Package ID Conflicts

ACE204 checks whether two or more packages have the same package GUID. If two packages have the same package GUID, they cannot be deployed simultaneously as separate packages. ACE204 can be run against an App-V source package and against only an App-V target package.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Package [PACKAGE_NAME] has a Package GUID that conflicts with package [PACKAGE_NAME]
```

Background

If two or more packages have the same package GUID, an error is generated.

Resolution

Manual Fix

If you are creating an upgrade package that can update earlier versions of the virtual package, the package GUID should stay the same.

If you are creating a new package that can be deployed simultaneously as another package, the package GUID in one of the packages must be changed. To change the package GUID, open the App-V package in the Virtual Package Editor and save the package as a new package.

ACE205: Package Name Conflicts

ACE205 checks whether two or more packages have the same name. This is not advisable from a best practice perspective, and it may cause some issues if you try to simultaneously deploy the App-V packages. ACE205 can be run against an App-V source package and against either an App-V or MSI target package.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Package [PACKAGE_NAME] has a name conflict with package [PACKAGE_NAME].
```

Background

If two or more packages have the same name, an error is generated.

Resolution

To resolve this ACE in an App-V package, use the Virtual Package Editor.

Manual Fix

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Package Information**, click **General Information**.
3. In the **Name** setting, replace the duplicate name with a unique name.

ACE206: File Extension and ProgID Conflicts

ACE206 checks whether two or more packages have support for the same file extension or ProgId. A file extension can be registered with only one application at a time. ACE206 can be run against an App-V source package and against either an App-V or MSI target package.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Package [PACKAGE_NAME] has a conflicting file extension [FILE_EXTENSION] or progid [PROGID] with package
    [PACKAGE_NAME].
```

Background

If two or more packages have support for the same file extension or ProgID, an error is generated.

Resolution

Manual Fix

To resolve ACE206, you may need to decide which package should contain the file extension association and which should not. Then you can use the Virtual Package Editor to remove the appropriate file extension.

ACE207: App-V Conflicts in Root Folder Names

ACE207 checks whether two or more packages have the same long or short name for the root folder. These names must be unique because two packages with the same root folder name cannot be deployed simultaneously. ACE207 can be run against an App-V source package and against only an App-V target package.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Package [PACKAGE_NAME] has a conflicting root Directory [DIRECTORY_NAME] with package [PACKAGE_NAME].
```

Background

If two or more packages have the same long or short name for the root folder, an error is generated.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Package Information**, click **General Information**.
3. In the **Root Folder Name** setting, replace the duplicate folder name with a unique folder name.

Note • *Instances of the old package's root folder name may still exist in location-related configuration data, such as in registry entries, .ini files, or XML files in the App-V package. The root folder name is not updated in those areas automatically if you change the root folder name in the General Information view.*

Therefore, if you know that the old package contains configuration data, you may need to identify where it is. Then you can use the Virtual Package Editor to update the root folder name as necessary. For example, you may want to use the Virtual Package Editor to extract a configuration file from the package. Next, you can update the root folder name in the file. In the Virtual Package Editor, you would then delete the old file from the App-V package, and add the updated file.

ACE215: App-V Shortcut Name and Version Conflicts

ACE215 indicates that an App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package. The combination of the name and version should be unique for shortcuts in different packages, since only one application is published and available at any given time. ACE215 can be run against an App-V source package and against only an App-V target package.

Test Group/Test Category

Application Conflicts/Microsoft App-V Conflict Tests

Severity

Error

Message

```
Shortcut [SHORTCUT_NAME] in package [PACKAGE_NAME] has a name and version that conflicts with shortcut
    [SHORTCUT_NAME] in package [PACKAGE_NAME].
```

Background

If an App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package, an error is generated.

Resolution

Manual Fix

Task ***To resolve this issue:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. Do one of the following:
   - Edit the shortcut: Select the target that contains the shortcut, and then modify the value in the **Name** setting or the **Target Version** setting.
   - Remove the shortcut from the App-V package: In the **Targets** explorer, right-click the shortcut, and then click **Remove**.

Remote Application Publishing Compatibility Tests

Edition • *The Remote Desktop Services tests are included in the AdminStudio Professional and Enterprise Editions.*

The **Remote Application Publishing Compatibility** category of tests includes tests that check Windows Installer packages for compatibility to be run via Windows Remote Desktop. These tests are grouped in the Remote Desktop Services Tests category.

The following subcategories of Remote Application Publishing Compatibility tests are available:

- Azure Application Services Tests
- Remote Desktop Services Tests

Azure Application Services Tests

Edition • *The Remote Desktop Services tests are included in the AdminStudio Professional and Enterprise Editions.*

The following Azure Application Services tests are described in this section:

- MAS0001: Port Bindings
- MAS0002: Authentication
- MAS0003: Global Assembly Cache (GAC)
- MAS0004: IIS5 Compatibility Mode
- MAS0005: Application Pools
- MAS0006: COM and COM+ Components
- MAS0007: ISAPI Filters
- MAS0008: Migration of Other Components Like SSL, FTP

MAS0001: Port Bindings

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

Azure Websites only support Port 80 for HTTP and Port 443 for HTTPS traffic. Different port configurations will be ignored and traffic will be routed to 80 or 443.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Warning

Message

```
Identified the port binding information other than 80 or 443 which will be ignored and traffic will be
    routed to port 80 or 443.
```

Resolution

Configure the Website to use port 80 for HTTP and 443 for HTTPS.

MAS0002: Authentication

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

Azure Websites support Anonymous Authentication by default and Forms Authentication where specified by an application. Windows Authentication can be used by integrating with Azure Active Directory and ADFS only. All other forms of authentication, e.g., Basic Authentication, are not currently supported.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Error

Message

```
Identified the Authentications other than Anonymous, Form and Windows [ASSEMBLY_NAME]. All other forms
    of authentication are not currently supported.
```

Resolution

Presence of Authentication other than Anonymous, Form and Windows need to be avoided.

MAS0003: Global Assembly Cache (GAC)

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

The GAC is not supported in Azure websites. If your application references assemblies which you usually deploy to the GAC, you will need to deploy to the application bin folder on Azure websites.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Warning

Message

```
Identified the presence of GAC assembly in this path [ASSEMBLY_NAME]. These need to be deployed to the
    Application bin folder on Azure websites.
```

Resolution

Deploy the GAC dlls along with application bin folder on Azure Websites.

MAS0004: IIS5 Compatibility Mode

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

This is not supported on Azure Websites.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Error

Message

```
IIS 5 compatibility is not supported for Azure websites. Identified IIS version [MAJOR_VERSION,
    MINOR_VERSION] - [ASSEMBLY_NAME].
```

Resolution

This is not supported on Azure Websites.

MAS0005: Application Pools

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

In Azure Websites, each site and its child applications run in the same application pool.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Warning

Message

```
More than one application pool has been identified for the applications under a single website
    [ASSEMBLY_NAME]. Try to consolidate them to one or create a separate website for each application.
```

Resolution

If your site has multiple child applications utilizing multiple application pools, consolidate them to a single application pool with common settings or migrate each application to a separate website.

MAS0006: COM and COM+ Components

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

Azure Websites do not allow the registration of COM Components on the platform.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Warning

Message

```
Presence of COM and COM+ component has been identified [ASSEMBLY_NAME]. this must be rewritten into the
    managed code and deployed with website or application.
```

Resolution

If your websites or applications make use of any COM Components, you must rewrite them in managed code and deploy them with the website or application.

MAS0007: ISAPI Filters

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

Azure Websites can support the use of ISAPI Filters, however, the DLL(s) need to be deployed with your site and registered via the `web.config`.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Warning

Message

```
Identified the presence of ISAPI filter [ASSEMBLY_NAME]. Deploy the dll with the site, register the dll
    via web.config and place apphostconfig.xdt in the application site root folder with content. For
    more info, refer to http://azure.microsoft.com/en-us/documentation/articles/web-sites-migration-
    from-iis-server/.
```

Resolution

ISAPI filters need to be deployed with site and registered via the `web.config`.

MAS0008: Migration of Other Components Like SSL, FTP

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

Other components like SharePoint, front page server extensions (FPSE), FTP, and SSL certificates will not be migrated.

Test Group/Test Category

Remote Application Publishing Compatibility/Azure Application Services Tests

Severity

Error

Message

```
Identified the presence of other migration components like FTP, SSL [ASSEMBLY_NAME]. This won't be
    migrated. Manually reconfigure after the migration is complete.
```

Resolution

Manually reconfigure the other components like SSL, FTP after the migration is complete.

Remote Desktop Services Tests

Edition • *The Remote Desktop Services tests are included in the AdminStudio Professional and Enterprise Editions.*

The following Remote Desktop Services tests are described in this section:

- WTS01: Per-User ALLUSERS Property Value for Remote Desktop Services

- WTS02: Registry Entries in Per-User Locations
- WTS03: Files in Per-User Locations
- WTS04: ODBC Data Source Entries in Per-User Locations
- WTS05: Per-User Environment Variables
- WTS06: Executable Files with Disabled TSAWARE Flags
- WTS07: TerminalServer or RemoveAdminTS Conditions
- WTS08: 16-Bit Binary Files
- WTS09: Administrator Manifest for Binary Files

WTS01: Per-User ALLUSERS Property Value for Remote Desktop Services

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS01 checks whether the ALLUSERS property is defined with a per-user value. In general, Windows Remote Desktop Services require installations to be installed for all users.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The product [PRODUCT_NAME] ('[VERSION]') is currently configured to be installed as per-user. This value
    affects deployment of this package in a terminal server environment.
```

Background

WTS01 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require installations to be installed for all users, rather than on a per-user basis. If the package is configured to be installed for the current user, WTS01 fails.

Resolution

Automatic Fix (WTSFIX)

WTSFIX01 automatically sets the value of the ALLUSERS property to 1.

WTS02: Registry Entries in Per-User Locations

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS02 checks for any registry entries that are installed to per-user specific locations. In general, Windows Remote Desktop Services require registry entries to be installed for all users.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The component [COMPONENT_NAME] in package [PACKAGE_NAME]('[VERSION]') contains a per-user registry key
    [KEY_NAME] with KeyPath [KEYPATH_VALUE]. This value affects deployment of this package in a terminal
    server environment.
```

Background

WTS02 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require registry entries to be installed for all users, rather than on a per-user basis. If the package contains registry entries that are configured to be installed to per-user locations, WTS02 fails.

Resolution

Automatic Fix (WTSFIX)

WTSFIX02 automatically clears the key path entry for the identified resource to ensure that Windows Installer repair mode is not invoked.

WTS03: Files in Per-User Locations

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS03 checks whether any of the files in the package are configured to be installed to per-user locations. In general, Windows Remote Desktop Services require files to be installed for all users.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The component '[COMPONENT_NAME]' in package '[PACKAGE_NAME]'('[VERSION]') has a per-user destination
    '[PATH_NAME]'. This value affects deployment of this package in a terminal server environment.
```

Background

WTS03 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require files to be installed for all users, rather than on a per-user basis. If the package contains one or more files that are configured to be installed to per-user locations, WTS03 fails.

Resolution

Automatic Fix (WTSFIX)

WTSFIX03 automatically clears the key path entry for the identified resource to ensure that Windows Installer repair mode is not invoked.

WTS04: ODBC Data Source Entries in Per-User Locations

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS04 checks for any ODBC data source entries in the package are configured to be installed to per-user locations.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The component '[COMPONENT_NAME]' in package '[PACKAGE_NAME]'('[VERSION]') has a per-user ODBC Data
    Source '[NAME]'('[VALUE]'). This value affects deployment of this package in a terminal server
    environment.
```

Background

WTS04 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require ODBC data source entries to be installed for all users, rather than on a per-user basis. If the package contains one or more ODBC data source entries that are configured to be installed to per-user locations, WTS04 fails.

Resolution

Automatic Fix (WTSFIX)

WTSFIX04 automatically clears the key path entry for the identified resource to ensure that Windows Installer repair mode is not invoked.

WTS05: Per-User Environment Variables

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS05 checks whether any environment variables in the package are configured to be installed for the current user. In general, Windows Remote Desktop Services require environment settings to be installed for all users.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Warning

Message

```
The component '[COMPONENT]' in package '[PACKAGE_NAME]' has a per-user Environment Setting
    '[NAME]'('[VALUE]'). This value affects deployment of this package in a terminal server environment.
```

Background

WTS05 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require environment variables to be installed for all users, rather than on a per-user basis. If the package contains one or more environment variables that are configured to be installed per user, WTS05 fails.

Resolution

Manual Fix

Duplicate the per-user Environment table entries and conditionalize the component that contains these new Environment table entries to be installed only if the ALLUSERS property is set to 1 for all users.

WTS06: Executable Files with Disabled TSAWARE Flags

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS06 checks a package to make sure that all of the executable files that it contains have the TSAWARE flag enabled.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The binary '[EXECUTABLE_FILE_NAME]' in package '[PACKAGE_NAME]'('PACKAGE_NAME') does not have the
    TSAWARE flag set. This affects the deployment of this package in a terminal server environment.
```

Background

WTS06 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Windows Remote Desktop Services generally require executable files to be marked as Terminal Server aware. If the package contains one or more executable files that have the TSAWARE flag disabled, WTS06 fails.

Resolution

Manual Fix

To resolve this issue, remove from the package the executable file that does not have the TSAWARE flag enabled.

WTS07: TerminalServer or RemoveAdminTS Conditions

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS07 checks the LaunchCondition table of a package and the conditions for custom actions for use of the `TerminalServer` property or the `RemoteAdminTS` property.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Best Practices/Remote Desktop Services Tests

Severity

Warning or error

Messages

- `The InstallExecuteSequence/InstallUISequence table has a Type 19 CustomAction [ACTION] with the condition [CONDITION]. This may affect the deployment of this package in a terminal server environment.`
- `The LaunchCondition table has the condition [CONDITION]. This may affect the deployment of this package in a terminal server environment.`

Background

WTS07 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. If the package contains one or more conditions that use the `TerminalServer` property or the `RemoteAdminTS` property, WTS07 fails.

Resolution

This issue cannot be resolved unless it is possible to delete the `TerminalServer` and `RemoteAdminTS` properties from the conditions in the package. To determine the ramifications of removing the condition, contact the software vendor.

WTS08: 16-Bit Binary Files

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS08 checks a package for 16-bit applications. Windows Server 2008 R2 and later are 64-bit systems, and they cannot run 16-bit applications.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

`The binary '[FILE_NAME]' in package '[PACKAGE_NAME]'('[PACKAGE_NAME]') is 16-bit. This affects the deployment of this package in a terminal server environment.`

Background

WTS08 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. If the package contains one or more 16-bit binary files (.exe, .dll, or .ocx), WTS08 fails.

Resolution

Manual Fix

To resolve this issue, remove the specified 16-bit binary file from the Windows Installer package.

WTS09: Administrator Manifest for Binary Files

Edition • *This test is included in AdminStudio Professional and Enterprise Editions.*

WTS09 checks a package for binary files (.exe, .dll, or .ocx) that have a manifest in which Administrator is defined as the required execution level.

Note • *If your package is not targeting the Windows Remote Desktop Services environment, you do not need to run this ACE.*

Test Group/Test Category

Remote Application Publishing Compatibility/Remote Desktop Services Tests

Severity

Error

Message

```
The binary '[FILE_NAME]' in package '[PACKAGE_NAME]'('[PACKAGE_NAME]') is manifested to run as
    Administrator. This affects the deployment of this package in a terminal server environment.
```

Background

WTS09 examines a Windows Installer package for compatibility with Windows Remote Desktop Services. Typically, users of Windows Remote Desktop Services do not have elevated privileges. Therefore, a file that has a manifest that defines the required execution level as Administrator is usually not a good candidate to run in a per-user environment.

If the manifest of a binary file (.exe, .dll, or .ocx) indicates that the Administrator execution level is required, WTS09 fails.

Resolution

Manual Fix

The only way to resolve this error is to remove the specified file from the Windows Installer package.

Analyze Tests Reference

Edition • *Analyze is included in the AdminStudio Professional and Enterprise Editions.*

The Analyze Tests Reference section includes additional reference information in the following categories:

- Analyze Resolutions
- Creating Your Own Custom ACE Tests
- Viewing ACE Metrics
- Location of ACE Files

Analyze Resolutions

Edition • *Analyze is included in the AdminStudio Professional and Enterprise Editions.*

- *CARDs are included in AdminStudio Professional and Enterprise Editions.*

Application Catalog offers different types of methods for resolving issues that are identified by Analyze tests:

- Resolutions for Operating System Compatibility Tests
- Conflict Application Resolution Definitions (CARDs)

Resolutions for Operating System Compatibility Tests

Edition • *Analyze is included in the AdminStudio Professional and Enterprise Editions.*

- *The operating system tests are included in AdminStudio with Application Compatibility.*
- *The browser tests are included in AdminStudio Enterprise Application Compatibility.*

The Operating System Compatibility tests that are available in Analyze offer the following types of fixes for the issues that the tests identify:

- **Basic auto fix**—This type of resolution is relatively safe. It results in minimal changes to an MSI package via a Windows Installer transform. It does not change the target system's security or a system policy.
- **Advanced auto fix**—This type of resolution may result in a loss of functionality, and it may not resolve all types of issues. This type of fix may change the target system's security or a system policy. One example of an advanced auto fix is the removal of a registry key that is protected by Windows Resource Protection.
- **Manual fix**—This type of resolution describes a procedure or task that you can perform to address an issue.

Some types of fixes are not applicable to some of the Operating System Compatibility tests.

Conflict Application Resolution Definitions (CARDs)

Edition • *Analyze is included in the AdminStudio Professional and Enterprise Editions.*

CARDs are included in AdminStudio Professional and Enterprise Editions.

Conflict Application Resolution Definitions (CARDs) are used to fix issues that Application Conflict Evaluators (ACEs) identify when conflict testing is performed.

The following table lists each individual CARD and its associated ACE. These CARDs are used to resolve conflicts between installation packages that were identified by the CARD's corresponding ACE.

Table 17-8 • CARD Index

CARD	Brief Description of ACE	Action Taken by CARD
CARD02 for ACE02	Checks whether components in different packages that have matching ComponentIds also have identical destination paths.	The destination path of the component in the Source package is automatically set to match that of the component in the Target package. To learn more, see ACE02: Identical Components with Different Destinations.
CARD04 for ACE04	Checks whether components with no files and no key paths have an associated entry in the CreateFolder.	A CreateFolder entry is created for the component. To learn more, see ACE04: Components Without Files or Key Paths.
CARD05 for ACE05	Checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package.	Modifies the component so that only one EXE or DLL exists, and it adds new components for remaining EXE, DLL, OCX, HLP, CHM, TLB, SYS, and DRV files. To learn more, see ACE05: More Than One Executable File Per Component.
CARD06 for ACE06	Checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file.	The executable module is automatically made the key file. To learn more, see ACE06: Executable File Not Marked as Key File of Component.
CARD07 for ACE07	Checks for the existence of the same file in components with different ComponentIds.	The Source package ComponentId is changed to match the Target package Component Id. To learn more, see ACE07: Same File in Different Components.
CARD15 for ACE15	Checks for the existence of identical ODBC entries in components with different ComponentIds.	Application Catalog changes the Source ComponentId to match that of the Target ComponentId. To learn more, see ACE15: Duplicate ODBC Entries in Different Components.

Note • *Issues that are found by ACEs and that do not have associated CARDs must be resolved manually.*

Creating Your Own Custom ACE Tests

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

In addition to using the built-in ACEs that are shipped with Application Catalog, you can also create your own company-specific tests for use when detecting conflicts. For example, your organization may want to identify (and change) any VBScript custom actions that have a hard-coded drive letter, any applications that create desktop or uninstall shortcuts, any applications that have Startup registry entries, or any applications that place files in the system directory. You can create custom tests to identify these (and many more company-specific situations) using Application Catalog.

Note • *Use the Rules Wizard to create user-defined ACEs. To open the* ***Rules Wizard****, open the* ***ACE Tests*** *tab of the Application Catalog* ***Options*** *dialog box, click the* ***View Rules*** *button to open the* ***Rules Viewer*** *dialog box, and then click the* ***New*** *button. The* ***Rules Wizard*** *opens. See Rules Wizard for more information.*

Types of User-Defined ACEs

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

In addition to the ACEs included with Application Catalog, you can also create your own user-defined ACEs to use when detecting conflicts. You can create three types of ACEs:

Custom - Source Only Packages ACEs

Custom - Source Only Packages ACEs allow you to quickly test any column or any value of a table to support your business logic. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a Source Only Packages ACE, you must define an SQL “Where” clause.

For an example of this type of user-defined ACE, see Creating a Custom/Source Only Packages ACE.

Note • *Application Catalog supports external package conflict checking for Custom - Source Only Packages ACEs. The Source package can be selected from the Application Catalog Database or from an external MSI package.*

Custom - Source and Target Packages ACEs

Custom - Source and Target Packages ACEs allow you to compare columns or values of Source package tables (new packages that you want to install onto a user’s system) to columns or values of Target package tables (packages already installed on a user’s system).

For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the System directory of the Target system.

To define a Source and Target Packages ACE, you must define an SQL “Where” clause, and specify a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages.

For an example of this type of user-defined ACE, see Creating a Custom/Source and Target Packages ACE.

Note • *Application Catalog does not support external package conflict checking for Custom - Source and Target Packages ACEs. Both the Source and Target Packages must be selected from the Application Catalog Database.*

DLL - User Provided DLL Based ACEs

DLL - User Provided DLL Based ACEs allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and various programming languages to construct a Windows DLL. With DLL-Based ACEs, you can use a Conflict Application Resolution Definition (CARD) to fix the conflict.

For an example of this type of user-defined ACE, see Creating a User Provided DLL-Based ACE.

Where You Create User-Defined ACEs

When creating ACEs, you need to provide basic information for display in the **Rules Viewer** dialog box, on the **ACE Tests** tab of the **Options** dialog box, and in the **Output** window. You must associate a table with the ACE. You also must categorize the ACE (by either using existing ACE categories or creating your own).

Note • *Use the Rules Wizard to create user-defined ACEs. To open the **Rules Wizard**, open the **ACE Tests** tab of the Application Catalog **Options** dialog box, click the **View Rules** button to open the **Rules Viewer** dialog box, and then click the **New** button. The **Rules Wizard** opens. See Rules Wizard for more information.*

Creating User-Defined ACEs

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

The section describes how to create the three types of user-defined ACEs:

- Creating a Custom/Source Only Packages ACE
- Creating a Custom/Source and Target Packages ACE
- Creating a User Provided DLL-Based ACE

Note • *You can create user-defined ACEs for both Windows Installer and App-V packages.*

Creating a Custom/Source Only Packages ACE

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

You can create a user-defined ACE to apply to Source Only Packages. One common task you may want to create a Source Only Packages ACE to handle is to identify packages which create a desktop icon.

Application Catalog supports external package conflict checking for Custom - Source Only Packages ACEs. The Source package can be selected from the Application Catalog database or from an external package.

Task ***To create a Source Only Packages ACE that identifies desktop icon creation:***

1. Launch Application Catalog.
2. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.
3. On the **ACE Tests** tab, click the **View Rules** button. The **Rules Viewer** dialog box opens.
4. Click the **New** button. The **Rules Wizard** opens.
5. Complete the wizard panels to create your ACE:
   a. On the **General Information** panel**,** enter the following values:

Option	Value
Name	`MYACE`
Associated Table	`csmsiShortcut`
Package Type	`MSI`
Brief Description	`MYACE - Find desktop icons`
Description	`Locates package that create desktop icons.`
Information URL	`http://www.yourcompany.com/MYACE.htm`

   b. On the **Additional Information** panel**,** enter the following values:

Option	Value
Category	`Shortcuts`
Rule Type	`Custom - Source Only Packages`

   c. On the **Custom Options** panel**,** enter the following values:

Option	Value
Error String	`Desktop icons [Name] are not allowed.`
Display String	`A desktop icon called [Name] is created.`
Severity	`Warning`
Report 'No' Results	`Deselected`

In the example above, `[Name]` is a token. Tokens allow you to insert values at run-time from the internal Application Catalog Database or from an external MSI package into the Error or Display string. To use token replacement in a string, click the arrow to the right of the Error String and Display String text boxes and select a column name from the list. The column name is then inserted into the string in the following format: `[ColumnName]`.

The Token list is provided for your convenience; if you prefer, you can type the tokens directly in the text boxes. You could also use the [ProductName] pseudo-token to insert the name of the package in a message, even though ProductName is not a table column name.

Note • *For more information, see Token Grammar*

d. On the **Where Clause** panel, in the **Where Clause** panel, click **Build Expression**. The **Expression Builder** dialog box opens.

e. In the **Expression Builder**, enter the following values:

Option	Value
Table Columns	[Directory_]
Comparison Operator	= (Equal To)
String Constant	DesktopFolder

f. Click **OK** to close the **Expression Builder** and return to the **Where Clause** panel. The expression that you just built is now displayed in the Where Clause text box:

```
[Directory_] = 'DesktopFolder'
```

When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the Where Clause text box on the Where Clause panel or on the Where Clause tab of the ACE Rule Properties Dialog Box.

g. Click the **Test** button to validate the expression.

h. On the **Summary** panel, review the summary of your new ACE and click the **Finish** button.

6. Click the **Close** button. The **Rules Viewer** dialog box closes.

This new user-created ACE is now available for use in subsequent testing.

Creating a Custom/Source and Target Packages ACE

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

You can create a custom ACE to apply to Source and Target Packages. For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the System directory of the Target system.

Caution • *Application Catalog does not support external package conflict checking for Custom - Source and Target Packages ACEs. Both the Source and Target Packages must be selected from the Application Catalog Database. If you attempt to run a conflict check on an external MSI package using a Source and Target Packages ACE, that custom ACE will not be executed.*

Task ***To create a Source and Target Packages ACE that identifies .ini file conflicts:***

1. Launch Application Catalog.
2. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.
3. On the **ACE Tests** tab, click the **View Rules** button. The **Rules Viewer** dialog box opens.
4. lick the **New** button. The **Rules Wizard** opens.
5. Complete the wizard panels to create your ACE:

   a. On the **General Information** panel**,** enter the following values:

Option	Value
Name	`INICheck`
Associated Table	`csmsiIniFile`
Package Type	`MSI`
Brief Description	`INICheck - Identifies conflicts found in an .ini file.`
Description	`Determines if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the Target system's System directory.`
Information URL	`http://www.yourcompany.com/INICheck.htm`

   b. On the **Additional Information** panel, enter the following values:

Option	Value
Category	`INI Files`
Rule Type	`Custom - Source and Target Package`

   c. On the **Custom Options** panel, enter the following values:

Option	Value
Error String	`The INI file called [Source.FileName] in the directory [Source.csFullPath] writes to the [Source.Section] section which is also written by the target package, [Target.ProductName].`
Display String	`The INI file called [Source.FileName] in the directory [Source.csFullPath] writes to the [Source.Section] section.`
Severity	`Warning`
Report 'No' Results	Deselected

   In the example above, `[Source.Filename]`, `[Source.csFullPath]` and `[Source.Section]` are tokens. Tokens allow you to insert values at runtime from the internal Application Catalog Database into the Error or Display string. To use token replacement in a string, click the arrow to the right of the Error String or Display String text

boxes and select a column name from the list. The column name is then inserted into the string in the format of `[Source.ColumnName]` or `[Target.ColumnName]`, with the prefix identifying whether the column is in the Source or Target package.

Note • *If no prefix is used, Application Catalog assumes the "*`Source.`*" prefix.*

You can also use the `[Target.ProductName]` and `[Source.ProductName]` pseudo-tokens to insert the name of the Source or Target package in a message, even though ProductName is not a table column name.

Note • *The Token list is provided for your convenience; if you prefer, you can type the variables directly in the text boxes. For more information, see Token Grammar.*

d. On the **Where Clause** panel, in the **Where Clause** panel, click **Build Expression**. The **Expression Builder** dialog box opens.

e. In the **Expression Builder**, enter the following values:

Option	Value
Table Columns	`[Source].[csFullPath]`
Comparison Operator	`= (Equal To)`
String Constant	`SystemFolder`

f. Click **OK** to close the **Expression Builder** and return to the **Where Clause** panel. The expression that you just built is now displayed in the **Where Clause** text box:

```
[Source].[csFullPath] = 'SystemFolder'
```

Note • *When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the* ***Where Clause*** *text box on the* ***Where Clause*** *panel or on the* ***Where Clause*** *tab of the ACE Rule Properties Dialog Box.*

g. Click **Build Expression** again to open the **Expression Builder**.

h. In the **Expression Builder**, enter the following values:

Option	Value
Table Columns	`[Source].[Section]`
Comparison Operator	`= (Equal To)`
String Constant	`[Target].[Section]`
Expression Operator	`AND`

Note • *When using the Expression Builder dialog box to create a Source and Target Packages custom ACE to compare the value of a column in the source table to the value of a column in the target table, you can select the first table column name from the Table Columns list. However, you have to manually enter the second table column name in the Constant text box. When doing so, enter the table column name using the same syntax that is used in the Table Columns list: [Source].[ColumnName] or [Target].[ColumnName].*

i. Click **OK** to close the **Expression Builder** and return to the **Where Clause** panel. The expression that you just built is now displayed in the **Where Clause** text box, added to the end of the first expression you built:

```
[Source].[csFullPath] = 'SystemFolder' AND
[Source].[Section] = '[Target].[Section]'
```

j. In the **Join Column** list, select **csFullPath**.

The **Join Column** is a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages.

k. Click the **Test** button to validate the expression. A message appears stating that the query executed properly.

l. On the **Summary** panel, review the summary of your new ACE and click the **Finish** button.

6. Click the **Close** button. The **Rules Viewer** dialog box closes.

This new Custom ACE is now available for use in subsequent testing.

Creating a User Provided DLL-Based ACE

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

To demonstrate how to create a DLL-Based ACE, an ACE SDK was installed with AdminStudio, in the following directory:

`[AdminStudioInstallDirectory]\Common\ACESDK`

Included in this directory is a Visual Studio 2008 DLL-based project, which is a fully formed example ACE. This example includes utility functions that allow you to integrate your own ACE execution within our conflict persistence model.

Specifying the Visual Studio 2008 Type Library File Path

Application Catalog provides a data structure to each ACE constructed using the ACE SDK. This data structure includes an ADO Connection interface, which is the means by which you can execute queries against the Application Catalog.

In order to build the ACE SDK, you will need to make sure that Visual Studio can locate the needed files for ADO.

The *first time* you create a DLL-based ACE, you need to open Visual Studio 2008 and specify the path for the type library file (`msado15.dll`) by performing the following steps:

Task ***To specify the Visual Studio 2008 type library file path:***

1. Launch Visual Studio 2008.
2. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
3. Under **Projects and Solutions > VC++ Directories**, set the **Platform** field to **Win32** and the **Show directories for** field to **Library files**.
4. In the directories list, specify the location of the needed files for an ADO Connection interface. The directory specified below is a likely common directory for storing these files:

   ```
   C:\Program Files\Common Files\System\ADO
   ```

5. Click **OK** to exit the **Options** dialog box.

Caution • *If you fail to specify the correct Library files path, you will encounter the following error when building an ACESDK project with Visual Studio 2008:*

```
Fatal error C1083: Cannot open type library file: 'msado15.dll'; No such file or directory
```

Creating a DLL-Based ACE

To learn how to create a DLL-Based ACE, use the ACE SDK files to perform the following steps:

Task ***To create a DLL-based ACE:***

1. Launch Windows Explorer and navigate to the following directory:

   ```
   [AdminStudioInstallDirectory]\Common\ACESDK
   ```

2. Copy this folder and its contents and store it in a convenient location.
3. Launch Visual Studio 2008 and open the `ACESDK.dsp` project file within that newly created folder.
4. Review the code and make any desired changes.
5. Still in Visual Studio 2008, build the `ACESDK.dsp` project to create a new .DLL file.
6. Launch Application Catalog.
7. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.
8. On the **ACE Tests** tab, click the **View Rules** button. The **Rules Viewer** dialog box opens.
9. Click the **View Rules** button. The **Rules Viewer** dialog box opens.
10. Click the **New** button. The **Rules Wizard** opens.
11. Complete the wizard panels to create your ACE:

a. On the **General Information** panel**,** enter the following values:

Option	Value
Name	`ACELanguage`
Associated Table	`csmsiProperty` (This is the Application Catalog table that is associated with this example ACE.)
Package Type	`MSI`
Brief Description	`ACELanguage - Check product language consistency.`
Description	Confirm that source product language is the same as all target product languages.
Information URL	`http://www.yourcompany.com/ACELanguage.htm`

b. On the **Additional Information** panel, enter the following values:

Option	Value
Category	Type in a new category name: `Product Language`
Rule Type	`DLL - User Provided DLL`

c. On the **DLL-Based ACEs** panel, enter the following values:

Option	Value
ACE/CARD DLL File	Click Browse and select the DLL that you built in Step 5 above.
ACE Function Name	`ExampleACE` (as designed in the sample)
CARD Function Name	`ExampleCARD` (as designed in the sample)

d. Click **Test** next to **ACE Function Name** or **CARD Function Name** to validate that the exported function does exist.

e. On the **Summary** panel, review the summary of your new ACE and click **Finish**.

12. Click the **Close** button. The **Rules Viewer** closes.

This new DLL-Based ACE is now available for use in subsequent testing.

Editing User-Defined ACEs

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

Task ***To edit a user-defined ACE:***

1. Launch Application Catalog.
2. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.

3. On the **ACE Tests** tab, click the **View Rules** button. The **Rules Viewer** dialog box opens.
4. Click the **View Rules** button. The **Rules Viewer** dialog box opens.
5. In the **Rules** box, select the user-defined ACE you want to modify and then click the **Edit** button. The **ACE Rule Properties** dialog box opens.
6. Edit the options as necessary.
7. Click **OK**.
8. Click the **Close** button. The **Rules Viewer** dialog box closes.

The modified ACE is available for use in subsequent testing.

Deleting User-Defined ACEs

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

Task ***To delete a user-defined ACE:***

1. Launch Application Catalog.
2. On the **Application Catalog** tab menu, click **Options**. The **Options** dialog box opens.
3. On the **ACE Tests** tab, click the **View Rules** button. The **Rules Viewer** dialog box opens.
4. Click the **View Rules** button. The **Rules Viewer** dialog box opens.
5. Select the ACE you want to remove and click the **Delete** button.
6. Confirm the deletion.

The user-defined ACE is removed from the available tests. If this ACE was the only one in a user-defined ACE category, the category will be removed when you close the Rules Viewer dialog box.

Viewing ACE Metrics

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

When ACEs are run, Application Catalog generates metrics and logs them in the `AceLog.txt` file, located in the following directory:

```
AdminStudio Shared\ConflictSolver\AceLog.txt
```

The following is an example of the beginning of an AceLog.txt file:

```
11/06/03 09:26:35 ACE/CARD Execution started

-----------------------------------------------
ACE03
-----------------------------------------------
```

```
SELECT DISTINCT csmsiComponent.Component,
     csmsiComponent.ComponentId,csmsiComponent.RowID
     FROM csmsiComponent WHERE csmsiComponent.PkgRowID_ = 5

Records returned =>(124)
Query Time =>(0.23 seconds)

SELECT csmsiFile.FileName FROM csmsiFile WHERE
     csmsiFile.Component_ = 'DeletedLinks.10' AND
     csmsiFile.PkgRowID_ = 5

ORDER BY csmsiFile.FileName

Records returned =>(0)

Query Time =>(0.07 seconds)

Time taken to execute (ACE03) : 1.06 seconds

-----------------------------------------------
ACE07
-----------------------------------------------
SELECT [fs].[RowID], [ft].[RowID], [ft].[PkgRowID_],
     [ft].[FileName], [cs].[Component], [ct].[Component],
     [cs].[ComponentId], [ct].[ComponentId], [fs].[Version],
     [fs].[FileSize], [fs].[Language] FROM  (([csmsiFile] AS [fs]
     INNER JOIN  [csmsiFile] AS [ft] ON ( [fs].[FileName] =
     [ft].[FileName] AND [fs].[Version] = [ft].[Version] AND
     [fs].[FileSize] = [ft].[FileSize] AND [fs].[Language] =
     [ft].[Language]) ) INNER JOIN [csmsiComponent] AS [cs] ON
     [fs].[PkgRowID_] = [cs].[PkgRowID_] AND [cs].[Component] =
     [fs].[Component_]) INNER JOIN [csmsiComponent] AS [ct] ON
     [ft].[PkgRowID_] = [ct].[PkgRowID_] AND [cs].[ComponentId] <>
     [ct].[ComponentId] AND [cs].[csFullPath] = [ct].[csFullPath] AND
     [ct].[Component] = [ft].[Component_] WHERE  [fs].[PkgRowID_] = 5
     AND  [ft].[PkgRowID_] IN (4)

Records returned =>(0)
Query Time =>(0.10 seconds)

Time taken to execute (ACE07) : 0.74 seconds

-----------------------------------------------
ACE08
-----------------------------------------------

SELECT [cs].[Component] AS [SrcComponent], [fs].[RowID] AS [SrcRowID],
     [ft].[RowID] AS [TargetRowID], [ft].[PkgRowID_] AS
     [TargetPkgRowID], [fs].[Version] AS [SrcVersion],
     [fs].[FileName], [cs].[csFullPath], [ct].[Component] AS
     [TargetComponent], [ft].[Version] AS [TargetVersion] FROM
     (([csmsiFile] AS [fs] INNER JOIN [csmsiFile] AS [ft] ON
     [fs].[FileName] = [ft].[FileName]) INNER JOIN [csmsiComponent]
     AS [cs] ON [cs].[PkgRowID_] = [fs].[PkgRowID_] AND
     [cs].[Component] = [fs].[Component_]) INNER JOIN
     [csmsiComponent] AS [ct] ON [cs].[ComponentId] =
```

```
      [ct].[ComponentId] AND  [ct].[PkgRowID_] =
      [ft].[PkgRowID_] AND [ct].[Component] = [ft].[Component_] AND
      [cs].[csFullPath] = [ct].[csFullPath] WHERE
      [fs].[PkgRowID_] = 5 AND [ft].[PkgRowID_] IN (4) AND
      ([fs].[Version] <> [ft].[Version] OR ([fs].[Version] IS
      NULL AND [ft].[Version] IS NOT NULL) OR ([ft].[Version]
      IS NULL AND [fs].[Version] IS NOT NULL)

Records returned =>(0)
Query Time =>(0.09 seconds)

Time taken to execute (ACE08) : 0.73 seconds
```

Location of ACE Files

Edition • *This functionality is included in the AdminStudio Professional and Enterprise Editions.*

ACE information is stored in three files that are installed with Application Catalog.

Table 17-9 • ACE File Names and Locations

Type	File Name	Installation Location
Standard ACEs	`isconflict.ace`	In the following subdirectory of the AdminStudio installation directory: `Common\Support`
Merge Module ACEs	`ismmconflict.ace`	In the following subdirectory of the AdminStudio installation directory: `Common\Support`
Custom ACE File	`CustomConflictFile.ace`	In the following subdirectory of the AdminStudio Shared directory: `\ConflictSolver` ***Note*** • *The location of the AdminStudio Shared directory is specified on the* ***AdminStudio Shared Location*** *panel of the AdminStudio installation wizard.*

Application Catalog requires that the Standard ACE and Merge Module ACE files remain in their installed location. However, you can change the location of the Custom ACE file by opening the **ACE Tests** tab of the Application Catalog **Options** dialog box, and editing the path in the **Custom ACE Rule File** field.

18

Analyzing the Impact of Installing Microsoft Operating System Security Patches

Edition • *Support for importing Microsoft OS Security Patch files and the Patch Impact Analysis Wizard are included with AdminStudio Enterprise Edition.*

You can import Microsoft OS patch information into the Application Catalog so that you can analyze the full impact of installing these patches on user machines. Based on the analysis results, you can determine the level of testing you need to perform before distributing a Microsoft OS patch throughout your enterprise.

You can import Microsoft security patch files into the Application Catalog using the Import Wizard. You can then analyze the impact of installing a security patch file using the Patch Impact Analysis wizard.

Information about the patch impact analysis is presented in the following sections:

Table 18-1 • Patch Impact Analysis Help Library

Section	Description
About Microsoft Operating System Patch Files	Explains what Microsoft operating system security patches are, and why you should include them in your package testing processes.
Importing Microsoft OS Security Patch Files	Explains how to download and import a Microsoft operating system patch into the Application Catalog. • Identifying and Downloading Microsoft Operating System Patch Files • Importing a Microsoft Operating System Security Patch Into the Application Catalog
Analyzing the Impact of Installing a Microsoft Operating System Patch	Explains how to use the Application Catalog Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system security patches and the packages and OS Snapshots in the Application Catalog.
Reference	Describes all of the panels in the Patch Impact Analysis Wizard and Patch Properties dialog box.

About Microsoft Operating System Patch Files

Each month, Microsoft releases patches to address security vulnerabilities that are discovered in Microsoft operating system software. Microsoft defines security vulnerabilities as:

"A flaw in a product that makes it infeasible – even when using the product properly—to prevent an attacker from usurping privileges on the user's system, regulating its operation, compromising data on it, or assuming ungranted trust."

Microsoft publishes a monthly Microsoft Security Bulletin Summary that lists the patches released that month. You can view these bulletins on the Microsoft Security TechCenter website:

https://technet.microsoft.com/security/bulletin

The Microsoft Security Bulletin Summary for July 2014 is shown in the following figure:

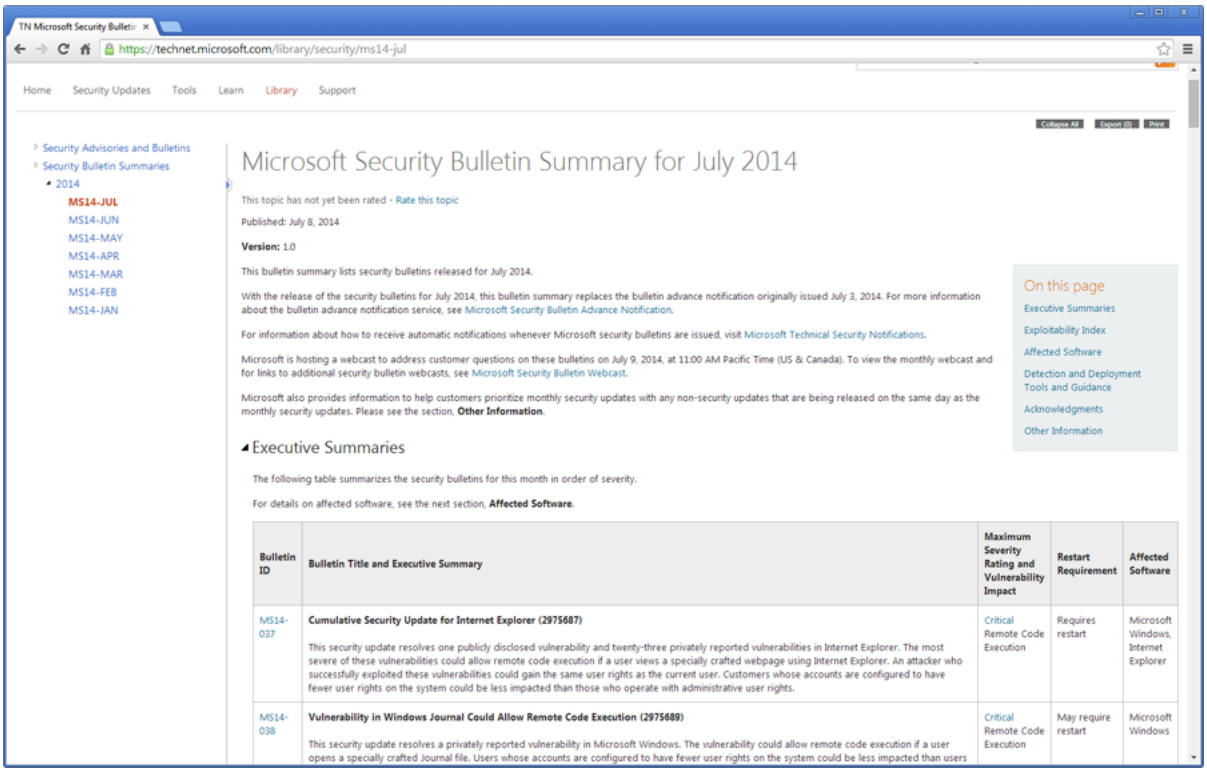


Figure 18-1: Microsoft Security Bulletin Summary for July 2014

The Security Bulletin Summary lists each patch released that month, grouped by status level, with a link to each patch's associated Security Bulletin, as shown in the following figure:

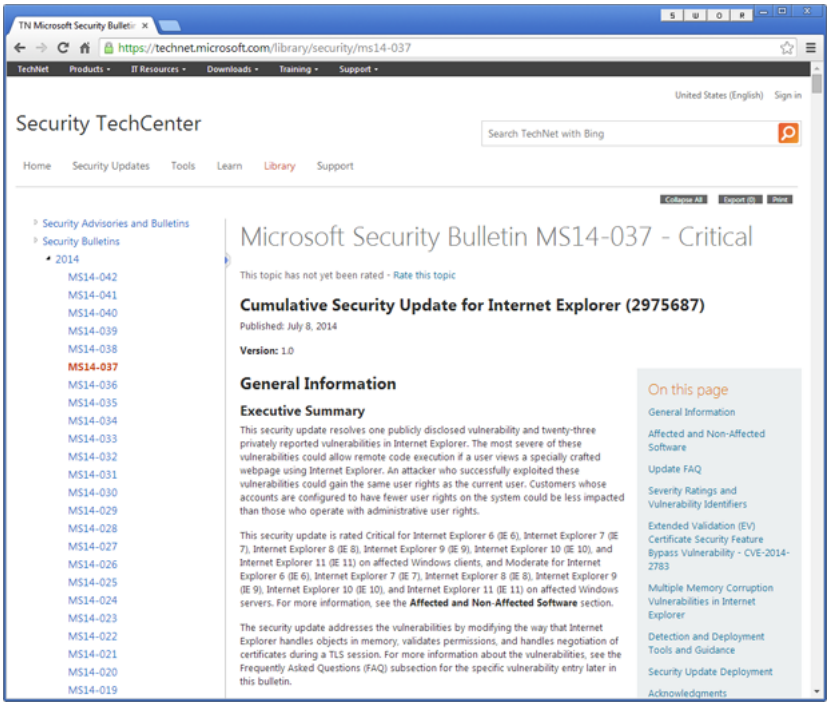


Figure 18-2: Microsoft Security Bulletin MS14-037

Security Bulletin Summaries, Security Bulletins, and patches can be accessed from the Microsoft Security TechCenter:

https://technet.microsoft.com/security/bulletin

Importing Microsoft OS Security Patch Files

You can use the Import Wizard to import Microsoft operating system patch files (`.msu`) into the AdminStudio Application Catalog one at a time.

Information about importing Microsoft operating system security patches is presented in the following sections:

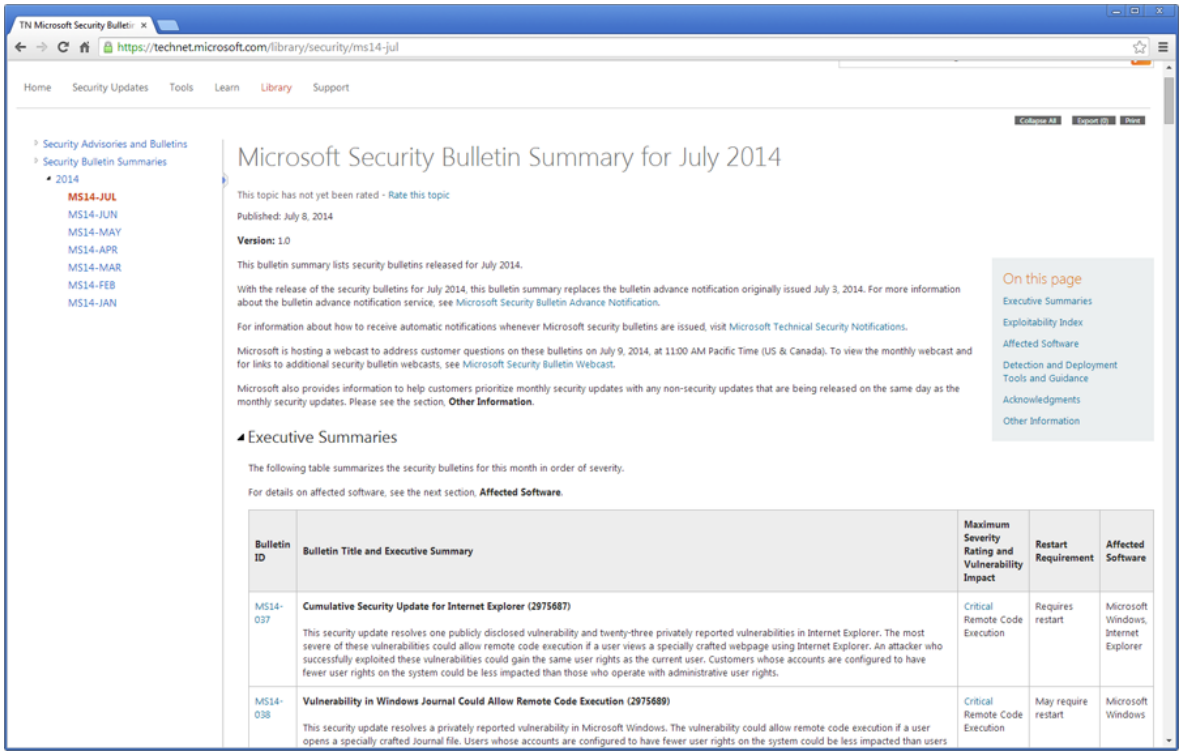
- Identifying and Downloading Microsoft Operating System Patch Files
- Importing a Microsoft Operating System Security Patch Into the Application Catalog

Identifying and Downloading Microsoft Operating System Patch Files

To identify and obtain the Microsoft OS security patch files that you want to import into the Application Catalog, perform the following steps.

Task ***To identify and download Microsoft OS patch files:***

1. Open the Microsoft Security Bulletin Summary that lists the patch that you want to import. The following figure is of the Microsoft Security Bulletin Summary for July 2014:



In this example, under **Executive Summaries**, 18 patch files are listed, all with a status of **Critical**.

2. Locate the bulletin that contains the OS security patch file that you want to download. The following is an example of Cumulative Security Update for Internet Explorer:

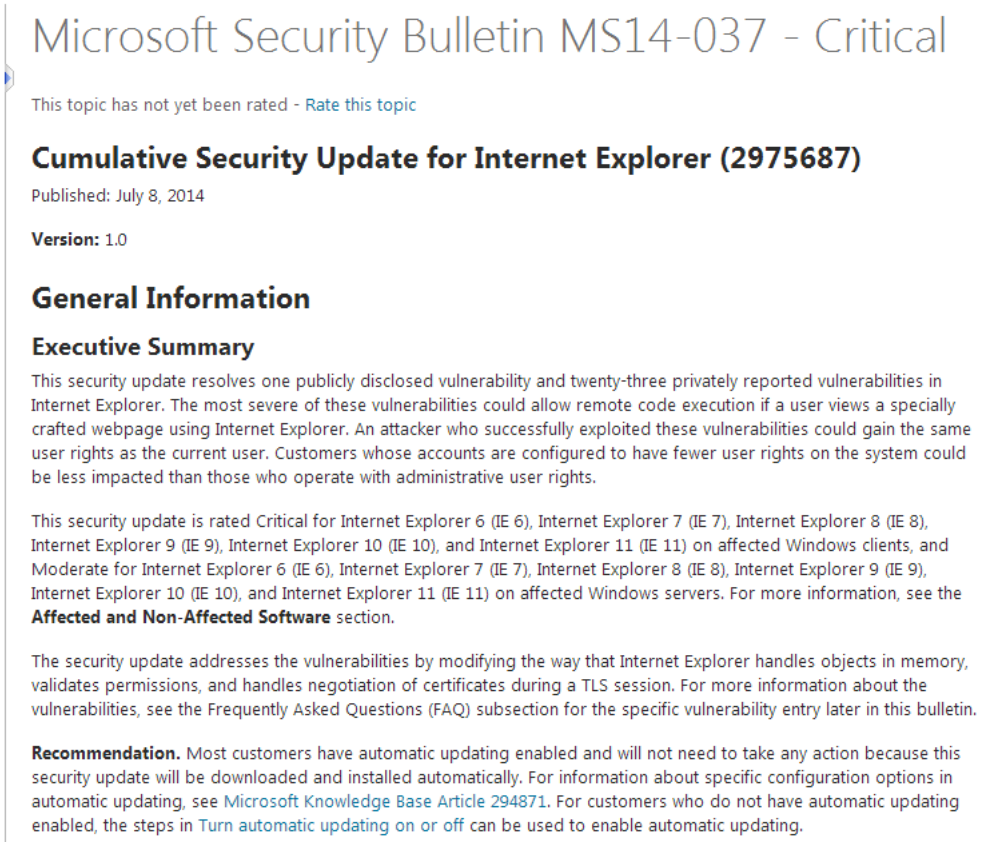
Executive Summaries

The following table summarizes the security bulletins for this month in order of severity.

For details on affected software, see the next section, **Affected Software**.

Bulletin ID	Bulletin Title and Executive Summary	Maximum Severity Rating and Vulnerability Impact	Restart Requirement	Affected Software
MS14-037	**Cumulative Security Update for Internet Explorer (2975687)** This security update resolves one publicly disclosed vulnerability and twenty-three privately reported vulnerabilities in Internet Explorer. The most severe of these vulnerabilities could allow remote code execution if a user views a specially crafted webpage using Internet Explorer. An attacker who successfully exploited these vulnerabilities could gain the same user rights as the current user. Customers whose accounts are configured to have fewer user rights on the system could be less impacted than those who operate with administrative user rights.	Critical Remote Code Execution	Requires restart	Microsoft Windows, Internet Explorer

3. Click the link in the **Bulletin ID** field. The Security Bulletin for that patch opens. In this example, Security Bulletin MS14-037 opens:



Microsoft Security Bulletin MS14-037 - Critical

This topic has not yet been rated - Rate this topic

Cumulative Security Update for Internet Explorer (2975687)

Published: July 8, 2014

Version: 1.0

General Information

Executive Summary

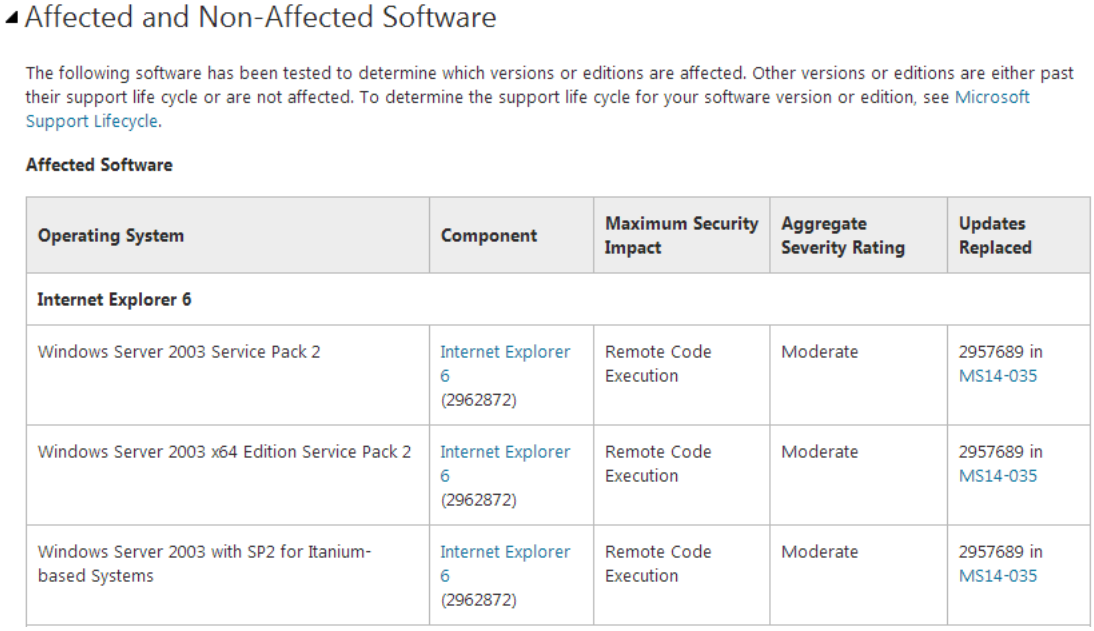
This security update resolves one publicly disclosed vulnerability and twenty-three privately reported vulnerabilities in Internet Explorer. The most severe of these vulnerabilities could allow remote code execution if a user views a specially crafted webpage using Internet Explorer. An attacker who successfully exploited these vulnerabilities could gain the same user rights as the current user. Customers whose accounts are configured to have fewer user rights on the system could be less impacted than those who operate with administrative user rights.

This security update is rated Critical for Internet Explorer 6 (IE 6), Internet Explorer 7 (IE 7), Internet Explorer 8 (IE 8), Internet Explorer 9 (IE 9), Internet Explorer 10 (IE 10), and Internet Explorer 11 (IE 11) on affected Windows clients, and Moderate for Internet Explorer 6 (IE 6), Internet Explorer 7 (IE 7), Internet Explorer 8 (IE 8), Internet Explorer 9 (IE 9), Internet Explorer 10 (IE 10), and Internet Explorer 11 (IE 11) on affected Windows servers. For more information, see the **Affected and Non-Affected Software** section.

The security update addresses the vulnerabilities by modifying the way that Internet Explorer handles objects in memory, validates permissions, and handles negotiation of certificates during a TLS session. For more information about the vulnerabilities, see the Frequently Asked Questions (FAQ) subsection for the specific vulnerability entry later in this bulletin.

Recommendation. Most customers have automatic updating enabled and will not need to take any action because this security update will be downloaded and installed automatically. For information about specific configuration options in automatic updating, see Microsoft Knowledge Base Article 294871. For customers who do not have automatic updating enabled, the steps in Turn automatic updating on or off can be used to enable automatic updating.

In the Security Bulletin, the **Affected Software** table lists the software affected by this patch, and provides a link to the download page for that specific patch:



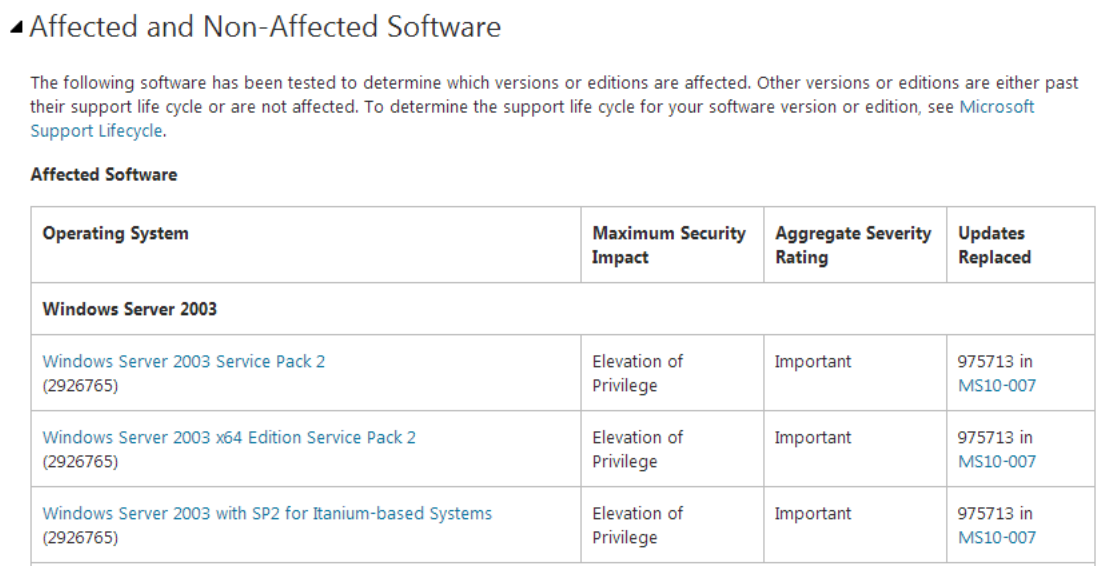
Affected and Non-Affected Software

The following software has been tested to determine which versions or editions are affected. Other versions or editions are either past their support life cycle or are not affected. To determine the support life cycle for your software version or edition, see Microsoft Support Lifecycle.

Affected Software

Operating System	Component	Maximum Security Impact	Aggregate Severity Rating	Updates Replaced
Internet Explorer 6				
Windows Server 2003 Service Pack 2	Internet Explorer 6 (2962872)	Remote Code Execution	Moderate	2957689 in MS14-035
Windows Server 2003 x64 Edition Service Pack 2	Internet Explorer 6 (2962872)	Remote Code Execution	Moderate	2957689 in MS14-035
Windows Server 2003 with SP2 for Itanium-based Systems	Internet Explorer 6 (2962872)	Remote Code Execution	Moderate	2957689 in MS14-035

For some patches, both an **Operating System** and **Component** are listed, while for others, only an **Operating System** is listed.



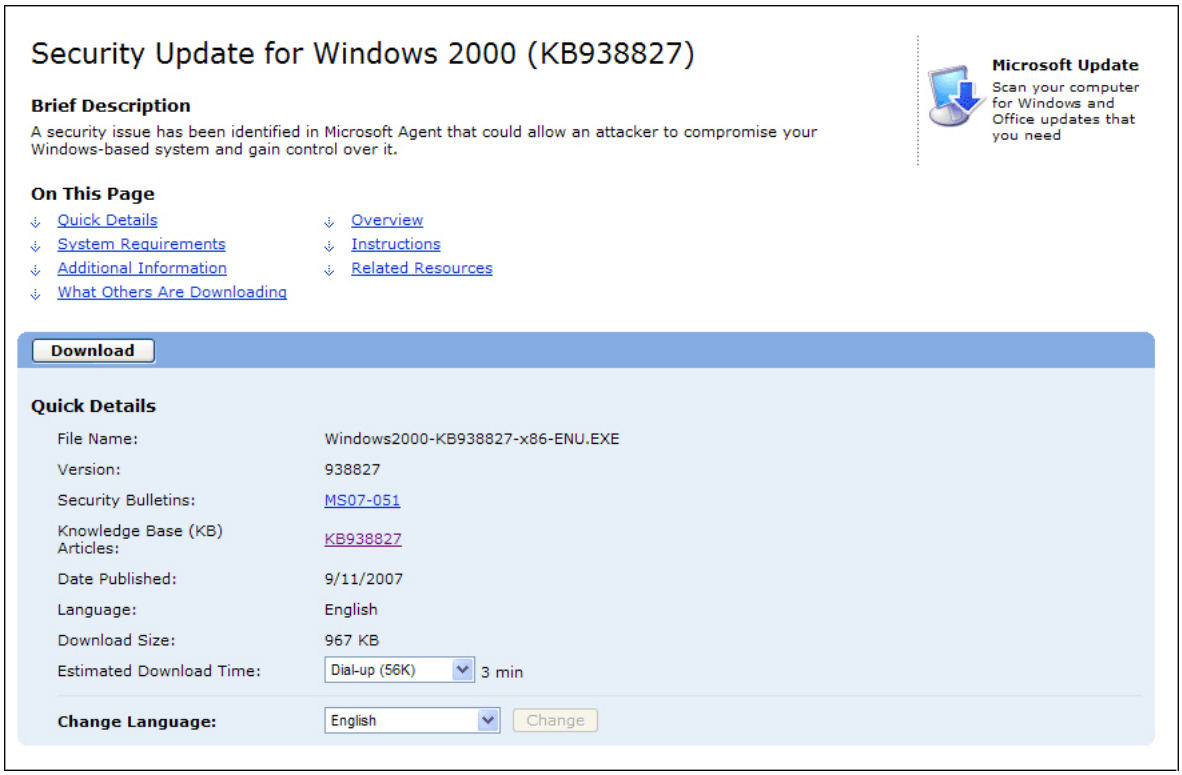
Affected and Non-Affected Software

The following software has been tested to determine which versions or editions are affected. Other versions or editions are either past their support life cycle or are not affected. To determine the support life cycle for your software version or edition, see Microsoft Support Lifecycle.

Affected Software

Operating System	Maximum Security Impact	Aggregate Severity Rating	Updates Replaced
Windows Server 2003			
Windows Server 2003 Service Pack 2 (2926765)	Elevation of Privilege	Important	975713 in MS10-007
Windows Server 2003 x64 Edition Service Pack 2 (2926765)	Elevation of Privilege	Important	975713 in MS10-007
Windows Server 2003 with SP2 for Itanium-based Systems (2926765)	Elevation of Privilege	Important	975713 in MS10-007

4. In the **Affected Software** table, click the link of the patch you want to import. The download page for that patch opens.



Security Update for Windows 2000 (KB938827)

Brief Description

A security issue has been identified in Microsoft Agent that could allow an attacker to compromise your Windows-based system and gain control over it.

Microsoft Update
Scan your computer for Windows and Office updates that you need

On This Page

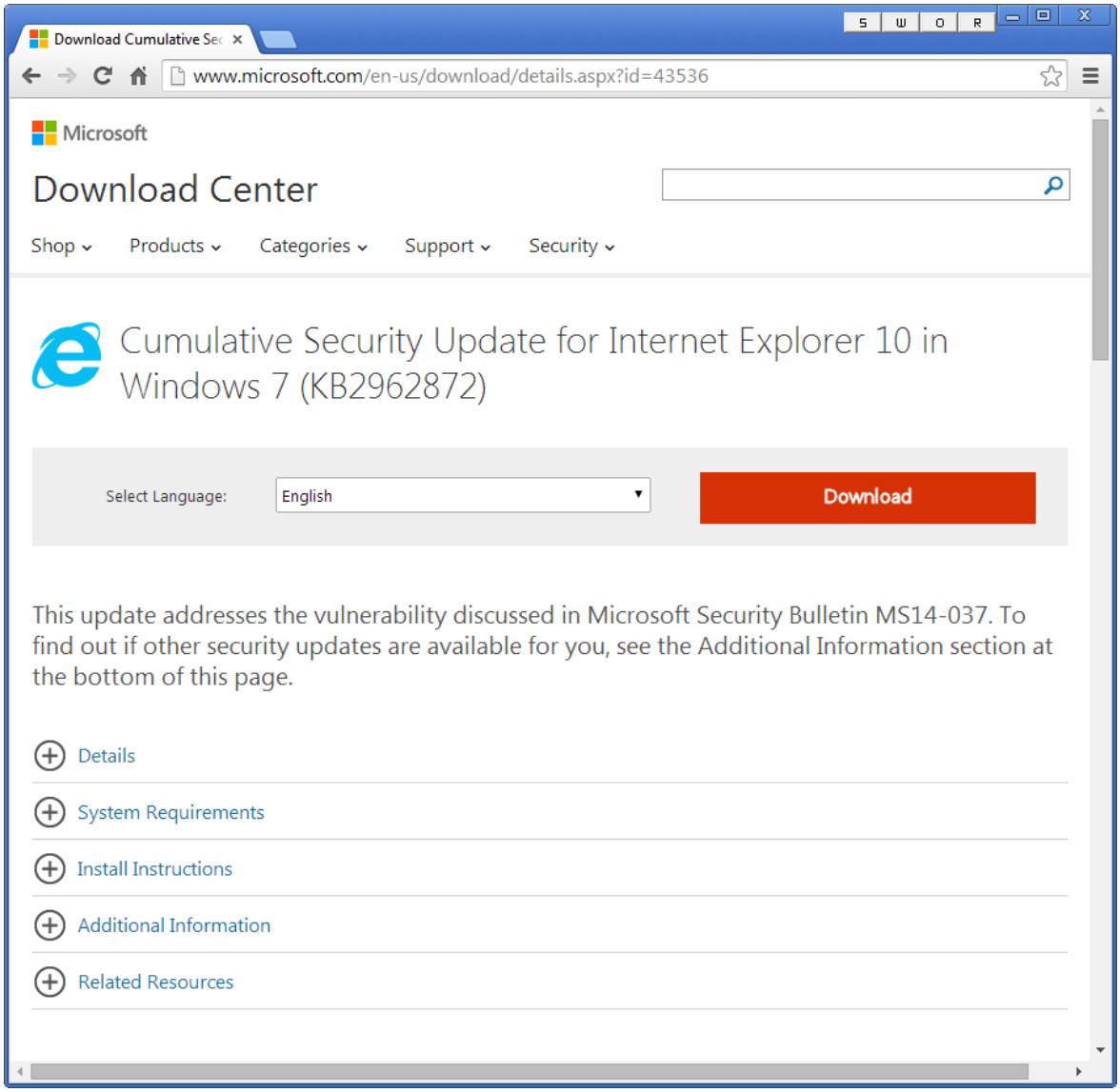
Quick Details
System Requirements
Additional Information
What Others Are Downloading
Overview
Instructions
Related Resources

Download

Quick Details

File Name:	Windows2000-KB938827-x86-ENU.EXE
Version:	938827
Security Bulletins:	MS07-051
Knowledge Base (KB) Articles:	KB938827
Date Published:	9/11/2007
Language:	English
Download Size:	967 KB
Estimated Download Time:	Dial-up (56K) 3 min
Change Language:	English Change

5. Click the **Download** button to download the patch.



6. Click **Download**. Download begins and the **Thank you for downloading** page opens.

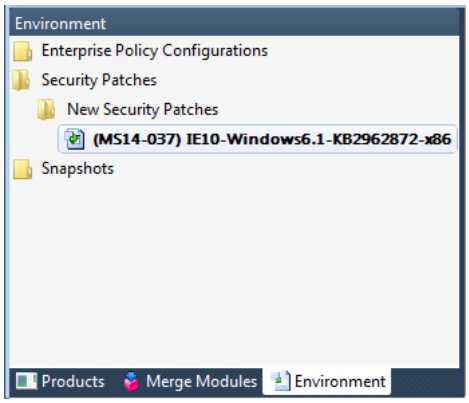
Importing a Microsoft Operating System Security Patch Into the Application Catalog

You can use the Import Wizard to import Microsoft operating system security patch files into the Application Catalog one at a time.

To import a Microsoft operating system security patch file into the Application Catalog using the Import Wizard, perform the following steps:

Task ***To import a Microsoft OS security patch file into the Application Catalog:***

1. On the **Home** tab, open the **Environment** tree and select either the **Security Patches** group or one of its subgroups.
2. Click **Import** in the ribbon. The **Security Patch File Selection** panel of the **Import Wizard** opens.
3. Click **Browse** and select a Microsoft patch file (`.msu`) that you have downloaded from the Microsoft Security TechCenter, as described in Identifying and Downloading Microsoft Operating System Patch Files.
4. Click **Next**. The **Summary** panel opens.
5. Click **Next** to begin the import. The **Running the Import** panel opens and displays progress messages.
6. When the import is complete, click **Finish** to close the Import Wizard. The patch is now displayed in the tree on the **Environment** tab.



Analyzing the Impact of Installing a Microsoft Operating System Patch

After you have imported a Microsoft operating system patch into the AdminStudio Application Catalog as described in Importing Microsoft OS Security Patch Files, you can use the Application Catalog Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system security patches and the packages and OS Snapshots in the Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft operating system patch is installed.

The section is organized in the following topics:

- Performing Patch Impact Analysis
- Viewing Patch Impact Analysis Results
- Viewing Patch and Patch Impact Information in Application Catalog
- Generating the Patch Report

Performing Patch Impact Analysis

You can use the Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system security patches and the packages and OS Snapshots in your Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft OS patch is installed.

Task ***To perform patch impact analysis:***

1. Import at least one Microsoft OS Security Patch file, as described in Importing a Microsoft Operating System Security Patch Into the Application Catalog.
2. On the **Analyze** tab, right-click on the application or group of applications that you want to perform patch impact analysis on, and select **Launch Patch Impact Analysis Wizard** from the shortcut menu. The Patch Impact Analysis Wizard will analyze the products you select here against the patches you will select.

   The **Welcome** panel of the **Patch Impact Analysis Wizard** opens.
3. Click **Next**. The **OS Snapshot Panel** opens.
4. Optionally, select an OS Snapshot to include in your analysis. The selected OS Snapshot will be used to identify specific file information for any patch impacts that are discovered. Only one OS Snapshot can be selected.

   Tip • *The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as "unknown". As such, it is recommended to include an OS Snapshot in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.*
5. Click **Next**. The **Source Patches Panel** opens.
6. On the **Source Patches Panel**, select the patches that you want to include in your analysis.

   When searching for patches to include, you can use the **Filter by product** list to restrict the patches displayed on this panel. Also, to view more information on a patch, select the patch and click the **Patch Properties** button to open the **Patch Properties** dialog box.
7. After you have selected the patches that you want to include in your analysis, click **Next**. The **Summary Information Panel** opens, listing a summary of the options you selected in the Patch Impact Analysis Wizard.
8. Click **Finish** to accept these options and begin the Patch Impact Analysis. Analysis messages are listed in the **Output** tab of the Output Window.

   When analysis is complete, patch conflicts are listed on the **Patch Impact** tab of the Output Window in table format.

Viewing Patch Impact Analysis Results

After Performing Patch Impact Analysis, patch conflicts are listed on the **Patch Impact** tab of the Output Window in table format.

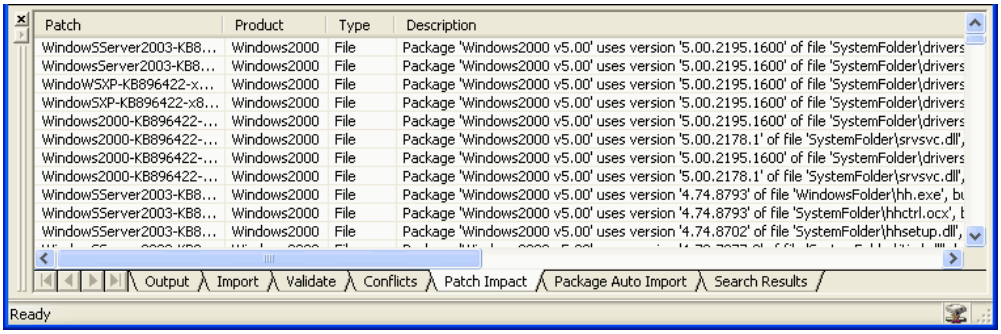


Figure 18-3: Sample Patch Impact Analysis Results

The following information is included in Patch Impact Analysis results:

Table 18-2 • Information Included in Patch Impact Analysis Results

Item	Description
Patch	Name of a Windows Installer Patch.
Product	Name of a Package or an OS Snapshot in the Application Catalog.
Type	Identifies the type of the impact as either a **File** or a **Registry** impact. ***Note*** • *Windows Installer Patches rarely impact Registry Entries, so most of the identified impacts will be identified as* ***File****.*

Table 18-2 • Information Included in Patch Impact Analysis Results

Item	Description
Description	Description of how the Windows Installer Patch impacted with the package or OS Snapshot. For example: `File 'CdrGfx.dll' in Package 'Coreldraw 12.0.0.458 v1.0' uses version 'Unknown' of file 'SystemFolder\RPCRT4.dll', but Patch 'WindowsXP-KB828741-x86-ENu.EXE' uses version '5.1.2600.1361'of the same file.` This warning message means that: • **The Windows Installer Patch**, `WindowsXP-KB828741-x86-ENu.EXE`, installs version `5.1.2600.1361` of `RPCRT4.dll`, an operating system file. and • **Corel Draw**, `Coreldraw 12.0.0.458 v1.0`, installs a file that is dependent upon the same operating system file: `RPCRT4.dll`. Therefore, this warning message means that you should evaluate Corel Draw on a system that includes this Windows Installer Patch to insure that you can safely distribute this package within your enterprise. ***Tip*** • *The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as "unknown". As such, it is recommended to include an OS Snapshot in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.*

Viewing Patch and Patch Impact Information in Application Catalog

Patch content and analysis information can be viewed on the Application Catalog **Products** and **Environment** tabs.

- Viewing Patch Content Information
- Viewing Associated Patches
- Viewing Patch Impacts on the Products Tab

Viewing Patch Content Information

To view patch content information, perform the following steps.

Task ***To view patch content information:***

1. Launch Application Catalog and, on the Home tab, open the **Environment** tree. The **Security Patches Group View** opens.
2. Expand the listing. All of the patches that have been imported into the Application Catalog are listed. Newly imported patches are listed in the **New Security Patches** Group.

   In Application Catalog, you can organize your patches into groups according to your business needs. See Organizing Your Application Catalog Using Groups.

3. Select a patch. The **Patches View** opens.

   The following information on the selected patch is listed:

   - **ID**—Microsoft Security Bulletin ID. Click on the ID number link to view this bulletin on the Microsoft website.
   - **Title**—Title of patch
   - **Release Date**—Date the patch was released by Microsoft.
   - **KB Article**—Microsoft Knowledge Base article ID. Click on the KB Article link to view that article on the Microsoft website.
   - **Imported On**—Date patch was imported into the Application Catalog
   - **Groups**—Listing of which groups this patch belongs to. A patch can belong to multiple groups. You can copy a patch into multiple groups.
   - **Description**—You can enter a description of the patch in this field.

4. In Application Catalog, you can view additional detailed patch information by right-clicking on a patch on the **Environment** tab and then selecting **Properties** from the shortcut menu. The **Patch Properties** dialog box opens.

   The following information is listed:

   - **General Tab**—View the title and a summary of a selected patch.
   - **Contents Tab**—Lists all of the files and registry data contained in the selected patch.
   - **Product Tab**—Lists the products that are updated by the selected patch.

Viewing Associated Patches

On the Associated Patches View, you can view a list of imported patches that, if installed, would update that product.

Task ***To view associated patches:***

1. Launch **Application Catalog** and open the **Analyze** tab in the ribbon.
2. Select the Windows Installer package that you want to examine. The **Analyze Deployment Type View** for that package opens.
3. Expand the product to view its constituent views and select the **Associated Patches** node. The Associated Patches View opens, displaying patches associated with that product.
4. If you double-click on a patch in the **Associated Patches View**, the **Security Patch View** (on the **Environment** tab) for that patch opens, listing general information on the selected patch.

   See Viewing Patch Content Information for more information.

Viewing Patch Impacts on the Products Tab

The information listed on the Patch Impact View depends upon the selection that is made in the **Impact category** list:

- When **Summary** is selected, the patches for which there is patch impact data persisted against the product are listed
- When **File Impacts** is selected, all impacts against this product are listed, and the patch that caused the impact is identified.

Task ***To view patch impacts:***

1. Launch Application Catalog and open the **Analyze** tab in the ribbon.
2. Select the Windows Installer package that you want to examine. The **Analyze Deployment Type View** for that package opens.
3. Expand the package to view its constituent views and select the **Patch Impacts** node. The **Patch Impact View** opens.
4. Select **Summary** from the **Impact category** list to view a list of patches for which there is patch impact data persisted against the product.
5. Select **File Impacts** from the **Impact category** list to view a list of all impacts against this product and the patch that caused the impact.
6. If you double-click on one of the patches in the list, the **Patch View** for that patch will open.

   If no patch impacts have been identified for this product, **File Impacts** will not be listed in the **Impact category** list.
7. To perform patch impact analysis, right-click on the product in the product tree on the **Analyze Deployment Type View**, and then select **Launch Impact Analysis Wizard** from the shortcut menu.

Note • *All patch information displayed in Application Catalog comes from the Application Catalog (for imported patches); no information about patches from mssecure.xml file is displayed in Application Catalog.*

Generating the Patch Report

In Application Catalog you can generate a Patch Report which lists detailed information about each patch that has been imported into the Application Catalog, including patch impact analysis information.

The Patch Report is generated in Web Archive format (`.mht`), a single, stand-alone HTML file that can be easily viewed in a Web browser and copied and emailed throughout your organization. The report is also printer-friendly.

Task ***To generate a Patch Report:***

1. Launch Application Catalog and click on the **Environment** tab. The **Security Patches Group View** opens.
2. Expand the patch listing and select the patch that you want to generate a report on. The **Security Patch View** opens.
3. Right-click on the patch and select **Generate Report** from the shortcut menu. The **Save Patch Report As** dialog box opens, prompting you to select a location for the `.MHT` file.
4. Confirm the report name and location and click **Save**. The report will be generated and will open in a new browser window. Click on the icons to expand or contract that section of the report. The report includes the following sections:

   Table 18-3:

Icon	Name	Description
	General Information	Includes patch title, Microsoft Security Bulletin ID, URL of the Microsoft Security Bulletin, patch summary, and the date the patch was released by Microsoft.
	Products Updated	Products updated by the patch.
	Files	Files included in the patch.
	Registry Entries	Registry entries that are added or modified by the patch.
	Impacts	Products that were checked for impacts during Patch Impact Analysis, and impacts that were detected during Patch Impact Analysis.

5. To print the report with all sections expanded, click the **Print Page** icon.

Reference

This Reference section includes the same topics that are displayed when you click a help button from a panel of the **Patch Impact Analysis Wizard** or from the **Patch Properties** dialog box. Reference information is organized into the following areas:

- Patch Impact Analysis Wizard
- Patch Properties Dialog Box

Patch Impact Analysis Wizard

You can use the Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system security patches and the packages and OS Snapshots in your Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft OS patch is installed.

The Patch Impact Analysis Wizard consists of the following panels:

- Welcome Panel
- OS Snapshot Panel
- Source Patches Panel
- Target Products Panel
- Summary Information Panel

When run, the output report is displayed on the **Patch Impact** tab of the Application Catalog Output Window.

Welcome Panel

The first panel of the Patch Impact Analysis Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 18-4 • Patch Impact Analysis Wizard Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Patch Impact Analysis Wizard panel.

OS Snapshot Panel

On the OS Snapshot Panel, you can optionally select an OS Snapshot to be used to identify specific file information for any patch impacts that are discovered.

The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as “unknown”. As such, *it is recommended to include an OS Snapshot* in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.

The following options are included:

Table 18-5 • OS Snapshot Panel Options

Option	Description
Group/OS Snapshot Tree	Select an OS Snapshot to include in your Patch Impact Analysis.

Source Patches Panel

On the Source Patches Panel, you select the patches that you want to include in your analysis.

The following options are included:

Table 18-6 • Source Patches Panel Options

Option	Description
Filter by Product	Lists all the Microsoft operating systems that have a patch that has been imported into the Application Catalog. Make a selection from this list to restrict the list of patches displayed on this panel.
Patch Listing	Patches that meet the selected filter criteria are listed. The following information is provided: • **Id**—Number identifying the Microsoft OS patch. Select the check box in this column to include this patch in your analysis. • **Name**—Name of the patch file. • **Title**—Description of the purpose of the patch file. • **Release Date**—Date that this patch was released by Microsoft.
Select All Button	Click to select all listed patches.
Clear All Button	Click to unselect any selected patches.
Patch Properties	Click to access the Patch Properties Dialog Box for this patch, which provides the following information: • **General Tab**—Summary information on the patch. • **Contents Tab**—Listing of the DLL files and registry entries associated with this patch. • **Products Tab**—A listing of the Product and that product's service packs that this patch is associated with.

Target Products Panel

On the Target Products Panel, select the products or groups of products that you want to perform patch impact analysis on. The Patch Impact Analysis Wizard will analyze the products you select here against the patches you selected on the **Source Patches Panel** for impacts.

The following options are included:

Table 18-7 • Target Products Panel Options

Option	Description
Group/Product Tree	A listing of all groups and products in the open Application Catalog.
Select All Button	Click to select all listed groups and products.
Clear All Button	Click to unselect all selected groups or products.

Summary Information Panel

The Summary Information Panel lists a summary of the options you selected in the Patch Impact Analysis Wizard.

Click **Finish** to begin the Patch Impact Analysis.

Patch Properties Dialog Box

You can access the **Patch Properties** dialog box from several locations:

- **Source Patches Panel of the Patch Impact Analysis Wizard**—Select a patch and then click the **Patch Properties** button.
- **Application Catalog Environment Tab**—Right-click a patch and then click **Properties**.

The **Patch Properties** dialog box consists of the following tabs:

Table 18-8 • Patch Properties Dialog Box Tabs

Tab	Description
General Tab	View the title and a summary of a selected patch.
Contents Tab	Lists all of the files and registry data contained in the selected patch.
Products Tab	Lists the products that are updated by the selected patch.

General Tab

The **General** tab of the **Patch Properties** dialog box lists the patch **Title** and includes a **Summary** of the purpose of the patch and the patch **Release date**. From the **General** tab, you can also click a link to go directly to the Microsoft website and view the Microsoft Security Bulletin and Microsoft Knowledge Base article for that patch:

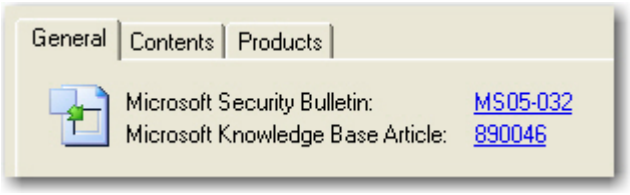


Figure 18-4: Links to Microsoft Security Bulletin and Knowledge Base Article

Contents Tab

The **Contents** tab lists all of the files and registry data contained in the selected patch. The following options are listed:

Table 18-9 • Patch Properties Dialog Box / Contents Tab Options

Option	Description
Files	This section lists all of the files included in the patch. The following information is displayed for each file: • **File**—File name. • **Directory**—Location where file will be installed. • **Version**—File version.
Registry data	The registry is a database repository for information about a computer's configuration. This section lists all of the registry data included in the patch. The following information is listed: • **Key**—Name of the registry key. • **Name**—Name of the registry value. • **Value**—Data stored for the registry value.

Products Tab

The **Products** tab lists all of the products updated by this patch, and each product's associated Service Packs. This tab includes the following options:

Table 18-10 • Products Tab Options

Option	Description
Products	All of the products updated by this patch are listed. Select a product from the list to see its associated Service Packs.
Service Packs	Listing of all of the Service Packs associated with the selected product.

19

Isolating Applications Using Application Isolation Wizard

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

Application Isolation Wizard user documentation is presented in the following sections:

Table 19-1 • Application Isolation Wizard User Documentation

Section	Description
About Application Isolation Wizard	Explains the reasons you would isolate applications and introduces you to the Application Isolation Wizard.
Launching the Application Isolation Wizard	Explains how to open the Application Isolation Wizard from the AdminStudio interface.
Isolation Methods	Describes the two isolation methods used by Application Isolation Wizard: Manifests and Assemblies, and Windows Installer Isolated Components.
Assemblies	Explains how Assemblies are used.
Manifests	Explains how Manifests are used.
Digital Signatures	Explains how Digital Signatures are used.
Isolating Applications	Describes how to use the Application Isolation Wizard to isolate applications.
Setting Assembly Naming Conventions	Explains assembly naming conventions.
Modifying the Default Isolation Recommendations	Describes how to modify the default isolation recommendations when using Windows Installer isolated components and when using manifests for isolation.

Table 19-1 • Application Isolation Wizard User Documentation (cont.)

Section	Description
Filtering File Listings when Manually Configuring Isolation	Explains how to filter the file listing when manually configuring application isolation using manifests, and when using Windows Installer isolated components.
Servicing Published Shared Assemblies	Explains how to service (alter) shared assemblies after publishing to update them as necessary.
Application Isolation Wizard Reference	Describes Application Isolation Wizard panels and dialog boxes.

About Application Isolation Wizard

Application isolation is one solution to component versioning conflicts, commonly known as "DLL Hell." Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. This is accomplished by providing DLLs and other shared components for specific applications, and placing information traditionally stored in the registry into other files that specify the locations of these isolated components. Application isolation provides increased stability and reliability for applications because they are unaffected by changes caused by installation and ongoing maintenance of other applications on the system.

Depending on the isolation method used in the Application Isolation Wizard, you can partially or totally isolate an application. When using assemblies and manifests to isolate applications for Windows XP systems, the assemblies can be updated following deployment without necessitating application reinstallation.

Reasons to Isolate Applications

You would want to isolate an application if:

- You want to resolve incompatibilities between different versions of shared components.
- You want to reduce the complexity of the installation by storing COM activation data in a manifest instead of the registry.
- You want to insulate the application from changes to shared components.

Tip *• Following isolation, you can use the Dynamic Dependency Scanner in InstallShield Editor to verify isolated files are loaded from a different directory.*

Reasons Not to Isolate an Application

You would not want to isolate an application if, following application isolation, you discover that the application no longer works because of an internal dependency on a component that has been moved during the isolation process.

Isolating Repackaged Setups Using Repackager

Application Isolation Wizard is a stand-alone tool which accepts a Windows Installer package as input and outputs a new, isolated Windows Installer package. You can also generate an isolated version of a repackaged setup immediately after the build step in Repackager.

If you open a Repackager project and choose the **Create an isolated version of the Windows Installer package** option on the Repackaged Output View, Repackager builds an isolated version of the Windows Installer package immediately after building the non-isolated version.

Both methods of isolating a package are performed using the same Application Isolation Wizard functionality. However, the Application Isolation Wizard provides a user interface experience that allows the user to extend the initial “dependency scanning” process for identifying file isolation candidates, while in Repackager, you specify your assembly and digital signing isolation options on the Isolation Options dialog box, and then those selections are applied to all isolated packages created by Repackager.

For more information, see Isolating Windows Installer Packages.

Launching the Application Isolation Wizard

To launch the Application Isolation Wizard, perform the following steps.

Task ***To launch the Application Isolation Wizard:***

1. Launch AdminStudio.
2. From the Tools Gallery, click the Application Isolation Wizard icon on the left side.

The Application Isolation Wizard launches and you can immediately begin the application isolation process.

Isolation Methods

There are two isolation methods supported by the Application Isolation Wizard™: Manifests and Assemblies and Windows Installer Isolated Components.

Assemblies and Manifests

Application isolation using assemblies and manifests is the recommended isolation method for Windows XP. These assemblies and manifests provide the same end result as Windows Installer isolated components, but keep all information outside of the registry and do not require the components to be installed in the same folder as the application. This reduces the chance of errors after isolation resulting from how the application was written.

Assemblies and manifests only work under the Windows XP operating system.

Windows Installer Isolated Components

Application isolation using Windows Installer isolated components is for Windows 98 SE, Me, and 2000. It can also be used on Windows XP, but using assemblies and manifests is the better solution. The isolated component method copies shared files (typically DLLs) into an application's folder instead of a shared location. The application then uses these files instead of global ones, preventing modifications made by other applications from affecting the shared files. As a result, the application always uses the versions of these files with which it was deployed.

To instruct an application use the private files rather than shared versions, the Application Isolation Wizard populates the IsolatedComponent table with the necessary logic to use private files stored in the same folder as the application. When Windows Installer performs the setup, data from the IsolatedComponent table populates a .local file, which ultimately directs how to use the private files.

Windows Installer isolated components still require some information to be written to the registry, and also require the associated components to be in the same folder as the application. While in most cases this will still provide required isolation, depending on how the application was written, the movement of these associated components from their original locations may prevent the application from functioning correctly.

Assemblies

Assemblies are DLLs or other portable executable files that applications require to function. Under Windows XP, these can be either shared or private. Private assemblies are typically stored in the same directory as the application they support. Shared assemblies are stored in the WinSxS directory, and are digitally signed.

By creating manifests for assemblies, the Application Isolation Wizard™ allows you to create self-contained applications that can use different versions of the same DLL or other portable executable, without any version conflicts.

Shared Assemblies

Shared assemblies are assemblies available to multiple applications on a computer. Applications that require these assemblies specify their dependence within a manifest. Multiple versions of shared assemblies can be used by different applications running simultaneously.

These assemblies are stored in the WinSxS directory, and must be digitally signed for authenticity. After deployment, the version of shared assemblies can be changed, allowing for changes in dependencies.

Private Assemblies

Private assemblies are assemblies created for exclusive use by an application. They are accompanied by an assembly manifest, which contains information normally stored in the registry. Private assemblies allow you to totally isolate an application, eliminating the possibility that dependent files may be overwritten by other applications.

These assemblies are always stored in the same location as their associated executable.

Manifests

The Application Isolation Wizard™ can create two types of manifests: application manifests and assembly manifests.

Table 19-2 • Manifest Types

Manifest Type	Description
Application	Application manifests are XML files that describe an isolated application. This descriptive information includes the relationship between the application and its dependent files. Typically, the naming convention for a manifest is ApplicationName.Extension.manifest. For example, if the application was HelloWorld.exe, the manifest file is called HelloWorld.exe.manifest.
Assembly	Assembly manifests are XML files that describe an application's assemblies. This includes components such as DLLs. Information stored in the assembly manifest, such as COM registration information, ProgIDs, etc., is usually stored in the Registry. However, by making it independent from the registry, only that application can use the dependent files described in the manifest. This enables you to have multiple versions of the same DLL or other portable executable file on a system without generating compatibility conflicts. Typically, the naming convention for a manifest is AssemblyName.Extension.manifest. For example, if the component was Goodbye.dll, the manifest file is called Goodbye.dll.manifest.

Manifests as New Components

When you create manifests, the Application Isolation Wizard supports putting them into new components. If you do not select this option from the Advanced Options dialog box, the manifest will be added to the same component as the assembly.

Digital Signatures

Like conventional signatures, digital signatures identify you (or your organization) to end users. In the context of application isolation, a digital signature identifies you or your organization as the creator of shared assemblies. This ensures that the identity of a shared assembly can be verified for authenticity. Digital signatures in the Application Isolation Wizard™ require a combination of a digital certificate and a code signing technology.

Certificates

Digital certificates identify you and/or your company to end users to assure them the assembly they are about to use has not been altered. They are issued by a certification authority such as VeriSign, or created using a combination of software publishing credentials (`.spc`) and a private key (`.pvk`), both also issued by a certification authority. The certificate includes the public cryptograph key, and, when used in combination with a private key, can be used by end users to verify the authenticity of the signor.

You can create a certificate file from the constituent PVK and SPC files and import it into the Certificate Store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the certificate store.

Caution • *Certificate files must be 2048-bit or higher. For more information, see the article: Assembly Signing Example on the Microsoft Developer Network website.*

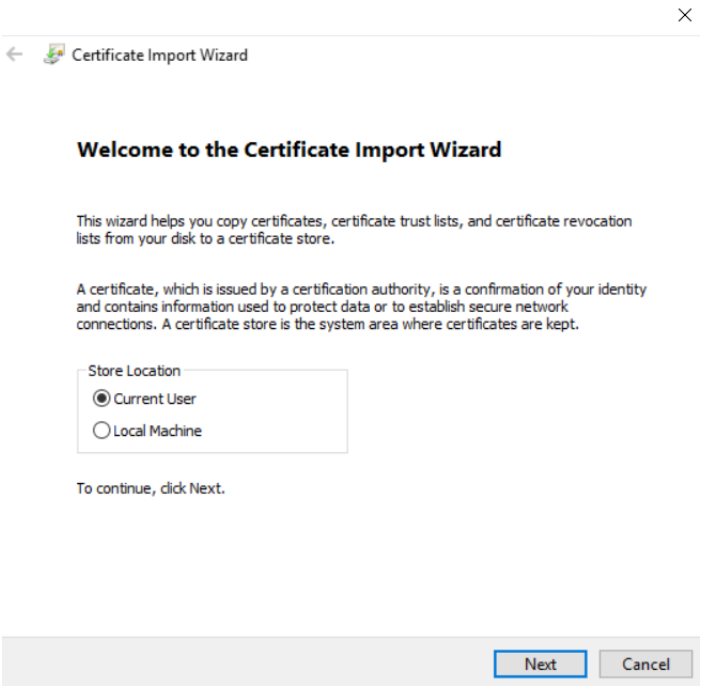
Import Certificate

You can create a certificate file from the constituent PVK and SPC files and import it into the Certificate Store using the PVK Digital Certificate Files Importer.

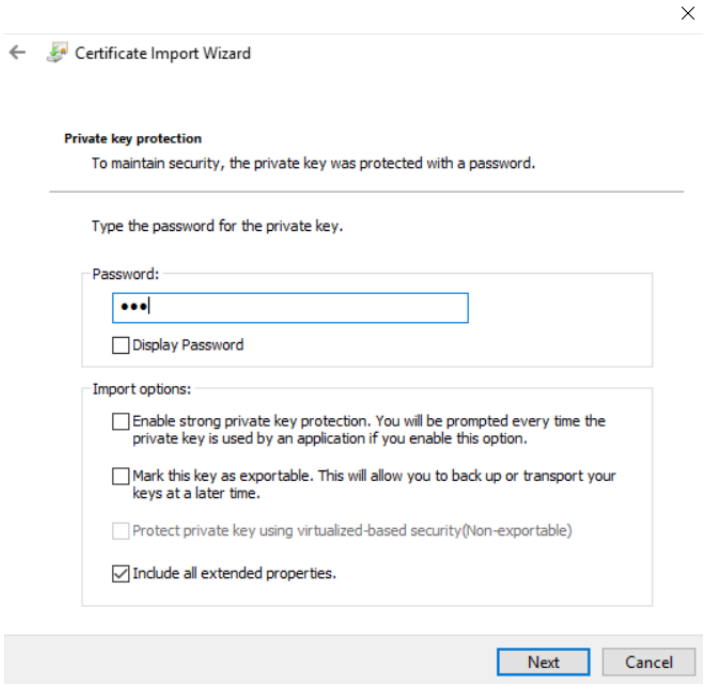
To import certificate, perform the following steps:

Task ***To Import Certificate:***

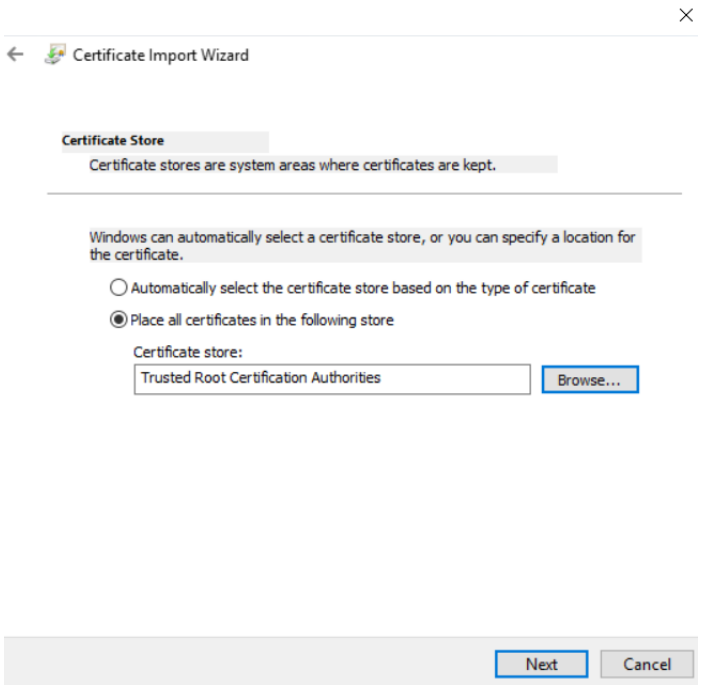
1. Choose the certificate which you want to import and double click. The **Certificate Import Wizard** appears.
2. Select the store location either **Current User** or **Local Machine** and click **Next**.



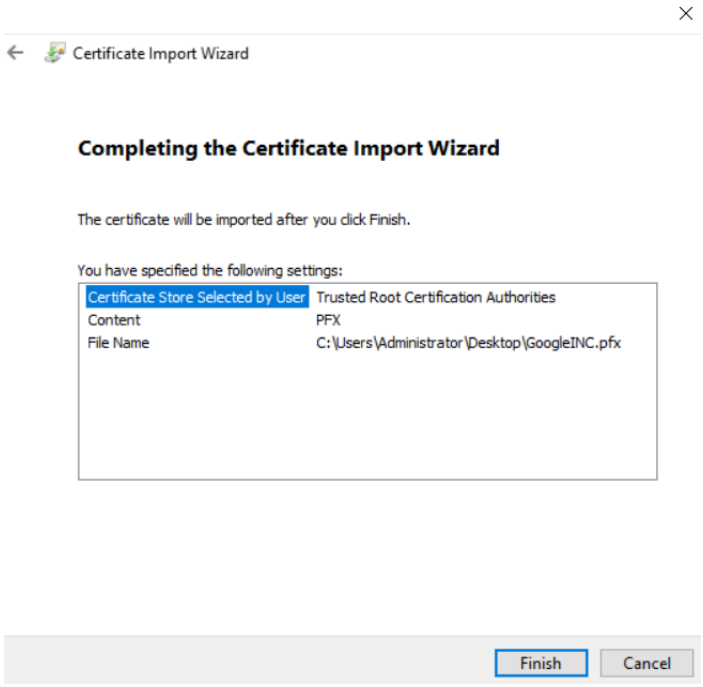
3. In the **File to Import** page, optionally, specify the file you want to import. Click **Next**.
4. In the **Private key protection** page, enter the certificate password, and then click **Next**.



5. In the **Certificate Store** page, select **Place all Certificate in the following store** radio button. The **Certificate Store** field appears, browse and select the folder were you want to install the certificate. Click **Next**.



6. Click **Finish** to import the certificate.



7. Click **Ok** and agree the confirmation dialog which appears.
8. Upon successful action, the certificate will be imported in store specified.

Code Signing Technologies

The Application Isolation Wizard™ supports two code signing technologies:

Table 19-3 • Supported Code Signing Technologies

Technology	Description
Credentials	Credentials consist of both Software Publishing Credentials (.spc file) and a private key (.pvk file). These two files are required in conjunction with the certificate to sign shared assemblies.
Certificate Name in the Store	Using Microsoft's Certificate Store technology, the combined software publishing credentials and private key can be placed in a repository for multiple uses. The name of the certificate is provided as opposed to the constituent files in the Credentials code signing technology.

Software Publishing Credentials

You must supply a certification authority with specific information about your company and software to obtain software publishing credentials in the form of an .spc file. Your software publishing credentials are used to generate a digital signature for your assembly.

The .spc file and .pvk (private key) file you enter in the Digital Signature tab of the Advanced Options dialog box compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Certificate Store

To perform code signing, both private key and software publishing credential information must be supplied. This must occur each time a package is signed. However, the certificate store serves as a central repository for this information, allowing you to associate the same credentials and key with multiple packages. This simplification is particularly useful when isolating applications, as typically the code signing information will be identical for all shared assemblies. Ultimately, the certificate store removes the burden of managing private key and software publishing credential information.

You can create a certificate file from the constituent PVK and SPC files and import it into the certificate store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the certificate store.

Private Keys

A private key (a file with the extension .pvk) is granted by a certification authority. The Application Isolation Wizard™ uses the private key you enter in the Digital Certificates tab of the Advanced Options dialog box to digitally sign your shared assembly and ensure end users of its content's authenticity.

The .spc (Software Publishing Credentials) file and .pvk file you enter in the Digital Signature tab compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Isolating Applications

To isolate applications within a Windows Installer package or a merge module, perform the following steps.

Task ***To isolate applications within a Windows Installer package (.msi) or merge module (.msm):***

1. Launch the Application Isolation Wizard™. The **Welcome Panel** appears.
2. From the **Welcome Panel**, click **Next**. The **Windows Installer File Selection Panel** appears.
3. From the Windows Installer File Selection Panel, specify the Windows Installer package (`.msi`), Windows Installer self-extracting executable file (`setup.exe`), merge module (`.msm`) containing applications you want to isolate. Click **Next**. The Isolation Method panel appears.
4. From the **Isolation Method Panel**, select the isolation method(s) you want to use.
5. If you are using manifests, you can click Advanced to configure manifest properties and digital signature information (if required) on the Advanced Options Dialog Box.
6. Click **Next**. The **Summary Information Panel** appears.
7. From the **Summary Information Panel**, confirm the isolation selections.
8. If you want to manually configure isolation, click Modify.

- **If you are using manifests** to isolate your application—either alone or in conjunction with Windows Installer isolated components—the **Manifest and Assembly Design** dialog box appears.
- **If you are only using Windows Installer isolated components** to isolate the application, the **Isolated Components Design** dialog box appears. After you have completed manually configuring the isolation, click OK to return to the Summary Information Panel.

9. Click Isolate. The **Application Isolation Progress Panel** appears.

   When the Application Isolation Wizard is complete, the **Completing the Application Isolation Wizard Panel** is displayed, providing feedback on whether the Application Isolation Wizard was successful.

10. From the **Completing the Application Isolation Wizard Panel**, click **Finish**.

Setting Assembly Naming Conventions

To set the default naming convention for assemblies, perform the following steps.

Task ***To set the default naming convention for assemblies:***

1. Launch the Application Isolation Wizard™. The **Welcome Panel** opens.
2. From the Welcome panel, click Next. The **Windows Installer File Selection Panel** opens.
3. From the Windows Installer File Selection panel, specify the Windows Installer package (.msi) or merge module (.msm) containing applications you want to isolate. Click Next. The **Isolation Method Panel** opens.
4. Select the Use manifests for isolation option.
5. Click Advanced. The Manifest Options tab of the **Advanced Options** dialog box opens.
6. Enter your Company name and Division. These two fields create the default assembly naming convention (in the form “Company.Division.Assembly” followed by a number).

   Note • *To edit the Assembly Name, you can also click Modify from the Summary Information Panel later in the Wizard to open the Manifest and Assembly Design dialog box, and then click Properties to open the Application Manifest Properties dialog box, where you can edit the Assembly Name.*

7. Click OK. You are returned to the **Isolation Method Panel**.
8. Click Next. The **Summary Information Panel** opens.
9. Click Isolate to proceed with isolation using the specified naming convention.

Assemblies created during application isolation will follow the naming convention as specified.

Modifying the Default Isolation Recommendations

You can modify default isolation recommendations for the following:

- When Only Using Windows Installer Isolated Components

- When Using Manifests for Isolation

When Only Using Windows Installer Isolated Components

To modify the default isolation recommendations when only using Windows Installer isolated components, perform the following steps.

Task ***To modify the default isolation recommendations:***

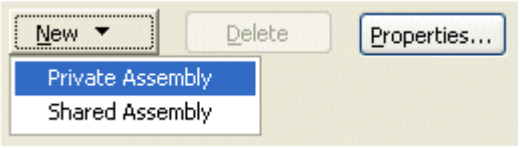
1. From the **Summary Information Panel** of the Application Isolation Wizard™, click **Modify**. The **Isolated Components Design** dialog box is displayed.
2. Select the Applications to be Isolated and then select the Files to Isolate for Selected Application. Repeat as necessary.
3. Click OK. When you return to the Summary Information Panel, verify your settings before isolating.

When Using Manifests for Isolation

To modify the default isolation recommendations when using manifests for isolation, perform the following steps.

Task ***To modify the default isolation recommendations when using manifests for isolation:***

1. From the **Summary Information Panel** of the Application Isolation Wizard, click Modify. The **Manifest and Assembly Design** dialog box is displayed.
2. If you need to create a new assembly, select the application for which you want to create the assembly, click New, and select the assembly type: Private Assembly or Shared Assembly.



   A new assembly is listed under the selected application.
3. Select the new assembly then add or remove files in the assembly.
4. Click OK. When you return to the Summary Information panel, verify your settings before isolating.

Filtering File Listings when Manually Configuring Isolation

Task ***To filter the file listing when manually configuring application isolation using manifests:***

1. In the **Manifest and Assembly Design** dialog box, select an assembly from the application tree.
2. Directly below the Files to be added in an assembly window, specify the files displayed from the Show filter.

Task ***To filter the file listing when manually configuring application isolation using Windows Installer isolated components:***

1. In the **Isolated Components** dialog box, select an application from the application tree.
2. Directly below the Files to Isolate for Selected Application window, specify the files displayed from the Show filter.

Servicing Published Shared Assemblies

Shared assemblies can be serviced (altered) after publishing to update them as necessary. This is accomplished using a publisher configuration, which overrides default configurations specified in the manifest.

For an exhaustive discussion of how to service shared assemblies, refer to the article How To Build and Service Isolated Applications and Side-by-Side Assemblies for Windows XP on the Microsoft Developer Network website (msdn.microsoft.com).

Application Isolation Wizard Reference

The Application Isolation Wizard scans Windows Installer packages (.msi) or merge modules (.msm) and isolates applications within them. Isolation ensures that applications always use the specific shared and support files with which they were installed. This prevents the overwriting of previous versions of shared components, and ensures that other applications do not overwrite the versions of shared and support files required by your application.

The following reference topics are available for the Application Isolation Wizard:

- Welcome Panel
- Windows Installer File Selection Panel
- Isolation Method Panel
- Summary Information Panel
- Application Isolation Progress Panel
- Completing the Application Isolation Wizard Panel
- Advanced Options Dialog Box
- Manifest and Assembly Design Dialog Box
- Isolated Components Design Dialog Box
- Assembly Properties Dialog Box
- Application Manifest Properties Dialog Box

Welcome Panel

The Welcome panel is the first panel displayed when you launch the Application Isolation Wizard. It provides a general explanation of application isolation.

Click Next to proceed to the **Windows Installer File Selection Panel**.

Windows Installer File Selection Panel

Enter the full path and file name of the Windows Installer package (.msi) or merge module (.msm) that you want the Application Isolation Wizard to scan for isolation candidates. Alternately, click Browse to navigate to the file.

Click Back to return to the **Welcome Panel**; click Next to proceed to the **Isolation Method Panel**.

Isolation Method Panel

Select the application isolation method(s) you want to use for this Windows Installer package or merge module.

Make the following selections under **Manifests** and **Windows Installer Isolated Components**:

Table 19-4 • Isolation Method Panel Selection Guidelines

If Installation Will Be Deployed ...	Manifests	Windows Installer Isolated Components
Only on **Windows 2003 Server**	Select	Do Not Select
Only on **Windows XP**	Select	Do Not Select
Windows 98 SE, **Me**, and/or **Windows 2000** but not **XP**	Do Not Select	Select
Windows XP *and* **Windows 98 SE**, **Me**, and/or **Windows 2000** ***Note*** • *Your installation package size will increase, but application isolation will work on the appropriate operating systems.*	Select	Select

If you use **Manifests**, click **Advanced** to display the **Advanced Options** dialog box, from which you can configure manifest options and digital signature information.

Summary Information Panel

From this panel, review a summary of your selections prior to isolation.

For granular control over the isolation process, click Modify. If you are only using Windows Installer Isolated Components as the isolation method, the **Isolated Components Design** dialog box appears. Otherwise, the **Manifest and Assembly Design** dialog box is displayed.

Click Back to return to the **Isolation Method Panel**; click Isolate to isolate the application according to your settings. The Application Isolation Progress panel is displayed.

Application Isolation Progress Panel

During application isolation, the progress is displayed on this panel. Information about the applications, assemblies (if using manifests as the isolation method), and files is displayed above the progress bar.

Upon isolation completion (or failure), the **Completing the Application Isolation Wizard Panel** is displayed.

Completing the Application Isolation Wizard Panel

The final panel in the Application Isolation Wizard provides feedback on whether the Application Isolation Wizard was successful.

If the Application Isolation Wizard was successful, the names and locations of the original and output packages are provided. If the Wizard was not successful, this panel informs you that the selected components could not be isolated.

Advanced Options Dialog Box

The **Advanced Options** dialog box, available from the **Isolation Method** and **Ready to Isolate** panels of the Application Isolation Wizard, allows you to configure assembly types, naming conventions, and digital signature options. The **Advanced Options** dialog box presents these options on two tabs: **Manifest Options** and **Digital Signature**.

Once you have finished configuring advanced options, click OK to save your changes, or Cancel to close the dialog box without saving your modifications. When the dialog box closes, you are returned to the panel where you clicked Advanced.

Manifest Options Tab

The Manifest Options tab, available in the **Advanced Options** dialog box, allows you to configure several settings associated with manifests.

These settings include:

Table 19-5 • Manifest Options Tab Options

Option	Description
Assembly Type	This option allows you to select the type of assemblies that Application Isolation Wizard™ will create and use: • Create private side-by-side assemblies in the application folder • Create shared side-by-side assemblies in the WinSxS folder (Default) If you want to use both assembly types, you need to manually configure assemblies from the Manifest and Assembly Design Dialog Box. ***Note*** • *Manifests for shared assemblies must be digitally signed. This can be done in the Digital Signature Tab.* ***Note*** • *A 2048-bit key is required to sign a Windows XP assembly/manifest being installed to the WinSxS folder.*
Assembly Naming Conventions	Specify your company and division information to define the default naming convention that Application Isolation Wizard will use when creating assemblies during application isolation By default, assembly names are specified in the form of: `Company.Division.Assembly`
Create a new component for each assembly	Select this option if you want to create a new component for each assembly created during isolation. This check box applies to all assemblies created. Individual assemblies can be configured from the **Assembly Properties** dialog box on a per-assembly basis. ***Caution*** • *If you are creating assemblies for applications files within multiple components, this option must be selected for successful application isolation.* *If you are planning to deploy this isolated package to operating systems prior to Windows XP, always check this box.*

Digital Signature Tab

The Digital Signature tab, available in the **Advanced Options** dialog box, allows you to configure the certificate information required when using shared assemblies. This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

Caution • *The Application Isolation Wizard™ uses timestamping when signing global assemblies. Consequently, you must have an Internet connection on the computer when you create a global assembly.*

You must configure the following options when signing these assemblies:

Table 19-6 • Digital Signature Tab Options

Option	Description
Certificate File	Click the Browse (…) button next to the field and navigate to the certificate file you are using to sign assemblies. A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: **SPC File** and **PVK** File. ***Note*** • *In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as such as VeriSign, with specific information about your company and software.*
SPC File	Specify the name and location of your software publishing credentials file (`.spc`).
PVK	Specify the name and location of your private key file (`.pvk`).
Certificate Name in the Store	Select this option to use an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary.

Note • *A 2048-bit key is required to sign a Windows XP assembly/manifest being installed to the WinSxS folder.*

Manifest and Assembly Design Dialog Box

If you are using manifests to isolate your application, either alone or in conjunction with Windows Installer isolated components, the Manifest and Assembly design dialog box is displayed when you click Modify from the Summary Information panel.

When you first display this dialog box, the settings the Application Isolation Wizard™ recommends for this package are displayed. By default, only executables that will be installed in the SystemFolder will be selected for isolation. You can select an application contained in the Windows Installer or merge module and create a new private or shared assembly for that application. You can then select the files to isolate for the selected application. A filter at the bottom of the dialog box allows you to restrict the file types visible.

Click Properties to display the **Application Manifest Properties** dialog box. From this dialog box, you can configure the naming convention for assemblies and manifests, and specify whether you want manifests placed into separate components.

When you have finished performing manual configuration, click OK to return to the Summary Information panel.

Isolated Components Design Dialog Box

If you are only using Windows Installer isolated components to isolate the application, this dialog box is displayed when you click Modify from the Summary Information panel.

When you first display this dialog box, the settings the Application Isolation Wizard™ recommends for this package are displayed. By default, only libraries that will be installed in the SystemFolder will be selected for isolation. You can select an application contained in the Windows Installer or merge module, and then select the files to isolate for the selected application. A filter at the bottom of the dialog box allows you to restrict the file types visible.

When you have finished performing manual configuration, click OK to return to the Summary Information panel.

Assembly Properties Dialog Box

The Assembly Properties dialog box displays information about the manifest and assembly, and can be launched from the Manifest and Assembly Design dialog by selecting an assembly and clicking Properties.

The following groups contain configurable options:

Table 19-7 • Assembly Properties Dialog Box Options

Group	Description
Manifest Details	In the Manifest Details group, you can view the file name for the manifest. It is structured in the form “Company.Division.Assembly.manifest” by default.
Assembly Identity	The Assembly Identity group contains fields for the Assembly Name and Version. When you change the assembly name, the manifest file name changes.
Assembly Type	This group allows you to select whether the current assembly is private or shared. If it is shared, you must configure digital signature information in the Advanced Options dialog box.
Create new component	Select this option if you want to create a new component for this assembly. ***Caution*** *• If this assembly contains files that originate from multiple components, this option must be selected for successful application isolation.* *If you are planning to deploy this isolated package to operating systems prior to Windows XP, always check this box.* This check box applies only to this assembly. Global settings for assemblies can be configured from the **Manifest Options** tab of the **Advanced Options** dialog box.

Application Manifest Properties Dialog Box

The Application Manifest Properties dialog box displays information about the manifest and assembly, and can be launched from the Manifest and Assembly Design dialog by selecting an application and clicking Properties.

The following groups contain configurable options:

Table 19-8 • Application Manifest Properties Dialog Box Options

Options Group	Description
Manifest Details	In the Manifest Details group, you can view the file name for the manifest. It is structured in the form “Company.Division.Assembly.manifest” by default.
Assembly Identity	The Assembly Identity group contains fields for the Assembly Name and Version. When you change the assembly name, the manifest file name changes.

Command-Line Options

The Application Isolation Wizard can also be run from the command line. You can specify the following options when running the AIW.exe executable from the command line:

Table 19-9 • Application Isolation Wizard Command-Line Options

Option	Description
-?	Displays command-line help for the Application Isolation Wizard.
-version	Displays the version of AdminStudio.
-i <configuration file>	Allows you to specify a configuration file for Application Isolation Wizard settings. The default file, AIWConfig.ini, is located in <AdminStudio Directory>\Common and can be used as a model. This parameter is optional.
-p <package name>	The name and location of the package or merge module which includes applications you want to isolate. This parameter is mandatory.

Configuration Files

When using the command-line options for the Application Isolation Wizard, you can specify an INI file for configuration using the `-i` parameter. This file should take the following format:

```
[IsolationMethods]
Manifests=1
IsolatedComponents=1

[DigitalSignature]
CertificateFile=
SPCFile=
PVKFile=
```

```
CertificateName=
TimeStampAssemblies=

[Manifest]
AssemblyType=0
Company="Company"
Division="Division"
NewComponents=0
```

Each configuration corresponds to a user interface setting in the Application Isolation Wizard, as described below:

Table 19-10 • Configuration File Settings

INI File	UI Setting	Explanation
Manifests	Use manifests for isolation option on Isolation Method panel	Set this value to 1 to use manifests. Manifests only work with Windows XP.
IsolatedComponents	Use Windows Installer isolated components for isolation on Isolation Method panel	Set this value to 1 to use Windows Installer isolated components.
CertificateFile	Certificate File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the CER file.
SPCFile	SPC File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the SPC file
PVKFile	PVK File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the private key.
CertificateName	Certificate Name in the store field on the Digital Signature tab of the Advanced Options dialog box	Provide the name of the certificate from the certificate store.
TimeStampAssemblies	No corresponding UI setting	Set this value to 0 to disable timestamping during shared assembly creation; set it to 1 to enable timestamping. By default, the Application Isolation Wizard uses timestamping if this value is not configured.
AssemblyType	Assembly Type on the Manifest Options tab of the Advanced Options dialog box	Set this value to 0 to use private assemblies; set the value to 1 to use shared assemblies.
Company	Company field on the Manifest Options tab of the Advanced Options dialog box	Put the name of your company in quotes.
Division	Division field on the Manifest Options tab of the Advanced Options dialog box	Put the name of your division in quotes.

Table 19-10 • Configuration File Settings (cont.)

INI File	UI Setting	Explanation
NewComponents	Create new component option in Manifest Options tab of Advanced Options dialog box	Set the value to 1 to create a new component for each manifest.

Manifest Examples

Following are examples of both an application manifest and an assembly manifest:

Application Manifest Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
<assemblyIdentity type="win32" name="InstallShield.Development. AppAssembly4" version="1.0.0.1"
processorArchitecture="x86" />
<description>This manifest was generated by the Application Isolation Wizard</description>
- <dependency>
   - <dependentAssembly>
      <assemblyIdentity type="win32"name="InstallShield.
         Development.LocalAssembly1" version="1.0.0.1"
         processorArchitecture="x86" />
     </dependentAssembly>
  </dependency>
  </assembly>
```

Assembly Manifest Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <assembly xmlns="urn:schemas-microsoft-com:asm.v1"
     manifestVersion="1.0">
  <assemblyIdentity type="win32" name="InstallShield.Development.
     LocalAssembly1" version="1.0.0.1" processorArchitecture="x86" />
  <file name="IsCommonServices.dll">
- <comClass description="CabinetBuilder Class" clsid="
     {8D3FE200-DA96-11D3-BEE7-00105A996B4E}" progid=
     "ISHerculesCommonServices.CabinetBuilder.1"
     threadingModel="Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-
     00105A996B4E}">
     <progid>ISHerculesCommonServices.CabinetBuilder</progid>
  </comClass>
- <comClass description="Cabinet Class" clsid="
     {3C35E807-C92D-11D3-BEDF-00105A996B4E}" progid=
     "ISHerculesCommonServices.CabinetExtractor.1"
     threadingModel="Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-
     00105A996B4E}">
     <progid>ISHerculesCommonServices.CabinetExtractor</progid>
  </comClass>
- <comClass description="InstallShield Common Services Registry
     object" clsid="{3032B526-2C3D-11D4-AB2C-00C04F09719A}"
     progid="ISHerculesCommonServices.Registry.1" threadingModel=
     "Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-00105A996B4E}">
```

```
   <progid>ISHerculesCommonServices.Registry</progid>
</comClass>
</file>
</assembly>
```

20

Ensuring Package Quality Using QualityMonitor

Edition • *QualityMonitor is included with AdminStudio Professional and Enterprise Editions.*

QualityMonitor allows you to run a series of built-in tests to installed Windows Installer-based products, helping to ensure they run correctly. When failures occur, QualityMonitor can help identify where problems exist, and ultimately direct you to the solution.

QualityMonitor user documentation is presented in the following sections:

Table 20-1 • QualityMonitor User Documentation

Section	Description
About QualityMonitor	Explains the purpose and benefits of using QualityMonitor.
Creating New QualityMonitor Project Files	Explains how to create a new QualityMonitor project.
Opening Existing QualityMonitor Project Files	Explains how to open QualityMonitor project files, which have an .iqm extension.
Working with Test Cases	Describes the most common tasks you may perform when working with Test Cases.
Deployment Testing	Explains how to perform deployment tests against the installed product, ensuring that the product has been installed correctly.
Lockdown and Runtime Testing	Explains how to test an application when its target environment is restricted in some way, such as in a locked-down environment.
Using MSI Doctor to Verify Package Deployment Status	Explains how to use MSI Doctor to verify if an MSI package is installed properly. This helps prevent users from seeing an auto-repair dialog box when they run the application.

Table 20-1 • QualityMonitor User Documentation (cont.)

Section	Description
Creating Custom Test Cases	Explains how to add additional, custom Test Cases to projects—based on your business needs.
Test Reports	Explains how to create an HTML test report for the current project.
Running QualityMonitor from the Command Line	Explains how to run QualityMonitor from the command line.
QualityMonitor Reference	Provides detailed reference on each user interface element, dialog box, and view in QualityMonitor.

About QualityMonitor

Prior to deploying a Windows Installer–based application, typically you need to test it in the targeted deployment environment to ensure the application works as expected. However, it is often not feasible (or possible) to test each piece of an application's functionality, due to the complexity of the application and/or its interface. Behind the scenes, there may be dozens or hundreds of attempts to access files, registry keys, or services; errors may only become apparent in rare and isolated circumstances.

One major source of failure is when the target environment is restricted in some way, such as in a locked-down environment. In this case, there may be prohibitions on certain COM activation or registry access, which ultimately prevents an application from working correctly.

QualityMonitor allows you to run a series of built-in tests to installed Windows Installer-based products, helping to ensure they run correctly. When failures occur, QualityMonitor can help identify where problems exist, and ultimately direct you to the solution.

Creating New QualityMonitor Project Files

You can create a new QualityMonitor project by selecting the **Create new project** option on the **Welcome to QualityMonitor View**.

Task ***To create a new QualityMonitor project file (.iqm):***

1. Launch QualityMonitor. The **Welcome to QualityMonitor View** opens.
2. Click **Open** on the **File** menu. The **Open QualityMonitor Project** dialog box opens.
3. Select the **Select an application that is installed on this machine ...** option.
4. Select an application from the available applications list.
5. Click **OK**. A new QualityMonitor project for the selected application is opened, and the **Product Information View** opens.

Opening Existing QualityMonitor Project Files

QualityMonitor's project files have an `.iqm` extension and are opened by performing the following steps.

Task ***To open an existing QualityMonitor project file (.iqm):***

1. Launch QualityMonitor. The **Welcome to QualityMonitor View** opens.
2. Click **Open** on the **File** menu. The **Open QualityMonitor Project** dialog box opens.
3. Select the **Open QualityMonitor project (.iqm) file** option.
4. Enter or browse to the file you want to open.
5. Click **OK**.

Working with Test Cases

The primary purpose of QualityMonitor is to serve as a diagnostic tool when applications fail to function correctly in deployment environments. This is accomplished primarily through running Test Cases and individual Test Items, and evaluating the results. Topics in this section cover the most common tasks you may perform when working with Test Cases.

- Running Individual Test Items
- Running Multiple Test Items
- Adding Test Item Comments
- Adding Test Case Comments
- Viewing Test Item Details
- Clearing Test Case Results
- Manually Setting Test Case Status
- Manually Setting Test Item Status
- Filtering Test Case Data

Running Individual Test Items

You can choose to run a **Deployment Test** on an individual **Test Item**.

Task ***To run an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:

- Class IDs
- File Associations
- Help Files
- Prog IDs
- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. In the **Test Items** list, select the **Test Item** you want to run and click the Run button.

   Depending on whether the **Test Item is** automatic or requires opening or launching files, you may need to perform some manual tasks prior to results being returned. When the test is complete, QualityMonitor displays **Passed** or **Failed** in the **Status** column of the selected **Test Item**.

Running Multiple Test Items

You can choose to run a **Deployment Test** on a multiple **Test Items** at once.

Task ***To run multiple Test Items:***

1. Create or open a QualityMonitor project. The **QualityMonitor Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Test Items** list, select the **Test Items** you want to run. Multiple selection is supported using the Shift and Ctrl keys.
4. To run only the selected **Test Items**, click the **Run** button. To run all **Test Items**, click the **Run All** button.

Depending on whether the **Test Item** is automatic or requires opening or launching files, you may need to perform some manual tasks prior to results being returned. When the tests are complete, QualityMonitor displays **Passed** or **Failed** in the **Status** column of the selected **Test Items**.

Adding Test Item Comments

You can choose to add comments to a **Test Item**—perhaps to document why it passed or failed, or to note an issue that needs attention.

Task ***To add comments to an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Test Items** list, right-click the **Test Item** to which you want to add comments, and select **Test Item Information** from the shortcut menu.The **Test Item Information** dialog box appears.
4. Enter comments into the **Comments** field.
5. When finished, click **OK**.

Adding Test Case Comments

You can add comments associated with an entire Test Case.

Task ***To add comments to a Test Case:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs

- File Associations
- Help Files
- Prog IDs
- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. In the **Comments** box at the top right of the View, enter comments. Your comments are automatically saved.

Viewing Test Item Details

When a Test Item fails, you can view details about it, including the error message associated with it. This information is displayed on the **Test Item Information** dialog box, along with the Test Item name, status, and any comments that have been entered.

Task ***To view Test Case details:***

1. Create or open a QualityMonitor project. The **QualityMonitor Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. Right-click on a **Test Item** and select **Test Item Information** from the shortcut menu. The **Test Item Information** dialog box opens, and the following details are listed:
   - **Test Item**—Name of the selected Test Item.
   - **Status**—Status of the selected Test Item: Passed, Failed, or Pending.
   - **Comments**—Any comments that were previously entered.
   - **Test Details**—If this Test Item has **Failed**, a brief explanation of the reason the Test Item failed the test is listed.

4. When finished viewing test details, click **OK** to close the dialog box.

Clearing Test Case Results

You can clear Test Case results for all Test Items from a previous Test Case execution.

Task ***To clear Test Case results (all Test Items) from a previous Test Case execution:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. Click the **Reset Results** button.

When you click **Reset Results**, the status of *all* Test Items is reset. To reset the status of one individual **Test Item**, right-click that **Test Item**, point to **Set Status** and select **Pending** from the shortcut menu. See Manually Setting Test Case Status.

Manually Setting Test Case Status

Depending on your business practices and standards, you may want to override the status of a Test Case in the View List from its current state. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.

Task ***To manually set the Test Case status:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files

- Prog IDs
- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. In the **Test Case Status** area, change the status to the desired state by selecting **Pending**, **Passed**, or **Failed**.

Manually Setting Test Item Status

You can manually set the status of an individual Test Item to Passed, Failed, or Pending.

Task ***To manually set the status of an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. Right-click on the **Test Item**, point to **Set Status,** and select the status from the Set Status submenu: **All**, **Pending**, **Passed**, or **Failed**.

Filtering Test Case Data

On the **Product Information View**, you can choose to display only those Test Cases with a selected status: **Pending**, **Passed**, or **Failed**.

Task ***To filter the displayed data in a Test Case:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. From the **View these test items** list, select the filter you want to apply to the data: **All**, **Passed**, **Failed**, or **Pending**. The View is automatically updated based on the selected filter.

Deployment Testing

Deployment tests are performed against the installed product, ensuring that the product has been installed correctly and all key functionality works in the installed environment. Test Cases in this area are primarily designed to identify whether the application fails to work properly due to permission settings on the registry or individual files.

Some of the primary areas checked are:

Table 20-2 • Areas Checked During Deployment Testing

Area	Description
COM Data	Ensure all COM objects can be instantiated programatically. This includes Class IDs, Prog IDs, and Type Libraries. COM data is tested silently, returning results in the Test Case Progress area and the queue. See Checking Class IDs. Checking Prog IDs, or Checking Type Libraries.
File Associations	Ensure all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used. See Checking File Associations.
Help Files	Ensure help files are installed and can be launched correctly. See Checking Help Files.
Shortcuts	Ensure each shortcut is installed and if it successfully launches the shortcut target. See Checking Shortcuts.

Table 20-2 • Areas Checked During Deployment Testing (cont.)

Area	Description
Type Libraries	Determines if the Type Libraries COM objects can be instantiated programatically. See Checking Type Libraries.
Manifests	Tests the manifests and assemblies used to isolate a Windows Installer package. See Checking Manifests.
ODBC Data Sources	Verify ODBC data sources. See Checking ODBC Data Sources.
ODBC Drivers	Verify ODBC drivers. See Checking ODBC Drivers
Services	Ensure all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine. See Checking Services.

Automatically Running All Deployment Tests Silently

You can choose to run all deployment tests silently (without prompting for user input) using either the Interface or the command line.

From the Interface

You can choose to run all deployment tests silently (without prompting for user input) by making a selection in the QualityMonitor interface.

Task ***To run all deployment tests silently from the Interface, do one of the following:***

1. On the QualityMonitor **Product Information View**, select the **Deployment Tests** root node and then do one of the following:
   - Click the **Execute All Deployment Tests** button.
   - Select A**ll Deployment Tests** from the **Execute** menu.
   - Click the **Execute All Deployment Tests** toolbar button:

   When you select one of these options, a dialog box with a progress bar and an option to cancel will be displayed.

From the Command Line

You can also run all deployment tests silently by entering a command in the command line. See Running QualityMonitor from the Command Line for more information.

Checking Class IDs

The **Class ID** Deployment Test is run to determine if the Class ID COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task ***To check Class ID functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Class IDs**. The **Class IDs View** opens.
3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are listed in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box, right-click on the Test Item and then select* ***Test Item Information*** *from the shortcut menu.*

Checking File Associations

The **File Associations** Deployment Test is run to determine if all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used.

Task ***To test file associations:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **File Associations** from the View List. The **File Associations View** opens.
3. Right-click on the Test Item you want to run and select **Run**.You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to exercise the file association.
5. From the resulting **Open** dialog box, browse to a file with the appropriate extension and click **Open**. The file is launched with its associated application. Following the application launch, the **Test Result** dialog box appears.
6. Click **Yes** or **No** depending on whether the file launched with the expected program. You can also enter comments in the **Comment** field on this dialog box.

Checking Help Files

The **Help Files** Deployment Test is run to determine if the help files are installed and can be launched correctly.

Task ***To test help file functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Help Files** from the View List. The **Help Files View** opens.

3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to launch the help file. Following an attempt to launch the shortcut, the **Test Result** dialog box appears.
5. Click **Yes** or **No** depending on whether the help file launched correctly. You can also enter comments in the **Comment** field on this dialog box.

Checking Prog IDs

The **Prog IDs** Deployment Test is run to determine if the Prog ID COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task ***To check Prog ID functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Prog IDs** from the View List. The **Prog IDs View** opens.
3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box, right-click on the Test Item and then select **Test Item Information** from the shortcut menu.*

Checking Services

The **Services** Deployment Test is run to determine if all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine.

Task ***To check Service functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Services** from the View List. The **Services View** opens.
3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

***Note** • When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box, right-click on the Test Item and then select **Test Item Information** from the shortcut menu.*

Checking Shortcuts

The **Shortcuts** Deployment Test is run to determine if each shortcut is installed and if it successfully launches the shortcut target.

Task ***To check shortcuts:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Shortcuts** from the View List. The **Shortcuts View** opens.
3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to launch the shortcut. The **Test Result** dialog box opens.
5. Following an attempt to launch the shortcut, the **Test Result** dialog box opens. Click **Yes** or **No** depending on whether the shortcut launched correctly. You can also enter comments in the **Comment** field on this dialog box.

Checking Type Libraries

The **Type Libraries** Deployment Test is run to determine if the Type Libraries COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task ***To check type library functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Type Libraries** from the View List. The **Type Libraries View** opens.
3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

***Note** • When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box, right-click on the Test Item and then select **Test Item Information** from the shortcut menu.*

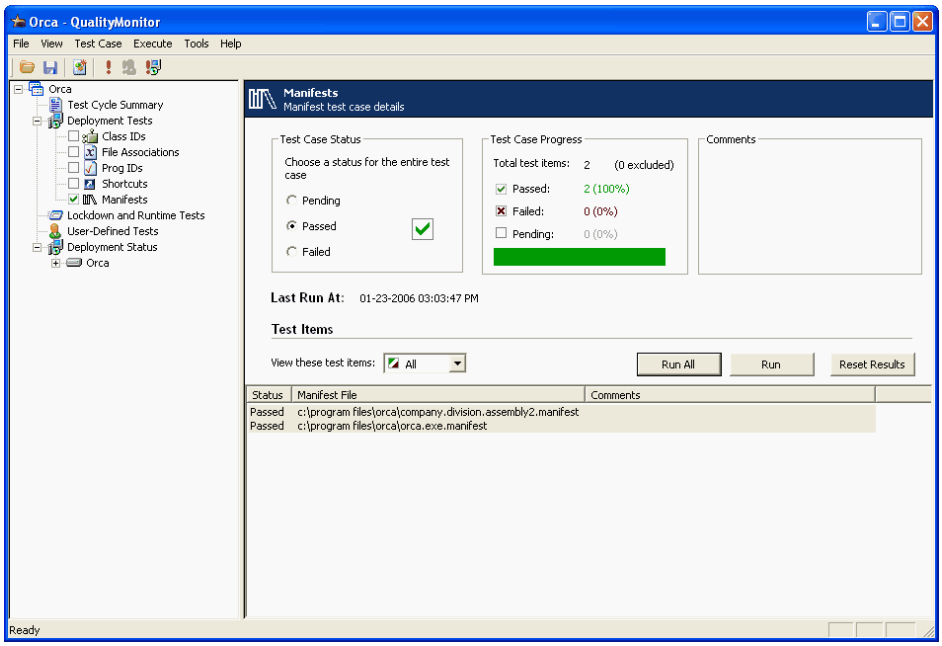
Checking Manifests

The **Manifests** Deployment Test is run to test the manifests and assemblies used to isolate a Windows Installer package.

The Manifests Deployment Test tests information from the `MsiAssembly` and `MsiAssemblyName` tables. QualityMonitor reads through the manifest/assembly files and performs the baseline Class IDs, Prog IDs, or Type Libraries testing for each entry in the files.

Task ***To check shortcuts:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Manifests** from the View List. The **Manifests View** opens.



3. Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.

   When testing is finished, results are recorded in the **Test Case Progress** area. Also, the **Status** of each test item (**Passed**, **Failed**, or **Pending**) is listed next to the **Manifest File** name.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box. right-click on the Test Item and then select* ***Test Item Information*** *from the shortcut menu.*

4. If desired, you can also enter comments in the **Comment** field on this dialog box.

Checking ODBC Data Sources

The **ODBC Data Sources** Deployment Test is run to verify ODBC data sources.

Task ***To check ODBC data sources:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **ODBC Data Sources** from the View List. A list of data sources in that application appears in the lower portion of the **ODBC Data Sources View**. Only those data sources that belong to the current logged-in user are listed on the **ODBC Data Sources View**.
3. Select the Test Item you want to run and click **Run**. You can also select multiple Test Items to run, or click **Run All** to run all available Test Items.

   For certain ODBC data sources, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box, click on the Test Item and then select* ***Test Item Information*** *from the shortcut menu.*

Checking ODBC Drivers

The **ODBC Drivers** Deployment Test is run to verify ODBC Drivers.

Task ***To check ODBC drivers:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **ODBC Drivers** from the View List. A list of drivers in that application appears in the lower portion of the **ODBC Drivers View**. Only those drivers that belong to the current logged-in user are listed.
3. Select the Test Item you want to run and click **Run**. You can also select multiple Test Items to run, or click Run All to run all available Test Items.

   For certain ODBC drivers, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box. right-click on the Test Item and then select* ***Test Item Information*** *from the shortcut menu.*

Specifying Exclusions for Deployment Testing

When a deployment test is run on a package, some of the tests related to Class IDs, Prog IDs, and Type Library IDs fail because they refer to components which belong to the operating system rather than the software which is being tested. These errors have no impact on the integrity of the software being tested, and cause confusion among some users testing the software. Users need to be able to prevent error messages caused by files that are not affecting the performance of the software package to be listed in the test results.

To prevent these operating systems errors from being reported, you can specify a list of files to be excluded when any of the Class ID, Prog ID, or Type Library ID Deployment tests are run. You can maintain a different list for each of these three Deployment Tests.

Note • *After a Deployment Test has been run, the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View. The items included in the exclusion lists are not shown in these views, but are still stored in the QualityMonitor Project File (`.iqm`). When this project file is opened again in QualityMonitor, the results are checked against the exclusion list before being displayed in the Class IDs View, Prog IDs View, or Type Libraries View.*

To Add a File to the Exclusion List

On the **Exclusions** tab of the Options Dialog Box, you can manage all three exclusion lists. On the **Exclusions** tab, you can view the exclusion lists, and can add or remove entries from a list.

Using the Options Dialog Box

Task ***To exclude Class IDs, Prog IDs, or Type Library IDs using the Options dialog box:***

1. From the QualityMonitor interface, select **Options** on the **Tools** menu. The **Options** dialog box opens.
2. Click the **Exclusions** tab. On the **Exclusions** tab, excluded items are listed for the selected **Exclusions list**.
3. From the **Exclusions list**, select the Deployment Test that you want to modify the exclusion list for: **Class ID**, **Prog ID**, or **Type Library**.
4. Click **Add**. The **Add Exclusions** dialog box opens.
5. Next to the **File Name** box, click **Browse** and select the **Application (.exe)**, **Application Extension (.dll)**, **Type Library (.tlb)**, or **ActiveX object (.oxc)** file that contains Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the selected Deployment Test.

   The Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file are listed, displaying the **Identifier** and a **Description** of each.
6. Select the Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the Deployment Test and click **OK**.

Directly From the Results Window

After a Deployment Test has been run and the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View, you can add an item to the exclusion list directly by right-clicking on the item you want to exclude and choosing **Add to exclusion** from the shortcut menu.

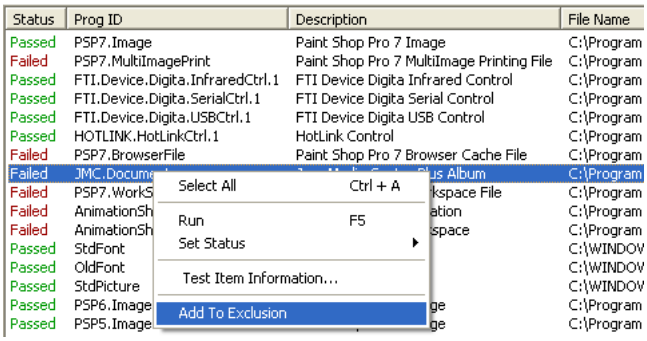


Figure 20-1: Adding a Prog ID to the Prog ID Exclusion List from the Test Results List

Deleting a Item from an Exclusion List

To delete an item from an exclusion list, perform the following steps.

Task ***To delete a Class IDs, Prog IDs, or Type Library IDs from an exclusion list:***

1. From the QualityMonitor interface, select **Options** on the **Tools** menu. The **Options** dialog box opens.
2. Click the **Exclusions** tab. On the **Exclusions** tab, excluded items are listed for the selected **Exclusions list**.
3. From the **Exclusions list**, select the Deployment Test that you want to modify the exclusion list for: **Class ID**, **Prog ID**, or **Type Library**.
4. Select the item that you want to delete from the list and click **Remove**. The item is removed from the list.

Selecting the Default Exclusion List

On the General Tab of the **Options** dialog box, you can select the default **Exclusion file** to use to filter the test results in the **Lockdown and Runtime Tests** views. By selecting an exclusion file from a shared location, multiple people can use the same error exclusion settings.

Lockdown and Runtime Testing

Lockdown and runtime tests are available through the Lockdown and Runtime Tests View. You are provided with a list of available shortcuts in the package and all of the executables in the package. You can then launch via the shortcut or executable, and exercise functionality in the application. When you close the application, information about the executable is listed under the **Runtime Checks** node. This information, grouped into **Files**, **Registry Entries**, and **Folders** views, allows you to see failures in the application execution. These are potential issues with the application, and may or may not have any affect on the overall package integrity.

If you want to execute tests in the context of a different user (under a different user account), click **Run As** instead of **Run** to execute the test. You would then be prompted to enter a **User Name** and **Password**. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

This section includes the following topics:

- Performing Lockdown and Runtime Tests
- Performing Lockdown and Runtime Tests Under a Different User Account

- Running Lockdown and Runtime Tests in Restricted Environments
- Performing Isolation Tests
- Filtering Results of Lockdown and Runtime Tests

Note • *Lockdown and runtime tests are only supported on Windows NT4, 2000, and XP.*

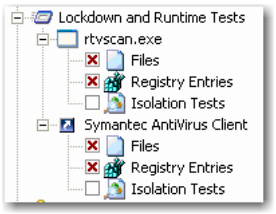
Performing Lockdown and Runtime Tests

Lockdown and Runtime Tests allow you to see failures in the application execution. These are potential issues with the application, and may or may not have any affect on the overall package integrity.

Task ***To perform lockdown and runtime tests:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run**.
6. When the application launches, use the application in a normal way, performing various operations.
7. Exit the application.

   The name of the executable or shortcut is now listed as a new node under the **Lockdown and Runtime Tests** node in the View List, and a subnode is listed for any access failures for **Files**, **Folders**, or **Registry Entries**.



8. Select the **Files**, **Folders**, and **Registry Entries** nodes. The right side of the **Lockdown and Runtime Tests View** displays a list of failed **Test Items** for each node. For information on specifying which errors are listed, see Filtering Results of Lockdown and Runtime Tests.
9. To view the error message for a **Test Item**, right-click the Test Item and select **Test Item Information** on the shortcut menu. The Test Item Information dialog box opens, listing an **Error Description** in the **Test Details** area.

Performing Lockdown and Runtime Tests Under a Different User Account

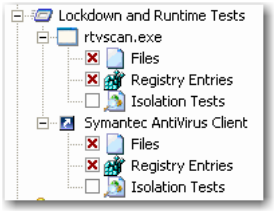
You can use the **Run As** feature to execute Lockdown and Runtime tests in the context of a different user. This allows you to validate an application in a locked-down environment without actually requiring a user to log-in with a different set of credentials. This will reduce the test cycle effort significantly.

Task ***To perform tests under a different user account:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run As**.

   You are then prompted to enter a **User Name** and **Password**.
6. Enter the User Name in the format of: `DOMAINNAME\UserName`. The default value is the current **User Name**.
7. When the application launches, use the application in a normal way, performing various operations.
8. Exit the application.

   The name of the executable or shortcut is now listed as a new node under the **Lockdown and Runtime Tests** node in the View List, and a subnode is listed for any access failures for **Files**, **Folders**, or **Registry Entries**.

9. Select the **Files**, **Folders**, and **Registry Entries** nodes. The right side of the **Lockdown and Runtime Tests View** displays a list of failed **Test Items** for each node. For information on specifying which errors are listed, see Filtering Results of Lockdown and Runtime Tests.
10. To view the error message for a **Test Item**, right-click the Test Item and select **Test Item Information** on the shortcut menu. The Test Item Information dialog box opens, listing an **Error Description** in the **Test Details** area.

Note • ***Run As*** *can also be selected using the Shift+F5 shortcut, or by selecting* ***Run As*** *from the* ***Execute*** *menu.*

Running Lockdown and Runtime Tests in Restricted Environments

When executing runtime tests in a locked-down environment, you may encounter an error such as:

```
Unable to monitor the application execution.
```

QualityMonitor needs to run under a user with Admin privileges when executing Lockdown and Runtime Tests.

To emulate lockdown environment under a restricted user, click **Run As** instead of **Run** to execute the test, and provide the **User Name** and **Password** of a locked down user. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

Performing Isolation Tests

You can run Isolation Tests to display the location of all portable executable (PE) files (`dll/ocx/exe/tlb/olb`) that are launched from a process while performing a Lockdown and Runtime test. Viewing a listing of these portable executable file names and paths makes it easier for you to ensure that the application is fully isolated.

After you perform a Lockdown and Runtime Test for an executable (`.exe`) or a shortcut on the Lockdown and Runtime Tests View, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.

When you select this **Isolation Tests** node, the filenames of the portable executable files and their paths are listed. By default, the status of all these items is **Pending**. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

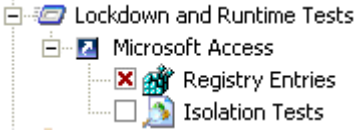
Note • *Note the following regarding Isolation testing:*

- *The Isolation Tests node will be added to the Lockdown and Runtime Tests tree only if the selected executable launches at least one portable executable file.*
- *You can specify when you would like the **Isolation Tests** node to appear in the Lockdown and Runtime Tests tree by selecting an option from the **Show Isolation Tests** list on the **General** tab of the Options Dialog Box.*
- *QualityMonitor does not support isolation testing under Windows 2000.*

Task ***To perform isolation tests:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run**.

   After a Lockdown/Runtime test is performed for an executable (`.exe`) or a shortcut, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.



6. Select the **Isolation Tests** node. The filenames of the portable executable files and their paths is displayed in the list control of the view. By default, the status of all of the items in this test case is **Pending**.

7. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

Filtering Results of Lockdown and Runtime Tests

On the Lockdown and Runtime Tests Files View, Folders View, Registry Entries View, and Isolation Tests View you can choose to filter the results that are listed.

Task ***To filter test results:***

1. To filter the list by Test Item status, select an option from the **View these test items** list: **Passed**, **Failed**, **Pending**, or **All**.
2. To select errors to exclude from future Lockdown and Runtime result listings, perform the following steps:
   a. Click the **Set Filter** button. The Runtime Test Filters Dialog Box opens, listing all errors that were generated during this test.
   b. Select those errors that you want to exclude from future Lockdown and Runtime tests.

      These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
3. To filter the list by one type of error that was generated, make a selection from the **Having these errors** list. This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). To see all the errors, select **Show All**.

Using MSI Doctor to Verify Package Deployment Status

You can use QualityMonitor's MSI Doctor to verify if an MSI package is installed properly. This helps prevent users from seeing an auto-repair dialog box when they run the application. Auto repair messages are displayed by applications to attempt to reinstall missing/corrupted components.

By examining an application using QualityMonitor MSI Doctor, you can quickly identify any problems by checking the status of all products and features. Using MSI Doctor, you can:

- See the status of all components
- Verify if any files are missing or if any files do not match the version or size specified in the MSI file
- See the components status segregated by features
- Configure or reinstall features
- Reinstall components

To use MSI Doctor, select the **Deployment Status** node ( ) from the QualityMonitor View List to access the Deployment Status View. Under the **Deployment Status** node, a tree view of the application's features are components are listed:

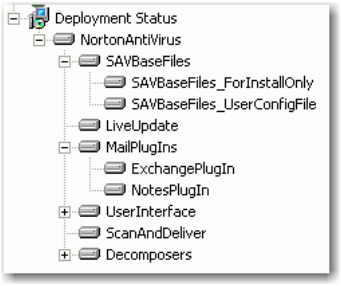


Figure 20-2: Deployment Status View: Features and Components

The following icons are displayed in the tree view and in the component list and indicate the feature or component's status:

Table 20-3 • Deployment Status View Icons

Icon	Name	Description
	Installed	Feature or Component is installed on the local machine.
	Uninstalled	Feature or Component is not installed on the local machine.
	Broken	Component is broken (a key file in the Component is missing) or Feature contains a broken Component.
	Run From Source	Feature or Component is configured to run from a source location (rather than being installed on the local machine).
	On Demand	Feature is configured to be installed when needed. Not applicable to Components.

Shortcut Menu Functionality

The following table lists the functions available on the shortcut menus for the **Deployment Status** icon, **Features**, and **Components**, and the dialog boxes that appear when those functions are selected:

Table 20-4 • Dialog Boxes Invoked from Deployment Status View Shortcut Menu

Shortcut Menu	Deployment Status Node	Feature Node	Component Node
Configure	Install or Configure Product Dialog Box	Install or Configure Feature Dialog Box	N/A
Re-install	Re-install Product/Feature Dialog Box	Re-install Product/Feature Dialog Box	Installation program is launched. (This option is only enabled if the selected Component is broken.)
Properties	Product Properties Dialog Box	Feature Properties Dialog Box	Component Properties Dialog Box

Using MSI Doctor, you can perform the following tasks:

- View Product, Feature, or Component Deployment Status Properties
- Verify Product, Feature, or Component Data
- Install or Configure Products or Features
- Reinstall Features

View Product, Feature, or Component Deployment Status Properties

To view the deployment status properties of a product, feature, or component, perform the following steps.

Task ***To view product, feature, or component Deployment Status properties:***

1. Launch QualityMonitor and open the package that you want to view the deployment status properties of. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To view Product properties**—Select the **Deployment Status** icon from the View List.
   - **To view Feature properties**—Select a **Feature** icon under the **Deployment Status** icon on the View List.
   - **To view Component properties**—Select either the **Deployment Status** icon or a **Feature** icon from the View list, and then select a **Component** from the list on the right.
3. Select **Properties** from the shortcut menu. The **Product Properties**, **Feature Properties**, or **Component Properties** dialog box appears, displaying property information for the selected Product, Feature, or Component.
4. On the Properties dialog boxes, you can also click the **Verify Data** button to verify if the files and registry information for the selected item are installed properly. See Verify Product, Feature, or Component Data for more information.
5. Click **OK** to exit the **Properties** dialog box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Verify Product, Feature, or Component Data

You can verify if the files and registry information for a Product, Feature, or Component are installed properly. Verification errors are displayed on the Installed Data Dialog Box.

Task ***To verify product, feature, or component data:***

1. Launch QualityMonitor and open the package that you want to verify the files of. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To verify all of the files in all of the Features in the Product**, select **Deployment Status** from the View List.
   - **To verify only the files in the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.
   - **To verify only the files in a specific Component of a Feature**, select either the **Deployment Status** icon in the View List or a **Feature** icon under it, and then select a component from the list on the right.
3. Select **Properties** from the shortcut menu. The **Product Properties**, **Feature Properties**, or **Component Properties** dialog box appears, displaying property information for the Product, Feature or Component.
4. Click **Verify Data**. When you click **Verify Data**, QualityMonitor checks all of the files and registry entries included in the Product, Feature, or Component and then displays them on the **Files** and **Registry** tabs of the **Installed Data** dialog box.
   - On the **Files** tab, the following icons are used to identify verification errors:

Icon	Description
	File is missing.
	File has a different version or size than that specified in the Windows Installer package

   - On the **Registry** tab, all registry entries for the selected item are listed. Registry data is verified by checking the existence of the registry key and the value name (if one exists). (The value data is not checked.). The icon is used to indicate that a registry key or value name is incorrect or missing:
5. Click **Close** to exit the **Installed Data** dialog box, and click **OK** to exit the **Properties** dialog box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Install or Configure Products or Features

You can use **Configure** to install a product or feature that is not currently installed.

Task ***To install or configure products or features:***

1. Launch QualityMonitor and open the package that you want to install or configure. The QualityMonitor **Product Information View** opens.
2. Select one of the following:

- **To install or configure the entire Product**, select **Deployment Status** from the View List.
- **To install or configure only the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.

3. Select **Configure** from the shortcut menu. The **Install or Configure Product** or **Install or Configure Feature** dialog box appears, prompting you to select the installation location and the installation type (on the **Install or Configure Product** dialog box only).
4. Select one of the following options to specify installation location:
   - **Default**—Files will be installed to their default location.
   - **Local**—Files will be installed on the local machine.
   - **Source**—Files will be run from the installation source.
   - **On Demand**—Files will be installed when needed.
5. (Product only) Select one of the following options to specify installation type:
   - **Minimum**—Only the essential features will be installed.
   - **Typical**—Most commonly used features will be installed.
   - **Complete**—All of the program's features will be installed.
6. Click **OK**. The Product or Feature is installed, per the options you specified.

Reinstall Features

When a Feature is broken (identified by the icon), you can fix it by re-installing the entire Product or just re-installing the broken Feature.

Task ***To reinstall features:***

1. Launch QualityMonitor and open the package that you want to reinstall. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To reinstall the entire Product**, select **Deployment Status** from the View List.
   - **To reinstall only the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.
3. Select **Re-install** from the shortcut menu. The **Re-install Product/Feature** dialog box appears, prompting you to select a reinstall mode.
4. Select one of the following reinstall modes:
   - Repair all detected reinstall problems
   - Reinstall only if file is missing
   - Force all files to be reinstalled
   - Reinstall if file is missing, or an older version exists

- Reinstall if file is missing, or an older or equal version exists
- Reinstall if existing file has different version
- Verify that required user registry entries are present
- Verify that required local machine registry entries are present
- Recreate all shortcuts

5. Click **OK**. The Product or Feature is re-installed, per the option you specified.

Reinstall Components

When a Component is broken (identified by the icon), you can fix it by re-installing the broken Component.

Task ***To reinstall components:***

1. Launch QualityMonitor and open a package. The QualityMonitor **Product Information View** opens.
2. Select **Deployment Status** from the View List or select a **Feature** icon under **Deployment Status** on the View List. All components associated with the selected feature(s) are listed.
3. Right-click on the component that you want to re-install and select **Re-install** from the shortcut menu. The component is automatically reinstalled.

Creating Custom Test Cases

QualityMonitor supports adding additional, custom Test Cases to projects—based on your business needs.

You can define a template that includes information that is re-used when defining Test Cases. Then, when creating a new user-defined Test Case, you can load this template file and have the components of the new Test Case pre-populated with the information saved in the template. A single template file can load multiple user defined Test Cases.

Adding User-Defined Test Cases

Task ***To add a user-defined case:***

1. Launch QualityMonitor and open the package that you want to create a custom Test Case for. The QualityMonitor **Product Information View** opens.
2. Right-click on the **User-Defined Tests** node and select **Add Test Case** from the shortcut menu.

   A new Test Case appears below the **User Defined Tests** node, and you are prompted to enter a name.
3. Name the Test Case appropriately.
4. Select the new Test Case to open the **Test Case View.**
5. Under **Test Case Status**, specify the status for this Test Case: **Pending**, **Passed**, or **Failed**.
6. Click **Browse** and select an executable to associate with this Test Case, if necessary.

7. In the **Instructions** text box, enter any necessary comments.

   For example, you may want to create a custom Test Case to ensure that a specific database is updated properly after running an application.

8. Your entries are automatically saved in the new Test Case.

Creating and Using Test Case Templates

Task ***To create and use Test Case templates:***

1. Create a new Test Case as described in Adding User-Defined Test Cases.
2. To save the Test Case as a template to re-use when creating new Test Cases, perform the following steps:
   a. Right-click on this Test Case node in the **User Defined Tests** tree and select **Save as template** from the shortcut menu.
   b. Specify a name and a location for the Test Case template and select **Save**. The template is saved in `.xml` format in the location you specify.
3. To add this Test Case to an existing template, perform the following steps:
   a. Right-click on this Test Case node in the **User Defined Tests** tree and select **Add to template** from the shortcut menu. You are prompted to select the template file that you want to add this Test Case to.
   b. Select the template that you want to add this Test Case to and select **Save**.
4. To create a new Test Case based upon a template, perform the following steps:
   a. Right-click on the **User Defined Tests** node and select **Load template** from the shortcut menu. You are prompted to select the template that you want to use.
   b. Select a template and click **Open**. All of the Test Cases that were saved in the template are now listed in the **User Defined Tests** tree.

You can specify that you want a specific template file automatically loaded each time a QualityMonitor project is opened. To do this, select a **Template file** on the General Tab of the **Options** dialog box, and also select the **Load Templates on Project Open** option.

Renaming User-Defined Test Cases

Task ***To rename a user-defined Test Case:***

1. Under the **User-Defined Tests** node, right-click on the Test Case that you want to rename, and select **Rename** from the shortcut menu.
2. Provide a new name for the Test Case.

Test Reports

QualityMonitor allows you to create an HTML test report for the current project. This can be done by selecting **Generate Report** from the **File** menu and providing the name and location for the report. The report will then automatically open in your default browser.

Note • *If an error message occurs when generating the report, it may be because msxml4.dll or isqm.xlst is not in the same directory as the QualityMonitor executable (isqm.exe). These files must be present to create the report.*

Running QualityMonitor from the Command Line

QualityMonitor can be run from the command line by using `isqm.exe`. It can accept the following parameters:

Table 20-5 • QualityMonitor Command Line Parameters

Syntax	Options	Description

-c <MSI Product Code>		Launch QualityMonitor by opening the MSI product specified by the product code. For example:
		`isqm.exe -c {BDC62375-07B4-4CBD-9991-4C25C24F3071}`
		`isqm.exe -c {BDC62375-07B4-4CBD-9991-4C25C24F3071} -sb -f <c:\projectfile.iqm>`

	-sn	Run QualityMonitor silently without any user interaction and no progress display.
	-sb	Run QualityMonitor silently with a progress display. QualityMonitor displays the test names as they are executed and provides an option for the user to cancel.
	-r <Report File>	Generates a report file `<c:\report.htm>` with the test results. Works only when using either `-sn` or `-sb`.
	-f <Project File>	Save test results in this file. This is necessary when using either `-sn` or `-sb`. If this file does not exist, it will be created and then results will be saved.

Table 20-5 • QualityMonitor Command Line Parameters (cont.)

Syntax	Options	Description

-f <Project File>		Launch QualityMonitor with this Project File. For example:
		`isqm.exe -f c:\mydocuments\mytesting\tesresults.iqm`
		`isqm.exe -f c:\mydocuments\mytesting\tesresults.iqm -sn`

	-sn	Run QualityMonitor silently without any user interaction and no progress display.
	-sb	Run QualityMonitor silently with a progress display. QualityMonitor displays the test names as they are executed and provides an option for the user to cancel.
	-r <Report File>	Generates a report file <`c:\report.htm`> with the test results. Works only when using either `-sn` or `-sb`.
-h or -?		Help

Note *• When using any -sn or -sb command line options, you can specify the target product using the existing options /c or /f.*

- *If you use the /f option to specify the product, the input file will be modified with the test results.*
- *If you use /c to specify the target product, /f options must be used to specify the project file path which will have the test results.*
- *If both /c and /f parameters are specified, then QualityMonitor gives preference to /c and operates with the product code specified by /c.*

QualityMonitor Reference

Topics contained in this section provide detailed reference on each user interface element, dialog box, or view in QualityMonitor. This is the same documentation displayed when you click F1 from the QualityMonitor interface. Topics are organized as follows:

- Menus and Toolbar
- QualityMonitor Interface
- Dialog Boxes
- Views

Menus and Toolbar

The following table provides a description of each of QualityMonitor's menu commands and toolbar buttons:

Table 20-6 • QualityMonitor Menus and Toolbar Options

Menu	Command	Toolbar Button	Keyboard Shortcut	Description
File	**Open**		Ctrl+O	Allows you to open an existing QualityMonitor project file (.iqm) or create a new one based on an installed MSI-based application.
File	**Close**			Closes the current project.
File	**Save**		Ctrl+S	Saves the current project.
File	**Save As**			Saves the current project using the name and location you specify.
File	**Generate Report**			Creates an HTML test report for the current project. The report will automatically open in your default browser.
File	**1,2,3,4**			Allows you to open the four most recently accessed QualityMonitor projects.
File	**Exit**			Exits QualityMonitor.
View	**Toolbar**			Toggles display of the toolbar.
View	**Status Bar**			Toggles display of the status bar.
Test Case	**Add Test Case**			Adds a custom Test Case beneath the Additional Tests view in the View List.
Execute	**All Deployment Tests**		Ctrl+F5 or Alt+E+D	Runs all the deployment tests in the current project.
Execute	**Run**		F5 or Alt+E+R	Runs the selected deployment, lockdown and runtime, or user defined test.
Execute	**Run As**		Shift+F5	Runs the selected lockdown and runtime test in the context of a different user. You are then prompted to enter a User Name and Password.
Tools	**Options**			Displays the Options dialog box.
Help	**Contents**			Launches the Help Library, displaying the Contents tab.

Table 20-6 • QualityMonitor Menus and Toolbar Options (cont.)

Menu	Command	Toolbar Button	Keyboard Shortcut	Description
Help	**Index**			Launches the Help Library, displaying the Index tab.
Help	**Search**			Launches the Help Library, displaying the Search tab.
Help	**Support Central**			Accesses AdminStudio Support Central on the Web.
Help	**Web Community**			Accesses the AdminStudio Community on the InstallShield website.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.
Help	**Feedback**			Accesses the AdminStudio feedback form on the InstallShield website.
Help	**AdminStudio on the Web**			Accesses the AdminStudio website.
Help	**About QualityMonitor**			Displays the About dialog box.

QualityMonitor Interface

The QualityMonitor interface is divided into two main areas. The View List, which appears at the left side of the screen, provides a visual representation of the QualityMonitor project file. It provides easy access to individual views and Test Cases. The right side of the interface changes depending on the view or Test Case selected.

Dialog Boxes

The following dialog boxes are accessible from within QualityMonitor:

- About QualityMonitor Dialog Box
- Add Exclusions Dialog Box
- Component Properties Dialog Box
- Feature Properties Dialog Box
- Install or Configure Feature Dialog Box
- Install or Configure Product Dialog Box
- Installed Data Dialog Box
- Open QualityMonitor Project Dialog Box
- Options Dialog Box

- Product Properties Dialog Box
- Re-install Product/Feature Dialog Box
- Test Item Information Dialog Box
- Test Progress Dialog Box
- Test Result Dialog Box

About QualityMonitor Dialog Box

The About InstallShield QualityMonitor dialog box, available by selecting About from the Help menu, displays information about QualityMonitor and AdminStudio, including version and activation code information.

Add Exclusions Dialog Box

On the Add Exclusions dialog box, which is opened by clicking **Add** on the Exclusions Tab of the Options dialog box, you can select a Class ID, Prog ID, or Type Library ID to exclude from a Deployment Test. See Specifying Exclusions for Deployment Testing.

The following options are included:

Table 20-7 • Add Exclusions Dialog Box Options

Options	Description
File Name	Click Browse and select the **Application (.exe)**, **Application Extension (.dll)**, **Type Library (.tlb)**, or **ActiveX object (.oxc)** file that contains Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the selected Deployment Test.
Identifier	Identifies the Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file.
Description	Description of the Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file.

Component Properties Dialog Box

The Component Properties dialog box is displayed when you right-click on a Component in the Component list on the right side of the Deployment Status View and select Properties from the shortcut menu.

The following information is listed:

Table 20-8 • Component Properties Dialog Box Options

Options	Description
Name	Name of the selected Component.
GUID	Number which uniquely identifies this Component.
Status	Status of Component, such as Installed on Local Drive.

Table 20-8 • Component Properties Dialog Box Options (cont.)

Options	Description
Location	Location on server where Component is installed.
Verify Data	Click to verify the following for the selected Component: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note *• To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Feature Properties Dialog Box

The Feature Properties dialog box is displayed when you right-click on a Feature under the **Deployment Status** node and then select **Properties** from the shortcut menu.

This dialog box contains the following options:

Table 20-9 • Feature Properties Dialog Box Options

Options	Description
Name	Name of selected Feature.
Title	Title of selected Feature.
Parent	Parent Feature of this Feature (if one exists).
Description	Description of this Feature.
Last Used	Date this Feature was last used.
Usage Count	Number of times this Feature has been used.

Table 20-9 • Feature Properties Dialog Box Options (cont.)

Options	Description
Verify Data	Click to verify the following for the selected Feature: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note *• To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Install or Configure Feature Dialog Box

The Install or Configure Feature dialog box is displayed when you right-click on a Feature under the Deployment Status node and then select Configure from the shortcut menu.

If you select an option on this dialog box and click OK, QualityMonitor will attempt to install or configure the selected Feature to the settings you specify. Select one of the following options:

- **Default**—Files will be installed to their default location.
- **Local**—Files will be installed on the local machine.
- **Source**—Files will be run from the installation source.
- **On Demand**—Files will be installed when needed.

Note *• To complete this operation, you may need the source from which the selected Feature was installed.*

Install or Configure Product Dialog Box

The **Install or Configure Product** dialog box is displayed when you right-click on the **Deployment Status** node and then select **Configure** from the shortcut menu. If you select an option on this dialog box and click **OK**, QualityMonitor will attempt to install or configure the Product to the settings you specify.

Installation Location

Select one of the following options:

- **Default**—Files will be installed to their default location.
- **Local**—Files will be installed on the local machine.

- **Source**—Files will be run from the installation source.
- **On Demand**—Files will be installed when needed.

Note • *To complete this operation, you may need the source from which the selected feature was installed.*

Installation Type

Select one of the following options:

- **Minimum**—Only the essential Features will be installed.
- **Typical**—Most commonly used Features will be installed.
- **Complete**—All of the program's Features will be installed.

Installed Data Dialog Box

The Installed Data dialog box appears when you are using MSI Doctor to verify package deployment status, and you perform the following steps:

Task ***To view the Installed Data dialog box:***

1. Go to the Deployment Status View and right-click on the **Deployment Status** node, one of the Features listed under it, or a component listed on the right.
2. Select **Properties** from the shortcut menu to display the Product Properties Dialog Box, Feature Properties Dialog Box, or Component Properties Dialog Box.
3. Click **Verify Data**. QualityMonitor then checks all of the files and registry entries included in the selected Product, Feature, or Component and then displays them on the **Installed Data** dialog box.

Files Tab

Files are verified by checking their existence in the specified location, and comparing the file size specified in the .Windows Installer package to that of the file on the system. On the **Files** tab, the following icons are used to identify verification errors:

Table 20-10 • Verification Error Icons

Icon	Description
	File is missing.
	File has a different version or size than that specified in the Windows Installer package

If you double-click on an item listed on the **Files** tab, the Windows Explorer opens to the folder containing the selected file.

Registry Tab

The **Registry** tab lists all of the registry keys and value names in the selected Product, Feature, or Component. Registry data is verified by checking the existence of the registry key and the value name (if one exists). The icon is used to indicate that a registry key or value name is incorrect or missing.

Note • *Only the registry keys and value names are verified; the values themselves are not verified.*

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Open QualityMonitor Project Dialog Box

The Open QualityMonitor Project dialog box opens when you select to open a QualityMonitor project from the Welcome page, or when you either select **Open** from the **File** menu or click the **Open** button on the toolbar.

From this dialog box, you can either select to open an existing QualityMonitor file (and subsequently enter or browse to it), or select to create a project file based on an installed MSI-based application.

Options Dialog Box

The Options dialog box, available by selecting **Options** from the **Tools** menu, has options on two tabs: General Tab and Exclusions Tab.

General Tab

The following options are included on the **General** tab:

Table 20-11 • Options Dialog Box—General Tab Options

Options	Description
Update the test case status automatically after executing test items	Select this option if you want to automatically update test case status after executing test items. If you do not use this functionality, the View List will not automatically update after a test item has been executed.

Table 20-11 • Options Dialog Box—General Tab Options (cont.)

Options	Description
Show Isolation Tests	Use this option to choose when you would like to Show Isolation Tests after performing a Lockdown and Runtime test. You have the following options: • **Always**—Show this view for all the executables run, irrespective of the presence of records in the IsolatedComponent table and in MsiAssembly SXS records. • **Never**—This view will not be shown irrespective of the data. • **Only if the Application is Isolated**—Show this view only if the MSI Package has either IsolatedComponent records or MSIAssembly SXS records. • **Only if the Running Operating System supports Isolation** ***Note*** *• AdminStudio always stores the information in the project but, the UI selection you make here determines whether this view will be populated.*
Template file	Select a User Defined Tests template file. If you also select the **Load Templates on Project Open** open option, all of the Test Cases in the selected **Template file** will be automatically loaded when a QualityMonitor project is open.
Exclusion file	Select an exclusion file to use to filter the test results in the **Lockdown and Runtime** views. By selecting an exclusion file from a shared location, multiple people can use the same error exclusion settings.

Exclusions Tab

When a Deployment Test is run on a package, some of the tests related to Class IDs, Prog IDs, and Type Library IDs fail because they refer to components which belong to the operating system rather than the software which is being tested. These errors have no impact on the integrity of the software being tested, and cause confusion among some users testing the software. Users need to be able to prevent error messages caused by files that are not affecting the performance of the software package to be listed in the test results.

On the **Exclusions** tab, to prevent these operating systems errors from being reported, you can specify a list of files to be excluded when any of the Class ID, Prog ID, or Type Library ID Deployment Tests are run. You can maintain a different list for each of these three Deployment Tests.

Note *• After a Deployment Test has been run, the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View. The items included in the exclusion lists are not shown in these views, but are still stored in the QualityMonitor Project File (`.iqm`). When this project file is opened again in QualityMonitor, the results are checked against the exclusion list before being displayed in the Class IDs View, Prog IDs View, or Type Libraries View.*

The following options are included on the **Exclusions** tab:

Table 20-12 • Options Dialog Box—Exclusion Tab Options

Options	Description
Exclusion list	Select the type of Deployment Test that you want to modify the exclusion list for: • Class ID • Prog ID • Type Libraries
Listing	Listing of the Class IDs, Prog IDs, or Type Libraries that you have chosen to exclude from the selected Deployment Test. The following information is included: • **Identifier**—Number identifying the Class ID, Prog ID, or Type Library. • **File Name**—Name of file that contains the Class ID, Prog ID, or Type Library. • **Status**—The exclusion status is either **Active** or **Inactive**. Only **Active** exclusions are excluded from the test results. **Inactive** exclusions are neglected/omitted from the exclusion process. The only way to change the status from **Active** to **Inactive** (or vice versa) is to manually edit the exclusions file. • **Type**—The exclusion type is either **User** or **System**. Exclusions added by the user are of **User** type. Users can delete only **User** type exclusion entries.
Add	Click to open the Add Exclusions Dialog Box, where you can select a Class ID, Prog ID, or Type Library to exclude from Deployment Tests.
Remove	Click to delete the selected Class ID, Prog ID, or Type Library from the exclusion list.

Product Properties Dialog Box

The Product Properties dialog box is displayed when you right-click on the **Deployment Status** node and then select **Properties** from the shortcut menu.

This dialog box contains the following options:

Table 20-13 • Product Properties Dialog Box Options

Options	Description
Version	Product version.
Publisher	Manufacturer of Product.
Product Code	Number which uniquely identifies this Product.
Local	Directory on local machine where this MSI file is located.
Registered to	Registered user of Product.

Table 20-13 • Product Properties Dialog Box Options (cont.)

Options	Description
Product ID Status	Installation status of this Product, such as "The product is installed for the current user."
Help Link	Main help link for the Product.
Installed on	Date Product was installed.
Installed from	Location where Product was installed from.
Verify Data	Click to verify the following for the selected Product: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note *• To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Re-install Product/Feature Dialog Box

The Re-install Product/Feature dialog box is displayed when you right-click on a Feature under the **Deployment Status** node or you right-click on the **Deployment Status** node and then select **Re-install** from the shortcut menu.

If you select an option on this dialog box and click **OK**, QualityMonitor will attempt to reinstall the selected Feature(s) to the settings you specify. Select one option from the **Select Reinstall Mode** or **Additional Reinstall Modes** property:

Table 20-14 • Re-install Product/Feature Dialog Box Options

Option	Description
Select Reinstall Mode	Select one of the following options: • Repair all detected reinstall problems • Reinstall only if file is missing • Force all files to be reinstalled

Table 20-14 • Re-install Product/Feature Dialog Box Options (cont.)

Option	Description
Additional Reinstall Mode	Select one of the following options: • Reinstall if file is missing, or an older version exists • Reinstall if file is missing, or an older or equal version exists • Reinstall if existing file has different version • Verify that required user registry entries are present • Verify that required local machine registry entries are present • Recreate all shortcuts

Runtime Test Filters Dialog Box

The Runtime Test Filters Dialog box opens when you click the **Set Filters** button on one of the Lockdown and Runtime Test views: Files View, Folders View, Registry Entries View, or Isolation Tests View.

All of the errors that were generated for that Lockdown and Runtime Test Case are listed. If you want to exclude specific errors from future Lockdown and Runtime tests, select those errors and click **OK**.

These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file.

Test Item Information Dialog Box

The Test Item Information dialog box is displayed when you right-click on a Test Item and select **Test Item Information** from the shortcut menu. The following information is included:

Table 20-15 • Test Item Information Dialog Box Options

Option	Description
Test Item	The file name and path of the selected Test Item.
Status	The status of the selected Test Item: **Passed**, **Failed**, or **Pending**.
Comments	Any comments that were entered to document this Test Item.
Test Details	Specific test details that can include the error message associated with the selected Test Item that was generated during testing. These error messages can help you diagnose issues with the package.

Test Progress Dialog Box

The Test Progress dialog box opens when you execute Test Items. Note the following:

- If you are performing Test Cases which have automatic execution (such as Type Libraries, Prog IDs, Services, or Class IDs), this dialog box opens briefly and automatically closes when execution is complete.
- For non-automatic Test Cases (Help Files, File Associations, and Shortcuts), this dialog box opens for each Test Item selected, allowing you to run the test and perform any necessary manual actions.
- Following execution of Test Items, the Test Result Dialog Box appears.

Test Result Dialog Box

The Test Result dialog box opens following execution of each Test Case requiring manual operations (Help Files, File Associations, and Shortcuts).

You can enter **Comments** about the execution of the functionality, and click **Yes** or **No** depending on whether the Test Item passed.

Views

The following views are available in QualityMonitor:

- Welcome to QualityMonitor View
- Product Information View
- Test Cycle Summary View
- Deployment Tests View
  - Class IDs View
  - File Associations View
  - Help Files View
  - Prog IDs View
  - Shortcuts View
  - Type Libraries View
  - ODBC Data Sources View
  - ODBC Drivers View
  - Services View
- Lockdown and Runtime Tests View
  - Runtime Execution Details View
  - Files View
  - Folders View
  - Registry Entries View
  - Isolation Tests View
- User-Defined Tests View

- Test Case View
- Deployment Status View

Welcome to QualityMonitor View

The Welcome to QualityMonitor View is the view that is displayed before a project is created or opened. From this view you can choose to create a new QualityMonitor project, browse to an existing project, or open the most recently used project.

This view also lists the three major steps involved in using QualityMonitor to ensure package quality:

- **Open project**—Create or open a QualityMonitor project.
- **Run tests**—Run deployment tests, runtime and lockdown tests, or user-defined custom tests.
- **Analyze results**—Analyze the results of the tests.

Product Information View

The Product Information view displays information about the package you are testing in QualityMonitor.

The following data is displayed:

Table 20-16 • Product Information View Options

Option	Description
Application Name	The name of the application.
Author	The person or company who created the application.
Product Code	The package's product code.
Package Code	The package's package code.
Installed On	The date when the package was installed on the system.
Version	The package's version.

Note • *In the View List, this view is titled with the product name.*

Test Cycle Summary View

The Test Cycle Summary view provides statistics on the number of Test Cases and Test Items in the QualityMonitor project, and the ratio of cases and items passed, failed, or pending. If you add additional Test Cases, or perform runtime checking, the number of Test Cases and items will increase.

Deployment Tests View

The Deployment Tests View provides a summary of all Deployment Tests.

Deployment tests help you with up to several Test Cases to run on your Windows Installer-based application. Tests are only available if the application has the corresponding associated data (for example, if there are no shortcuts, you cannot run the Shortcuts Test Case).

QualityMonitor includes the following deployment tests:

- Class IDs View
- File Associations View
- Help Files View
- Prog IDs View
- Shortcuts View
- Type Libraries View
- ODBC Data Sources View
- ODBC Drivers View
- Services View

Automatically Running All Deployment Tests Silently

You can choose to run all deployment tests silently (without prompting for user input) using either the Interface or the command line.

From the Interface

You can choose to run all deployment tests silently (without prompting for user input) by making a selection in the QualityMonitor interface.

Task ***To run all deployment tests silently from the Interface, do one of the following:***

1. On the QualityMonitor **Product Information View**, select the **Deployment Tests** root node and then do one of the following:
   - Click the **Execute All Deployment Tests** button.
   - Select A**ll Deployment Tests** from the **Execute** menu.
   - Click the **Execute All Deployment Tests** toolbar button:

   When you select one of these options, a dialog box with a progress bar and an option to cancel will be displayed.

From the Command Line

You can also run all deployment tests silently by entering a command in the command line. See Running QualityMonitor from the Command Line for more information.

Class IDs View

The **Class IDs** Deployment Test is run to determine if the Class ID COM objects can be instantiated programatically.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

File Associations View

The **File Associations** Deployment Test is run to determine if all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Help Files View

The **Help Files** Deployment Test is run to determine if the help files are installed and can be launched correctly.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Prog IDs View

The **Prog IDs** Deployment Test is run to ensure that the Prog IDs COM objects can be instantiated programatically.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Shortcuts View

The **Shortcuts** Deployment Test is run to determine if each shortcut is installed and if it successfully launches the shortcut target.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Type Libraries View

The **Type Libraries** Deployment Test is run to determine if the Type Libraries COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Manifests View

The **Manifests** Deployment Test is run to test the manifests and assemblies used to isolate a Windows Installer package.

The Manifests Deployment Test tests information from the `MsiAssembly` and `MsiAssemblyName` tables. QualityMonitor reads through the manifest/assembly files and performs the baseline Class IDs, Prog IDs, or Type Libraries testing for each entry in the files.

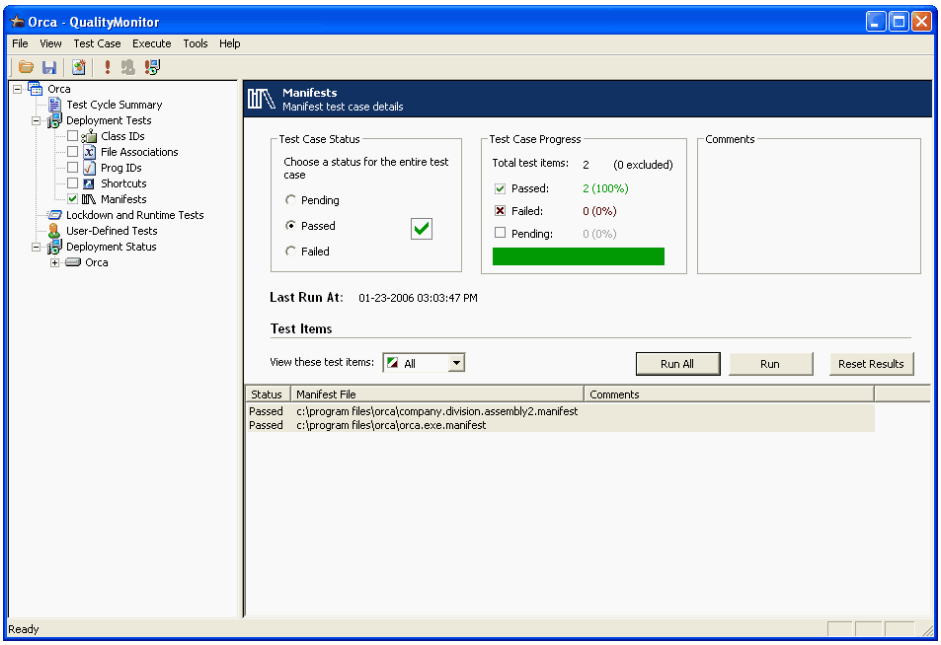


Figure 20-3: QualityMonitor Manifests View

Right-click on the **Test Item** you want to run and select **Run** from the shortcut menu. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.

When testing is finished, results are recorded in the **Test Case Progress** area. Also, the **Status** of each test item (**Passed**, **Failed**, or **Pending**) is listed next to the **Manifest File** name.

When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box, right-click on the Test Item and then select **Test Item Information** from the shortcut menu.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments. If desired, you can also enter comments in the **Comment** field on this view.

ODBC Data Sources View

The **ODBC Data Sources** Deployment Test is run to verify the ODBC data sources.

At the bottom of the view, you can see all ODBC DSNs associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Note • *On the ODBC Data Sources View, only those data sources that belong to the current logged-in user are listed.*

- *For certain ODBC data sources, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to prompt for more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.*

ODBC Drivers View

The **ODBC Drivers** Deployment Test is run to verify ODBC drivers.

At the bottom of the view, you can see all ODBC drivers associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Note • *On the ODBC Drivers View, only those drivers that belong to the current logged-in user are listed.*

- *For certain ODBC drivers, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.*

Services View

The **Services** Deployment Test is run to determine if all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Lockdown and Runtime Tests View

From the **Lockdown and Runtime Tests View**, you can select whether you want to perform runtime checking using a shortcut or an executable in the installed package.

You can select an item in the associated list and click **Run** to launch the application. After exercising the application's functionality and closing it, additional views will appear associated with the executable. These views initially display test items that failed during application operation, and are grouped into:

- Runtime Execution Details View
- Files View
- Folders View
- Registry Entries View
- Isolation Tests View

If you want to execute tests in the context of a different user (under a different user account), click **Run As**. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

Caution • *Lockdown and runtime checks cannot be performed on Windows 9x–based systems.*

Runtime Execution Details View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. When you select this executable or shortcut node, the Runtime Execution Details View opens, listing a summary of the execution of the Lockdown and Runtime tests.

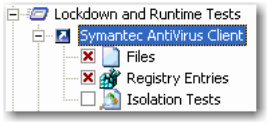


Figure 20-4: Shortcut Node under the Lockdown and Runtime Tests Node

On the Runtime Execution Details View, the following information is included:

Table 20-17 • Lockdown and Runtime Tests View / Runtime Execution Details View Options

Option	Description
Progress	The Progress area includes the following information: • **Total test cases**—Number of test cases (Files, Registry Entries, Folders, Isolation Tests for the selected executable or shortcut) that generated failures plus those test cases that have not yet been completed. • **Passed, Failed, Pending**—Percentage of total test cases that passed the test, failed the test, or have not yet been executed. If an executable or shortcut was run without any failures, the Progress area is not displayed.
Last Run At	Date and time that the last Lockdown and Runtime test was performed, and the name of the user who performed that test.

Files View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any file errors were generated during this test, then the **Files** node appears.

When the **Files** node is selected, the Files View opens and includes the following information:]

Table 20-18 • Lockdown and Runtime Tests View / Files View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. • If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. • If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. • If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.
Test Case Progress	The Test Case Progress area includes the following information: • **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. • **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the files that were executed when this Lockdown and Runtime Test was executed.
View these test items	Select one of the following to filter the file listing: **All**, **Passed**, **Failed**, or **Pending**.
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the file listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Folders View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any file errors were generated during this test, then the **Folders** node appears.

When the **Folders** node is selected, the Folders View opens and includes the following information:]

Table 20-19 • Lockdown and Runtime Tests View / Folders View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. • If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. • If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. • If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.
Test Case Progress	The Test Case Progress area includes the following information: • **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. • **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the folders that contained files that were executed when this Lockdown and Runtime Test was executed.
View these test items	Select one of the following to filter the listing: **All**, **Passed**, **Failed**, or **Pending**.
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.

Table 20-19 • Lockdown and Runtime Tests View / Folders View Options (cont.)

Option	Description
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Registry Entries View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any Registry Entry errors were generated during this test, then the **Registry Entries** node appears.

When the **Registry Entries** node is selected, the Registry Entries View opens and includes the following information:]

Table 20-20 • Lockdown and Runtime Tests View / Registry Entries View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. ● If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. ● If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. ● If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.
Test Case Progress	The Test Case Progress area includes the following information: ● **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. ● **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the Registry Entries that were tested when this Lockdown and Runtime Test was executed.

Table 20-20 • Lockdown and Runtime Tests View / Registry Entries View Options (cont.)

Option	Description
View these test items	Select one of the following to filter the listing: **All**, **Passed**, **Failed**, or **Pending**.
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Isolation Tests View

You can run Isolation Tests to display the location of all portable executable (PE) files (`dll/ocx/exe/tlb/olb`) that are launched from a process while performing a Lockdown and Runtime test. Viewing a listing of these portable executable file names and paths makes it easier for you to ensure that the application is fully isolated.

After you perform a Lockdown and Runtime Test for an executable (`.exe`) or a shortcut on the Lockdown and Runtime Tests View, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.

When you select this **Isolation Tests** node, the filenames of the portable executable files and their paths are listed. By default, the status of all these items is **Pending**. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

Note *• The Isolation Tests node will be added to the Lockdown and Runtime Tests tree only if the selected executable launches at least one portable executable file.*

- *You can specify when you would like the **Isolation Tests** node on the **General** tab of the Options Dialog Box. On the **General** tab, select an option from the **Show Isolation Tests** list .*
- *QualityMonitor does not support isolation testing under Windows 2000.*

User-Defined Tests View

As your business practices dictate, you can add additional, custom tests to the QualityMonitor project file. This is accomplished by right-clicking on the User-Defined Tests view and selecting Add Test Case.

Test Case View

When you select a user-defined Test Case under User-Defined Tests on the View List, the Test Case View opens.

This view contains the following options:

Table 20-21 • Test Case View Options

Option	Description
Test Case Status	Specify the status of the selected Test Case by selecting one of the following options: • **Pending**—Test case has not been executed. • **Passed**—Test case has been executed and has passed. • **Failed**—Test case has been executed and has failed.
Instructions	Enter any instructions to explain how to execute this Test Case.
Select an executable	Select an executable to launch when this Test Case is run.
Comments	Enter comments to document the purpose of this Test Case or to note any important issues.

Deployment Status View

The Deployment Status View lists all of the products and features in the MSI package.

Products and features in the MSI package are listed in the Deployment Status tree, with an icon indicating its status:

Table 20-22 • Deployment Status View Status Icons

Icon	Description
	installed
	not installed
	a key file is either missing or does not match the version or size of that file recorded in the MSI file

When you select the Deployment Status node, all of the components in all of the product features are listed on the right. If you select an individual feature, only those components within that feature are listed. The following information is displayed:

Table 20-23 • Deployment Status View Options

Option	Description
Component Name	Name of all components in the MSI package or selected feature.
Component Status	Status of the listed component: either installed (), not installed (), or a key file is either missing or does not match the version or size of that file recorded in the MSI file ()
Component Location	Location of installed component.
Component ID	GUID of the component, which uniquely identifies it.

21

Distributing Applications and Packages

AdminStudio provides several provides straightforward ways to distribute your applications and packages. Application Manager opens different versions of the Distribution Wizard depending upon what is selected in the tree when you click the **Distribute** button:

- **Application**—If you have an application (or a group containing applications) selected, a Distribution Wizard that is customized to publishing applications to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, Symantec Altiris Management Server, Microsoft App-V Server, Workspace ONE Server, JAMF Casper Suite Server, and Microsoft Intune opens.
- **Package**—If you have a package selected, the Package Distribution Wizard opens, which is customized to preparing packages for distribution to System Center 2007 or 2012 Configuration Manager, ZENworks Configuration Management, Altiris 6.5 Notification Server, an FTP location, a network location, or an administrative installation.

This section covers how to publish both applications and packages using AdminStudio's distribution tools.

Table 21-1 • AdminStudio Distribution Tools

Tool	Functionality
Distribution Wizard	You can use the Distribution Wizard to publish applications to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, Symantec Altiris Management Server, Microsoft App-V Server, Workspace ONE Server, Microsoft Intune, Custom Distribution System, and JAMF Casper Server. See Distributing Applications Using the Distribution Wizard. The following distribution types are supported: • **ConfigMgr (Formerly called as System Center Configuration Manager)**—Supports applications containing Windows Installer, App-V (4.x and 5.x), Apple iOS mobile apps (local and public store), Google Android mobile apps (local and public store), Microsoft UWP app packages, MSI packages, MSIX packages and legacy installer packages. • **Citrix XenApp Server**—Supports applications containing Citrix XenApp profiles and App-V 4.x packages. • **Microsoft App-V Server**—Supports applications containing Microsoft App-V (4.x and 5.0) packages. • **JAMF Casper Suite Server**—Supports macOS desktop applications, including Apple disk image packages (`.dmg`), Apple installer packages (`.pkg`) and links to Mac App Store apps. • **Workspace ONE Server**—Supports applications containing Apple iOS mobile apps (local and public store), MSI packages, EXE packages, and Google Android mobile apps (local and public store) • **Symantec Altiris Management Server**—Supports applications containing Windows Installer, VMware ThinApp, and legacy installer packages. • **Microsoft Intune**—Supports publishing MSIX and MSI packages via Microsoft Intune distribution system. • **Custom Distribution System**—Supports to publish applications to an endpoint management system that is supported/not supported out-of-the-box by AdminStudio.

Table 21-1 • AdminStudio Distribution Tools

Tool	Functionality
Package Distribution Wizard	You can use the Package Distribution Wizard to publish App-V, Windows Installer, and legacy packages to Microsoft System Center 2007 Configuration Manager or prepare them for distribution through a wide variety of distribution methods including: • ZENworks Configuration Management • Administrative installation • FTP location • Network location • Altiris 6.5 Notification Server See Distributing Packages Using the Package Distribution Wizard.

Distributing Applications Using the Distribution Wizard

You can use the Distribution Wizard to publish an application or group of applications from the Application Catalog to a distribution system. Each supported distribution system supports different deployment types.

- Supported Deployment Types Per Distribution System
- Supported Distribution Systems Per Deployment Type

Note *• AdminStudio supports the following ConfigMgr (Formerly called as System Center Configuration Manager):*

- *SCCM 2012 or later supports deployment types like Windows Installer, App-V (4.x and 5.0), Apple iOS (local and public store), Google Android (local and public store), Microsoft UWP app packages (.appx), Legacy installer, PowerShell wrapped packages (.ps1), and MSI Package*
- *ConfigMgr 2002 or later supports the MSIX Package deployment type.*

Supported Deployment Types Per Distribution System

The following table lists the supported deployment types per distribution system:

Table 21-2 • Supported Deployment Types Per Distribution System

Distribution System	Supported Deployment Types
ConfigMgr Server **(Formerly called as System Center Configuration Manager)**	• Windows Installer • App-V (4.x and 5.0) • Apple iOS (local and public store) • Google Android (local and public store) • Microsoft UWP app packages (.appx) • Legacy installer • PowerShell wrapped packages (.ps1) • MSI Package • MSIX Package ***Important*** • *Note the following:* • *To publish mobile apps, SCCM 2012 SP1 or later is required.* • *To publish MSIX packages, ConfigMgr 1906 is required.* • *To publish other deployment types, SCCM 2012 or later is required.*
Citrix XenApp Server	• Citrix XenApp profiles • App-V 4.x
Microsoft App-V Server	• App-V (4.x and 5.0)
Symantec Altiris Management Server	• Windows Installer • VMware ThinApp • Legacy installer • PowerShell wrapped packages (.ps1)
JAMF Casper Suite Server	• Apple disk image package (.dmg) • Apple installer package (.pkg) • Apple Mac App Store application

Table 21-2 • Supported Deployment Types Per Distribution System

Distribution System	Supported Deployment Types
Workspace ONE Server	• Apple iOS (local and public store) • Google Android (local and public store) • MSI Package • EXE Packages ***Note*** *• If you are using an Application Catalog that has been upgraded from a release of AdminStudio prior to AdminStudio 2013, and the iOS application was imported prior to the upgrade, you will need to reimport the iOS application before you will be able to successfully publish it to Workspace ONE Server.*
Microsoft Intune Server	• MSI Package • MSIX Package • Intunewin
Custom Distribution System	• Windows Installer • App-V (4.x and 5.0) • Apple iOS (local and public store) • Google Android (local and public store) • Microsoft UWP app packages (.appx) • Legacy installer • Citrix XenApp • VMware ThinApp • PowerShell wrapped packages (.ps1) • MSI Package • MSIX Package • Intunewin

Supported Distribution Systems Per Deployment Type

The following table lists the supported distribution systems for each deployment type:

Table 21-3 • Supported Distribution Systems Per Deployment Type

Deployment Type	ConfigMgr)	Citrix XenApp Server	Symantec Altiris Server	Microsoft App-V Server	Workspace ONE Server	Casper Server	Microsoft Intune	Custom Distributio n System
Windows Installer / MSI Package	✓		✓		✓		✓	✓
App-V 4.x	✓	✓		✓				✓
App-V 5.0	✓	*Supported via Microsoft App-V Server*		✓				✓
Apple iOS	✓				✓			✓
Apple macOS	✓					✓		✓
Google Android	✓				✓			✓
Microsoft UWP app packages	✓							✓
Legacy installer	✓		✓		✓			✓
Citrix XenApp		✓						✓
VMware ThinApp			✓					✓
PowerShell wrapped packages (.ps1)	✓		✓					✓
MSIX Packages	✓						✓	✓
Intunewin							✓	✓

Important • *When publishing applications to one of these distribution systems, the selected applications' supported packages will be published. However, if an application contains packages of other deployment types, those packages will be ignored.*

MST files (excluding SoftwareId.mst) are the only MSI supporting files that will be published to Workspace ONE. Other supporting files like MSPs will be skipped during publish.

Files dependent on MSI (other than MSTs) will not be published to Workspace ONE.

Example: cabinet files, configuration xml files, .dlls, .configs etc.

To publish applications from the Application Catalog to a distribution system, perform the following steps.

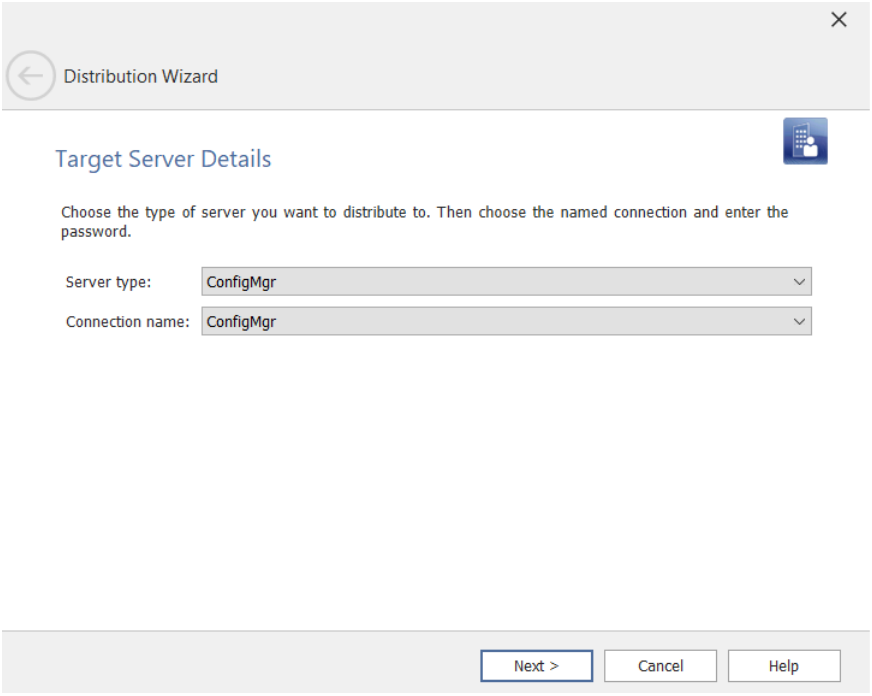
Task ***To publish an application or group of applications to a distribution system:***

1. Create a named connection to a distribution system on the **Server Options > Distribution System** tab of the **Options** dialog box, as described in Creating Multiple Named Connections to Distribution Systems.
2. For each application that you want to publish, specify deployment settings on the following subtabs of the Home Deployment Type View:
   - Deployment Data Tab
   - App-V Deployment Data Tab
   - XenApp Deployment Data Tab
   - Altiris Deployment Data Tab
   - Workspace ONE Deployment Data Tab
   - Microsoft Intune Deployment Data Tab

   Important • *In order to publish to Citrix XenApp Server, there are mandatory fields that you must specify on the XenApp Deployment Data Tab, as described in Specifying a Package's XenApp Deployment Settings. For all other deployment technologies, editing deployment data is optional.*

   Note • *If you are using an Application Catalog that has been upgraded from a release of AdminStudio prior to AdminStudio 2013, and the iOS application was imported prior to the upgrade, you will need to reimport the iOS application before you will be able to successfully publish it to Workspace ONE Server.*
3. In the Application Catalog tree, select the application or group of applications that you want to publish and click the **Distribute** button in the ribbon. The **Target Server Details** opens.

   Tip • *Instead of clicking the **Distribute** button in the ribbon, you could instead select **Distribute Application** or **Distribute Group** from the shortcut menu.*



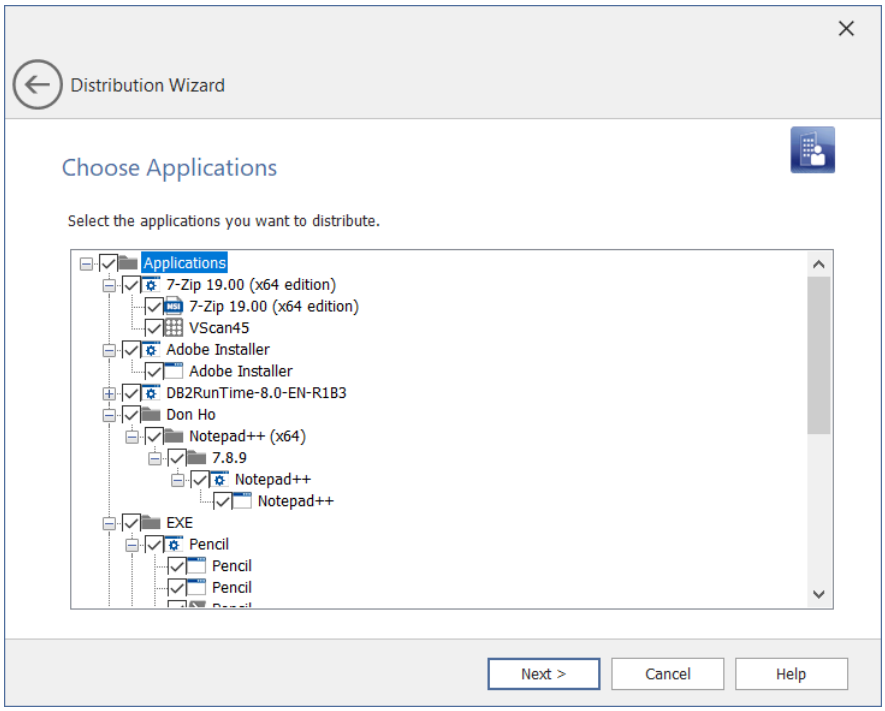
4. From the **Server type** list, select the type of distribution system you want to publish applications to:
   - **ConfigMgr**
   - **Microsoft App-V Management Server**
   - **Symantec Altiris Management Server**
   - **Citrix XenApp Server**
   - **Casper Suite Server**
   - **Custom Distribution System**
   - **Workspace ONE Server**
   - **Microsoft Intune**
5. From the **Connection name** list, select the named connection to the distribution system server that you want to publish applications to.

   Note • *In order to populate this list, you must have already set up at least one named connection to a distribution system server, as described in Creating Multiple Named Connections to Distribution Systems.*

   Important • *Because you cannot publish applications to System Center 2007 Configuration Manager, do not select a named connection to a System Center 2007 Configuration Manager server from this list. To publish a package to System Center 2007 Configuration Manager, you need to use the Package Distribution Wizard, as described in Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager).*
6. Choose the application or applications that you want to publish and click **Next**. The **Choose Applications panel** opens with the application or group that was selected when you clicked the **Distribute** button already selected.

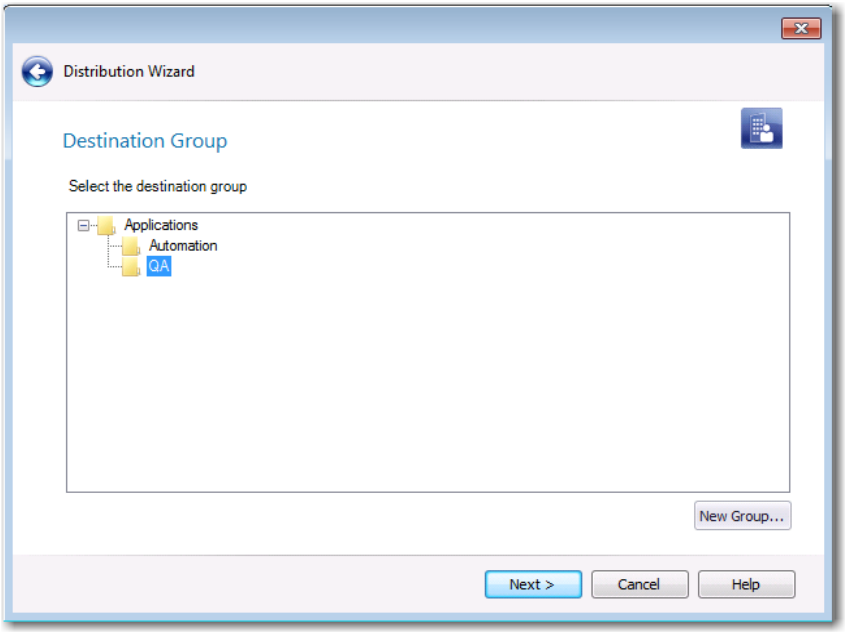
Note • *Based on the **Target Server Type** selection, only supported deployment types will be appearing under **Choose Applications** panel. You can select individual package under an application node which needs to be published to the desired **Distribution System**.*

Note • *When you select Custom Distribution System, you can select only one application/package. If you select more than one, an error message popup will appear.*



7. Click **Next**. One of the following occurs:

- If you are publishing to ConfigMgr or Workspace ONE, the **Destination Group** panel opens. Proceed with Step **8**.



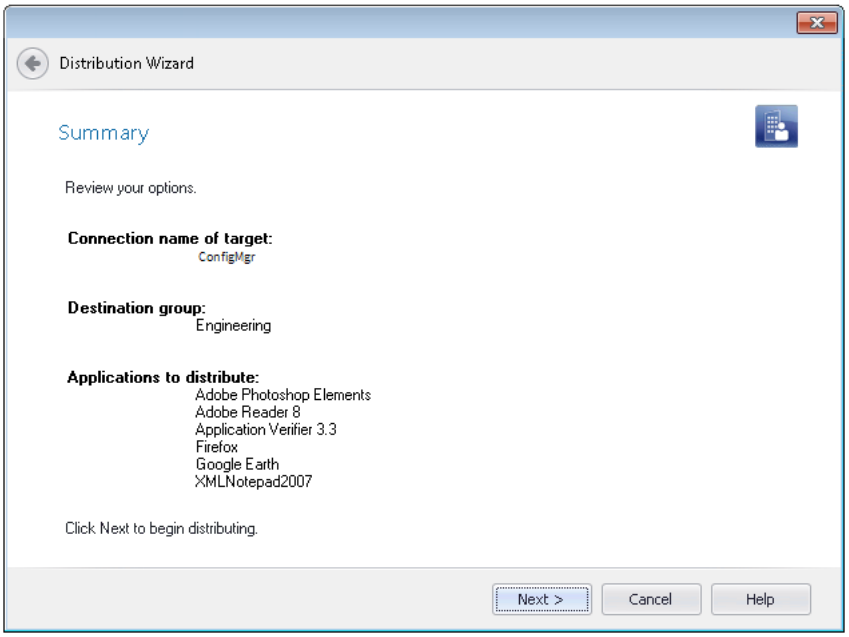
- For all other distribution systems, the **Summary** panel opens. Skip to Step **9**.

8. Select the group in the connected ConfigMgr (Formerly called as System Center Configuration Manager) or Workspace ONE server that you want to publish applications to.

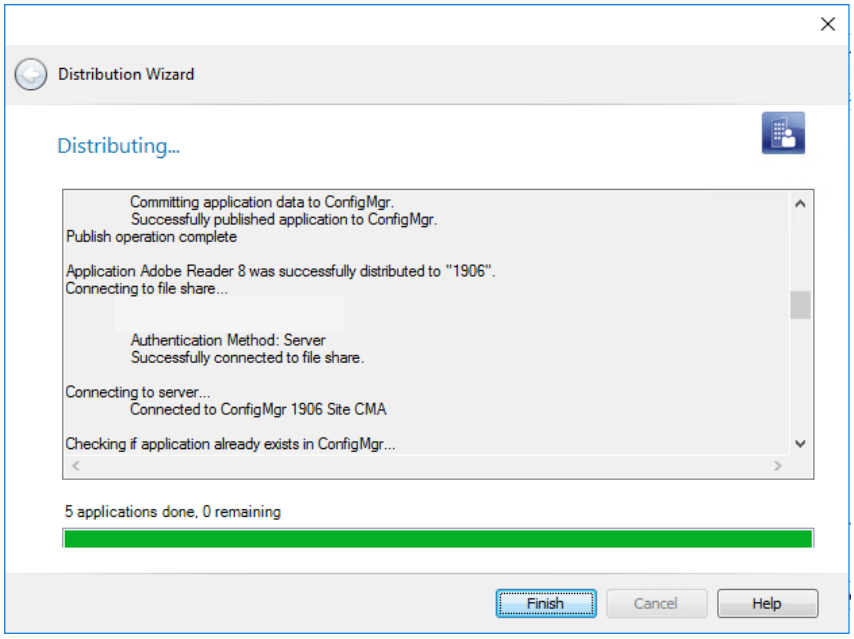
Note • *If you originally imported the application from ConfigMgr (Formerly called as System Center Configuration Manager) server, the* ***Destination Group*** *panel will not open, and the application will be published in its source location.*

Note • *Workspace ONE permits publishing a single application only once to an Organization Group. Therefore, if you attempt to publish an application to an Workspace ONE Organization Group (Distribution Group) that already contains that application, the publication will fail.*

9. Click **Next**. The **Summary** panel opens, displaying a summary of all settings configured in the previous panels.



10. Click **Next**. The distribution begins and the **Distributing** panel opens, which displays a progress bar and status messages during distribution.



11. When distribution is complete, click **Finish** to exit the wizard.

Distributing Packages Using the Package Distribution Wizard

The Package Distribution Wizard provides a straightforward way to distribute your packages or prepare them for distribution through a wide variety of distribution methods.

Important • *You can use the customized Distribution Wizard to publish applications to ConfigMgr (Formerly called as System Center Configuration Manager) and Citrix XenApp Server. See Distributing Applications Using the Distribution Wizard.*

You can use the Package Distribution Wizard to publish App-V, Windows Installer, or legacy packages to Microsoft System Center 2007 Configuration Manager. You can also use the Package Distribution Wizard to assist you in deploying your packages using any of the following distribution types:

Table 21-4 • Supported Distribution Types

Distribution Type	Topic
Administrative Install	Creating Administrative Installations for Packages
FTP Location	Distributing Packages to FTP Servers
Altiris 6.5	Preparing for Altiris 6.5 Distribution
Network Location	Distributing Packages to Network Locations
ConfigMgr	Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager)
ZENworks Configuration Management Distribution	Preparing for ZENworks Configuration Management Distribution

You can launch the Package Distribution Wizard from Application Manager by selecting a package in the tree and then clicking the **Distribute** button in the **Home** tab of the ribbon or by right-clicking on a package and selecting **Distribute Package** from the shortcut menu. You can also launch the Package Distribution Wizard from the Windows Start menu.

Important • *If you right-click on an application in the Application Catalog tree and select **Distribute Application** from the shortcut menu, Package Distribution Wizard will not display the **Distribution Type** panel; instead, you will immediately be prompted to connect to a ConfigMgr (Formerly called as System Center Configuration Manager) Server, since that is the only distribution type that supports applications.*

This section also explains how to deploy a Windows installer package using an InstallShield script-based setup. See Deploying InstallScript MSI Installations.

Creating Administrative Installations for Packages

In an administrative installation, the installation software is copied to a network directory using the administrative install option provided by Windows Installer.

Task ***To distribute your package (and any associated transforms) as an administrative installation:***

1. Launch the Package Distribution Wizard by either clicking on its icon in the Tools Gallery or by right-clicking on a package in the Application Catalog tree and selecting **Distribute Package** from the shortcut menu.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Administrative Install** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer (`.msi`) package that you want to distribute.

   If you launched the Package Distribution Wizard from Application Manager by right-clicking on a package and selecting **Distribute Package** from the shortcut menu, the name in the **Windows Installer Package (.msi)** field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

   - **Not in the Software Repository**—The full name and path of the file is displayed, and you can edit this entry or click **Browse** and select a different package.
   - **In the Software Repository**—Only the name of the file is displayed (not the full path) and this entry cannot be edited or changed.
5. If there are transforms associated with the package, click the New button ( ) in the **Windows Installer Transform Files (*.mst)** area and navigate to the transform you want to add. Repeat as necessary.
6. If desired, add additional Windows Installer properties in the **Specify Additional MSI Properties** field.
7. After specifying the package location, click **Next**. The **Administrative Install** panel opens.
8. From the **Administrative Install** panel, specify or browse to the **Network Directory** to which you want to distribute the package.
9. If desired, you can use short file names during the distribution by selecting the **Use short file names** option.

   Note • *Select this option to force the administrative installation to use the 8.3 file name convention (using the* `SHORTFILENAMES` *property).*
10. Click **Next**. The **Distribution Summary** panel opens.
11. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
12. Once the distribution finishes, click **Finish** to exit the Package Distribution Wizard.

Distributing Packages to FTP Servers

You can choose to distribute a package to an FTP server. If you select the **FTP Location** option on the Package Distribution Wizard **Distribution Type** panel, you will be required to enter the location of the FTP server, and the user name and password to connect to that server.

Task ***To distribute your package (and any associated transforms) to an FTP server:***

1. Launch the Package Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **FTP Location** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the **Windows Installer Package (.msi)** you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the **Windows Installer Transform Files (*.mst)** area and navigate to the transform you want to add. Repeat as necessary.
6. If desired, add additional Windows Installer properties in the **Specify Additional MSI Properties** field.
7. After specifying the package location, click **Next**. The **FTP Location** panel opens.
8. From the **FTP Location** panel, specify the location of the FTP server, and the user name and password to use to connect to the server. Click **Next**.
9. Review the selections you made in the **Distribution Summary** panel. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files.
10. The **Distribution Output** panel displays progress during distribution. Once the distribution finishes, click Finish to exit the Package Distribution Wizard.

Preparing for Altiris 6.5 Distribution

The Distribution Wizard supports the distribution of a setup along with any transforms and files via an Altiris 6.5 Notification Server. A custom script file is required for Altiris distribution. The Distribution Wizard creates this custom script file using an XML template file that is provided: `AltirisTemplate.Config`.

Task ***To prepare your package for Altiris distribution:***

1. Launch the Distribution Wizard. The **Distribution Wizard Welcome** panel opens.
2. Click **Next**. The **Distribution Type pane**l opens.
3. Select **Altiris 6.5** from the **Distribution Type** list and click **Next**. The **Package Information** panel opens.
4. Click the **Browse** button and locate the Windows Installer Package (`.msi`) you want to distribute.

Note • *The package that was selected when the Distribution Wizard was launched is automatically specified.*

5. If there are transforms associated with the package, click the **Add** button in the **Additional Transforms** area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **Altiris Integration** panel opens.
7. In the **Network Directory** field, specify or browse to the network location where you want to store the installation package. The Distribution Wizard remembers the last **Network Location** that is entered and displays it the next time this panel is accessed.

   The Distribution Wizard will copy the Windows Installer package along with any transforms and files to the UNC path specified. Also, the Distribution Wizard will use an XML template file (`AltirisTemplate.config`) to create a custom script file in this location named `<packageName>.Config`.

   Note • *You can edit `AltirisTemplate.config` to customize it for your organization. The file, which is installed with AdminStudio, is located in the Templates folder of the AdminStudio Shared directory. See Altiris XML Template for more information.*
8. In the **Windows Installer Command Line** field, enter any additional properties that you want to pass to the Windows Installer.
9. In the **Altiris Server Location** field, enter the `http:` address for the location of the Altiris Server. The Distribution Wizard remembers the last Altiris Server Location that is entered and displays it the next time this panel is accessed.
10. In the **User Name** and **Password** fields, enter a User Name and Password to log onto the server entered in the **Altiris Server Location** field. The Distribution Wizard remembers the last User Name that is entered and displays it the next time this panel is accessed.
11. Click **Next**. The **Distribution Summary** panel appears, listing the selections you made in the previous panels.
12. Review the information on the **Distribution Summary** panel. If you are satisfied with them, click **Next**.The **Distribution Output** panel displays progress during distribution.
13. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Note • *For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the AdminStudio Shared directory.*

Distributing Packages to Network Locations

To distribute a package to a network directory, select the **Network Location** option on the Package Distribution Wizard **Distribution Type** panel.

Task ***To distribute your package (and any associated transforms) to a network location:***

1. Launch the Package Distribution Wizard. The Package Distribution Wizard **Welcome** panel opens.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Network Location** and click **Next**. The **Package Information** panel opens.

4. On the **Package Information** panel, click the Browse button and locate the **Windows Installer Package (.msi)** you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the **Windows Installer Transform Files (*.mst)** area and navigate to the transform you want to add. Repeat as necessary.
6. If desired, add additional Windows Installer properties in the **Specify Additional MSI Properties** field.
7. After specifying the package location, click **Next**. The **Network Location** panel opens.
8. From the **Network Location** panel, specify or browse to the Network Directory location to which you want to distribute the package, and click **Next**. The **Distribution Summary** panel opens.
9. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
10. Once the distribution finishes, click **Finish** to exit the Package Distribution Wizard.

Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager)

Using AdminStudio's Package Distribution Wizard, you can publish individual Windows Installer (`.msi`), Microsoft App-V 4.x (`.sft`), or legacy setup (`.exe`) packages to ConfigMgr. You can use the Package Distribution Wizard to publish packages on your file system or from your the Application Catalog tree in Application Manager.

Important • *To publish* ***applications*** *to System Center 2012 Configuration Manager, you need to use the application-based Distribution Wizard, as described in Distributing Applications Using the Distribution Wizard.*

Note • *Using the Package Distribution Wizard, App-V 4.x packages can only be published to System Center 2007 Configuration Manager. However, Windows Installer and legacy setups can be published to both System Center 2007 Configuration Manager and System Center 2012 Configuration Manager using the Package Distribution Wizard.*

You can launch the Package Distribution Wizard from Application Manager by selecting a package in the tree and then clicking the **Distribute** button in the **Home** tab of the ribbon or by right-clicking on the package and selecting **Distribute Package** from the shortcut menu. You can also launch the Package Distribution Wizard from the Windows Start menu.

To publish a Windows Installer, App-V, or legacy package from the Application Catalog to a ConfigMgr server, perform the following steps.

Task ***To publish a package to ConfigMgr:***

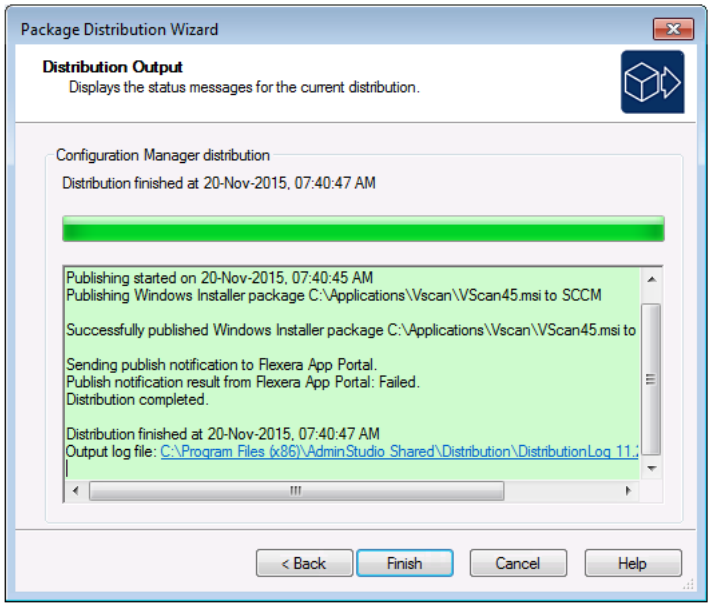
1. Select the package that you want to publish in the Application Catalog tree and click the **Distribute** button in the Application Manager ribbon (or right-click on the package and select **Distribute Package** from the shortcut menu). The **Welcome** panel opens.

   Tip • *You can also launch the Package Distribution Wizard from the AdminStudio Tools Gallery or from the Windows Start menu.*

2. Click **Next**. The **Distribution Type** panel opens.
3. Select **Configuration Manager** and click **Next**. The **Package Information** panel opens.
   - If you launched Package Distribution Wizard with a package selected, information about that package is listed.
   - If you launched Package Distribution Wizard without having a package selected, click the **Browse** button and locate the **Windows Installer Package (.msi), Microsoft App-V Package (.sft), or Legacy Setup Package (.exe)** that you want to distribute.
4. (Windows Installer packages only) If there are transforms associated with the package, click the New button ( ) in the **Windows Installer Transform Files (*.mst)** area and navigate to the transform you want to add. Repeat as necessary.
5. (Windows Installer packages only) If desired, add additional Windows Installer properties in the **Specify Additional MSI Properties** field.
6. Click **Next**. The **Connect to a ConfigMgr Server** panel opens.
7. In the **Server** field, enter the name of your ConfigMgr server.
8. In the **Site Code** field, enter the code that identifies your Configuration Manager site.
9. From the **Authentication** list, choose one of the following options:
   - **Server Authentication**—Choose this option if you want to use ConfigMgr server login identification to log into this server. Then enter the appropriate **User Name** and **Password**.
   - **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this ConfigMgr server.
10. Click **Next**. The **Select Destination Folder** panel opens prompting you to select a location that the ConfigMgr server has access to where you want to publish the selected packages.
11. In the **Location to Publish Packages** field, enter a target path, in UNC format (`\\Server\Share`), of the location where you want to publish the selected packages. Make sure that you enter a location that the ConfigMgr server has access to.
12. From the **Authentication** list, select one of the following options:
    - **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this location.
    - **ConfigMgr Authentication**—Choose this option if you want to use ConfigMgr server authentication (your ConfigMgr server login ID) to log into this location.

- **Server Authentication**—Choose this option if you are publishing to an alternate file server that requires credentials. Then enter the appropriate **User Name** and **Password**.

13. Click **Next**. The **Select Group** panel opens.
14. Select the **Target Group** on the Configuration Manager Server where you want to publish the package and click **Next**. The **Distribution Summary** panel opens.
15. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package. The **Distribution Output** panel displays progress during distribution. When distribution is successful, the message `Successfully published ...` is listed and the background color of the output window turns green.



16. Click **Finish** to exit the Package Distribution Wizard.

Preparing for ZENworks Configuration Management Distribution

Microfocus ZENworks Configuration Management 10 and 11 customers can use the Distribution Wizard for ZENworks Configuration Management to distribute a Windows Installer package (`.msi`)—including any associated transforms—to ZENworks Configuration Management.

To prepare your package for ZENworks Configuration Management distribution, perform the following steps.

Task ***To prepare your package for ZENworks Configuration Management distribution:***

1. Launch the Distribution Wizard for ZENworks Configuration Management by performing the following steps:
   a. Launch the **Package Distribution Wizard**.
   b. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.

c. Select **ZENworks Configuration Management Distribution** and click **Next**.

The **Welcome** panel of the Distribution Wizard for ZENworks Configuration Management opens.

Edition • *If you have AdminStudio ZENworks Limited Edition, you can use a shortcut under* ***AdminStudio Tools*** *to automatically launch the* ***Distribution Wizard for ZENworks Configuration Management****.*

Note • *If you do not want the* ***Welcome*** *panel to be displayed each time you open this wizard, select the* ***Do not show the Welcome panel again*** *option. If this option is selected, the* ***Login*** *panel will be the first panel opened for this wizard.*

2. Click **Next** to continue. The **Login** panel opens.
3. On the **Login** panel, enter the following login information for the ZENworks Configuration Management server that you are connecting to:

Property	Description
User Name / Password	Enter a valid User Name and Password for the ZENworks Configuration Management server you are connecting to.

Server URL	Enter the server URL, machine name, or IP address of the ZENworks Configuration Management server using the following format:
	`http://www.servername.com or http://111.22.333.44`
	If you need to specify a specific port number, append the port number to the end of the URL, such as:
	`http://www.servername.com:123`
	If you are using SSL and you want a secure connection, change the `http` prefix to `https`. For example:
	`https://www.servername.com`

4. Click **Login**. The **Windows Installer Package Information** panel opens.
5. Click **Browse** next to the **Windows Installer Package File (.msi)** field and select the Windows Installer package that will be referenced by this ZENworks Configuration Management bundle.

Important • *All of the files in the selected Windows Installer file's directory and all of its subdirectories will be uploaded to the ZENworks Configuration Management Server.*

After you make your selection, if there are any transform files (`.mst`) in the same directory, they are listed in the **Windows Installer Transform Files (.mst)** area.

Note • *All of the* `.mst` *files that are in the same directory as the selected Windows Installer package are automatically listed in the* ***Windows Installer Transform Files (.mst)*** *list, even if they are not applicable to the selected package. To prevent the inclusion of non-applicable transform files, delete those transforms from the list.*

6. To include transforms with the Windows Installer package:

- Click the New button ( ) in the **Windows Installer Transform Files (.mst)** area and select a transform file. If the package requires multiple transforms, you can repeat the procedure as necessary.
- Select a transform and click the Delete button ( ) to delete a transform from the list.

7. If you want to customize how this package is installed, enter parameters in the **Install Parameters** field. Any actions that you enter will be performed whenever this bundle is installed.
   - The root parameter, which should not be edited or deleted, is `/i packagename.msi`.
   - These parameters are applied to `msiexec.exe` to perform the desired action.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.
8. If you want to customize how this package is uninstalled, enter parameters in the **Uninstall Parameters** field. Any actions that you enter will be performed whenever this bundle is uninstalled.
   - The root parameter, which should not be edited or deleted, is `/x packagename.msi`.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.
9. If you want to customize how this package is repaired, enter parameters in the **Repair Parameters** field. Bundles are repaired by reinstalling missing or corrupted files.
   - The root parameter, which should not be edited or deleted, is `/f packagename.msi`.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - You can apply any of the following additional Repair parameters after the package name:

Parameter	Description
p	Reinstalls a file if it is missing
o	Reinstalls a file if it is missing or if an older version of the file is present on the user's system
e	Reinstalls a file if it is missing or if an equivalent or older version of the file is present on the user's system

Parameter	Description
c	Reinstalls a file if it is missing or if the stored checksum of the installed file does not match the new file's value
a	Forces a reinstall of all files
u or m	Rewrite all required user registry entries
s	Overwrites any existing shortcuts
v	Runs your application from the source and re-caches the local installation package

- For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.

10. Click **Next**. The **Bundle Creation Options** panel opens.
11. Specify whether to update an existing bundle or create a new one by selecting one of the following options:
    a. **Create a new bundle from these Windows Installer package files**—To create a new bundle to reference this Windows Installer package, select this option.
    b. **Update an existing bundle using these Windows Installer package files**—If you want to overwrite an existing bundle to contain this Windows Installer package, select this option, and then select an existing bundle from the tree:
       - **Recommended Bundles**—This group lists the bundles that contain the same Windows Installer package as the one you selected on the Windows Installer Package Information Panel.
       - **All Other Bundles**—This group lists the rest of the existing bundles on the server.
12. After making your selection, click **Next** to proceed. The **Bundle Information** panel opens.
13. Enter information to specify attributes for this bundle on ZENworks Configuration Management:

Property	Properties
Bundle Name	Enter the bundle's name as you want it to appears in ZENworks Control Center (ZCC) and the ZENworks Application Launcher (on managed devices).
Version Number	Enter the bundle's version number. If you are overwriting an existing bundle, and you enter a higher version number than the bundle's original version number, the bundle will be redeployed.
Icon	Click **Browse** and select a shortcut icon graphic (in `.ico`, `.gif`, `.jpg`, `.png`, `.bmp`, or `.exe` format) that ZENworks Application Launcher will display on managed devices. If you do not select an icon file, the standard ZENworks bundle icon will be used.

Property	Properties
Folder	From the **Folder** list, select the folder path that will be used by ZENworks Application Launcher when displaying the bundle on either the device's desktop or **Start** menu. All of the folders defined on the ZENworks server are listed. For example: • **Start Menu**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the **Start** menu, ZENworks Application Launcher creates an `Application\Accounting` folder on the root of the **Start** menu and adds the bundle to it. • **Desktop**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the desktop, ZENworks Application Launcher creates an `Applications\Accounting` folder on the desktop and adds the bundle to it. You can place multiple bundles in a single folder by specifying the same folder path for each of the bundles.
Description	Enter a description of the bundle. This description will be displayed in ZENworks Control Center and the ZENworks Application Launcher (on managed devices).

14. After you have entered bundle information, click **Next**. The **Summary** panel opens, displaying the options you have selected for distributing this Windows Installer package on ZENworks Configuration Management.
15. Click **Publish** to complete the distribution process. The **Publishing Process** panel opens, listing the progress messages while the bundle is being published on ZENworks Configuration Management.
    - **ZENworks error messages**—Any error messages with a numeric prefix that appear on this panel are generated by ZENworks Configuration Management. To resolve these errors, contact your ZENworks Configuration Management System Administrator.
    - **Canceling publication**—If you want to cancel the publication of the bundle on ZENworks Configuration Management, click **Cancel**.
16. When processing is complete, the **Finish** button becomes enabled. Click **Finish** to exit this wizard.

Deploying InstallScript MSI Installations

When deploying an InstallScript MSI installation, the file `setup.exe` needs to be deployed with the InstallScript `.msi` installation file. The `setup.exe` file is required because it launches a file (`isscriptn.msi`) that installs the InstallScript engine required to run the InstallScript code. The *n* in `installscriptn.msi` indicates the version of the InstallScript engine that was used to create the InstallScript MSI installation.

If you want to deploy an InstallScript MSI installation without using `setup.exe`, such as when using Active Directory, you need to first deploy the same version of the InstallScript engine that was used to build the InstallScript MSI installation.

Installing the InstallScript Engine

Sometimes the `isscriptn.msi` file (the file that installs the InstallScript engine) is located in the same directory as the InstallScript `.msi` file. However, in some instances, the `isscriptn.msi` file is compressed within the `setup.exe` file and cannot be accessed.

If the `isscriptn.msi` file is compressed within the `setup.exe` file, you have the following options:

- **If you know which version** of the InstallScript engine was used to create your InstallScript MSI installation, you can get a copy of the InstallScript engine from the AdminStudio installation CD. All the major releases of the InstallScript engine are available in the `InstallScript_Engines` folder on the AdminStudio installation CD.

  Note • *For more information, see the Update to the Latest InstallShield Installation Engines Knowledge Base article at:*

  https://community.flexera.com/t5/InstallShield-Knowledge-Base/Update-to-the-Latest-InstallShield-Installation-Engines/ta-p/4591

- **If you do not know which version** of the InstallScript engine was used to create your InstallScript MSI installation, contact the software vendor to find out the exact version.

Deploying an InstallScript MSI Installation

To deploy an InstallScript MSI installation, configure the setup and the target system in the following manner:

Task ***To deploy an InstallScript MSI installation:***

1. Run the `isscriptn.msi` file to install the appropriate InstallScript engine on the target machine (where *n* is the version of the InstallScript engine that was used to create the application's InstallScript MSI installation).
2. Create a transform for the InstallScript MSI that includes the following changes:
   a. Add the property `ISSETUPDRIVEN` to the property table via the Direct Editor and give it a value of **1**.
   b. Add a condition to the `OnCheckSilentInstall` custom action in the `InstallExecuteSequence` via the Direct Editor that will always resolve to false or remove the custom action from the sequence.
   c. Make any additional changes in the transform, such as populating the serial number, modifying shortcuts or feature states, depending upon your organization's needs and the features and requirements of the application.
3. Next deploy the InstallScript `.msi` package.

If deploying the package via Active Directory, make sure that you set the **Installation User Interface** to **Basic** and specify any transforms that you created for the InstallScript MSI package.

Reference

AdminStudio provides several provides straightforward ways to distribute your applications and packages. Application Manager opens different versions of the Distribution Wizard depending upon what is selected in the tree when you click the **Distribute** button:

- **Application**—If you have an application (or a group containing applications) selected, a Distribution Wizard that is customized to publishing applications to System Center 2012 Configuration Manager, Citrix XenApp Server, Workspace ONE Server, or Symantec Altiris Server opens.
- **Package**—If you have a package selected, the Package Distribution Wizard opens, which is customized to preparing packages for distribution through a wide variety of distribution methods.

AdminStudio includes the following distribution tools:

- Distribution Wizard

- Package Distribution Wizard
- Distribution Wizard for ZENworks Configuration Management

Distribution Wizard

You can use the Distribution Wizard to publish applications to System Center 2012 Configuration Manager, Citrix XenApp Server, Workspace ONE Server, or Symantec Altiris Management Server.

Prior to using the Distribution Wizard to publish applications, you must have already set up a named connection to one of these distribution systems, as described in Creating Multiple Named Connections to Distribution Systems.

Note • *To distribute a package to other distribution systems, a network location, an FTP server, or an administrative location, use the Package Distribution Wizard.*

Note • *For a list of supported deployment types per distribution system, see Distributing Applications Using the Distribution Wizard.*

The Distribution Wizard consists of the following panels:

- Choose Applications Panel
- Target Server Details Panel
- Destination Group Panel
- Summary Panel
- Distributing Panel

Choose Applications Panel

When you launch the Distribution Wizard from Application Manager by selecting an application (or a group that contains applications) in the tree and then clicking the **Distribute** button in the ribbon, the **Choose Applications** panel opens, prompting you to select the applications that you want to distribute.

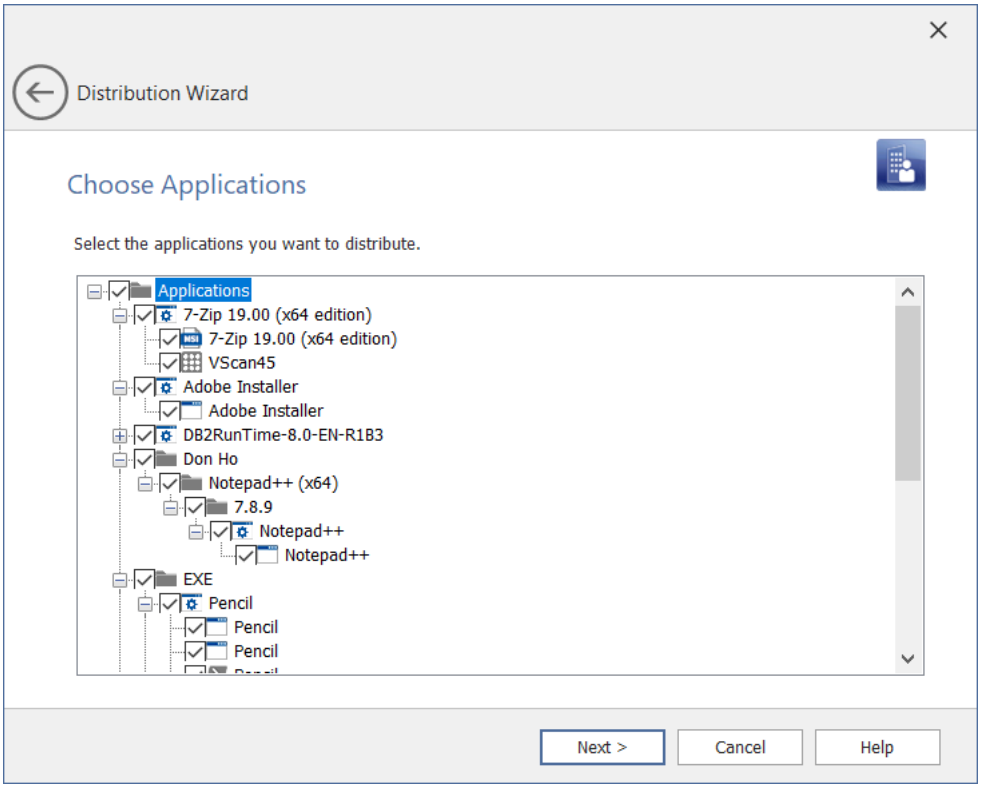


Figure 21-1: Distribution Wizard / Choose Applications Panel

Note • *Based on the **Target Server Type** selection, only supported deployment types will be appearing under **Choose Applications** panel. You can select individual package under an application node which needs to be published to the desired **Distribution System**.*

Note • *When you select Custom Distribution System, you can select only one application/package. If you select more than one, an error message popup will appear.*

Select the applications or groups of applications that you want to publish, and then click **Next** to continue.

Important • *You can only publish applications containing Citrix XenApp packages or Microsoft App-V 4.x packages to Citrix XenApp server. Therefore, when publishing to Citrix XenApp server, if you select an application that contains package deployment types other than Citrix XenApp and App-V 4.x, those packages will be ignored.*

Important • *To publish Apple iOS or Windows Store mobile apps, System Center 2012 Configuration Manager SP1 is required.*

Target Server Details Panel

On the **Target Server Details** panel of the Distribution Wizard, which opens after you have selected the applications that you want to publish on the **Choose Applications** panel, you specify the distribution server you want to distribute to, and you select the named connection to that server that you have already defined.

Note • *AdminStudio supports the following ConfigMgr distribution (Formerly called as System Center Configuration Manager):*

- *ConfigMgr (Formerly called as System Center Configuration Manager) 2012 or later supports the deployment types like Windows Installer, App-V (4.x and 5.0), Apple iOS (local and public store), Google Android (local and public store), Microsoft UWP app packages (.appx), Legacy installer, PowerShell wrapped packages (.ps1), and MSI Package*
- *ConfigMgr 2002 or later supports the MSIX Package deployment type.*

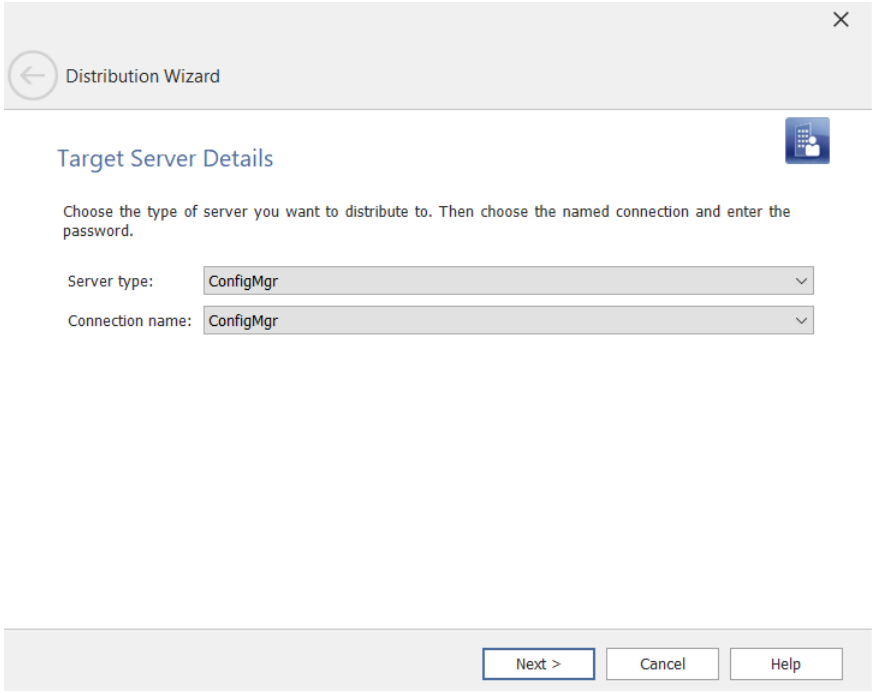


Figure 21-2: Distribution Wizard / Target Server Details Panel

The **Target Server Details** panel contains the following properties:

Table 21-5 • Target Server Details Panel

Property	Description
Server type	Indicate the type of distribution system that you want to publish applications to by selecting one of the following distribution server types: • **ConfigMgr (Formerly called as System Center Configuration Manager)** • **Citrix XenApp Server** • **Workspace ONE Server** • **Symantec Altiris Management Server** • **Microsoft Intune Server**
Connection name	Select the named connection to the distribution server that you want to publish applications to. ***Important*** • *In order to populate this list, you must have already set up at least one named connection to a distribution system, as described in Creating Multiple Named Connections to Distribution Systems.* ***Important*** • *Because you cannot publish applications to System Center 2007 Configuration Manager, do not select a named connection to a System Center 2007 Configuration Manager server from this list. To publish a package to System Center 2007 Configuration Manager, you need to use the Package Distribution Wizard, as described in Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager).*

Destination Group Panel

On the **Destination Group** panel of the Distribution Wizard, which opens after you have selected the target server that you want to publish to on the **Target Server Details** panel, you select the group in the connected ConfigMgr or Workspace ONE server that you want to publish applications to.

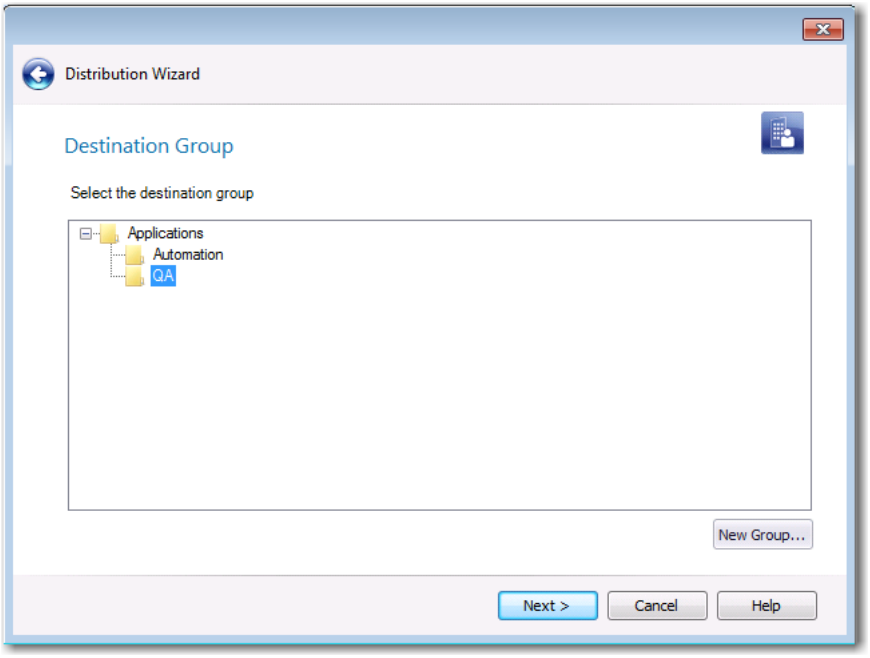


Figure 21-3: Distribution Wizard / Destination Group Panel

Note • *If you originally imported the application from ConfigMgr (Formerly called as System Center Configuration Manager), the* ***Destination Group*** *panel will not open, and the application will be published in its source location.*

Note • *When publishing applications to a Citrix XenApp or Symantec Altiris Management server, the* ***Destination Group*** *panel does not open; all applications are published to the same predesignated destination group, such as* `\\MyServerName\Shared`.

Click **Next** to continue.

Creating a New ConfigMgr Group

When publishing to ConfigMgr (Formerly called as System Center Configuration Manager), you can choose to create a new destination group directly from the **Destination Group** panel of the Distribution Wizard by clicking the **New Group** button and then entering a name for the new group in the **New Group** dialog box. AdminStudio will create that group on the specified ConfigMgr server, and publish the applications to that group.

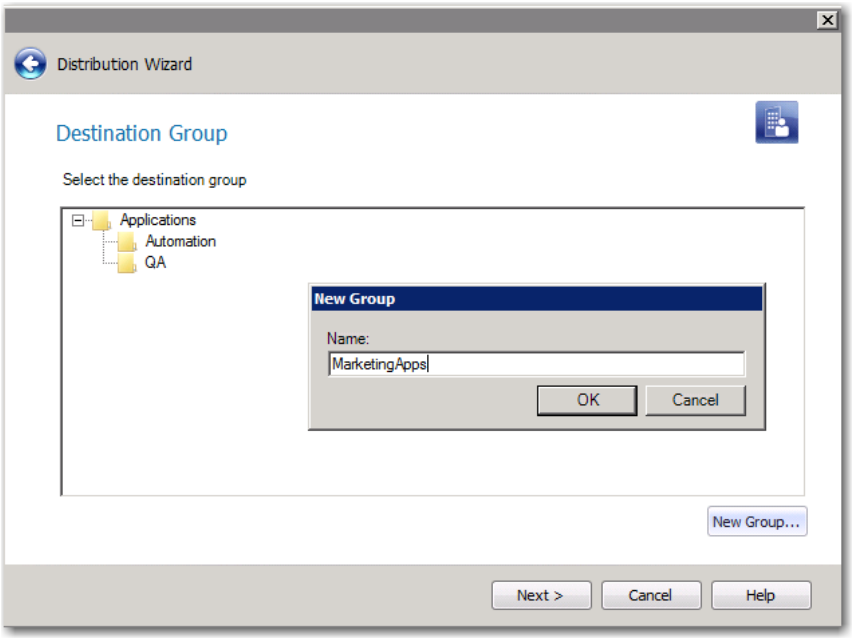


Figure 22: Adding a New Group on Destination Group Panel of Distribution Wizard

Summary Panel

The **Summary** panel displays a summary of all settings configured in the previous panels. When you click **Next**, the distribution begins and the **Distributing...** panel is displayed.

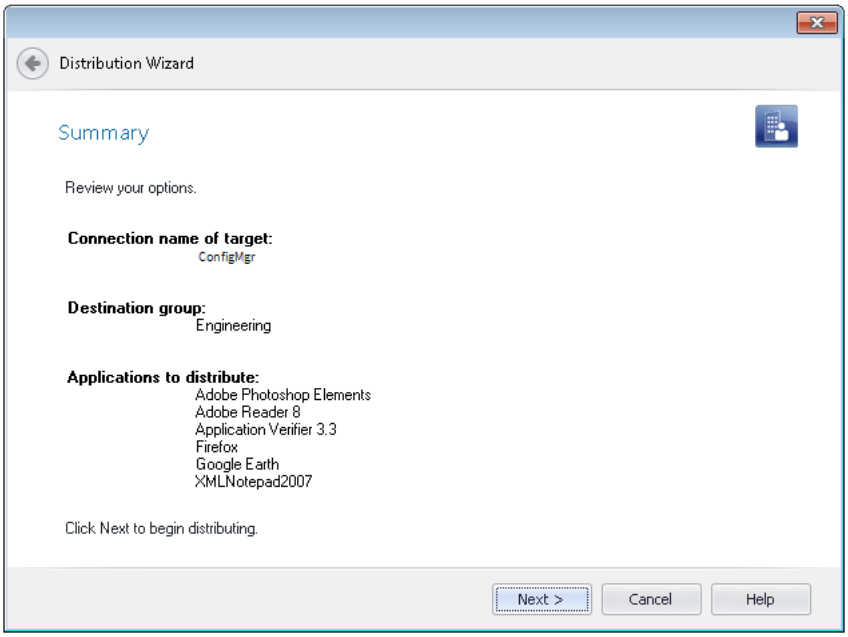


Figure 21-1: Distribution Wizard / Summary Panel

Distributing Panel

The **Distributing** panel displays a progress bar and status messages during distribution. When distribution is complete, click **Finish**.

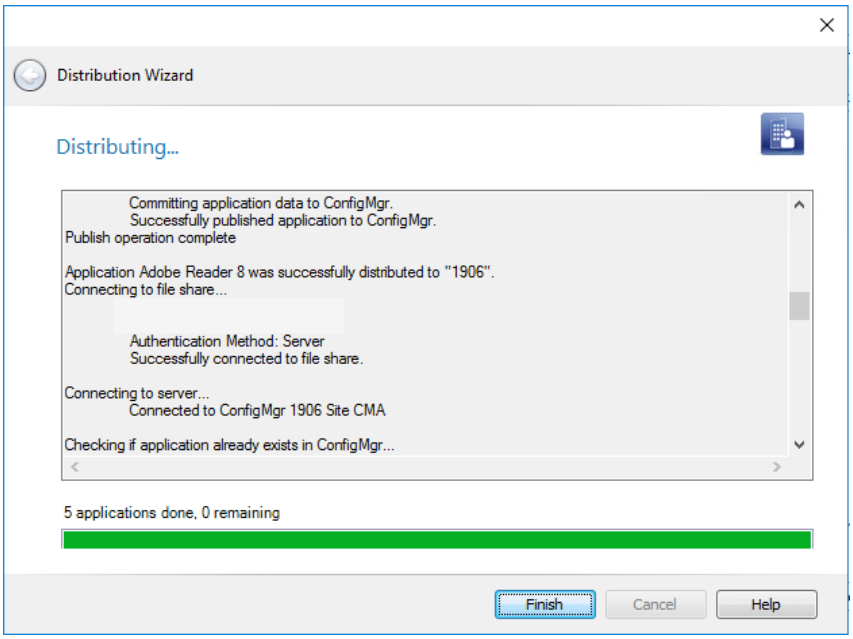


Figure 21-2: Distribution Wizard / Distributing Panel

Package Distribution Wizard

The Package Distribution Wizard is used to distribute packages to Microsoft System Center 2007 Configuration Manager, a network location, an FTP server, an administrative location, or using virtually any distribution system.

You can launch the Package Distribution Wizard from Application Manager by selecting a package in the tree and then clicking the **Distribute** button in the **Home** tab of the ribbon or by right-clicking on a package and selecting **Distribute Package** from the shortcut menu. You can also launch the Package Distribution Wizard from the Windows Start menu.

The Package Distribution Wizard consists of the following panels:

- Welcome Panel
- Distribution Type Panel
  - Administrative Install Panel
  - Connect to a ConfigMgr Server Panel
  - Select Destination Folder
  - Select Group
  - FTP Location Panel
  - Altiris Integration Panel
  - Network Location Panel

- Network Location Panel
- Package Information Panel
- Distribution Summary Panel
- Distribution Output Panel

Welcome Panel

The Package Distribution Wizard is used to distribute packages to ConfigMgr, a network location, an FTP server, an administrative location, or several other distribution systems.

You can launch the Package Distribution Wizard from Application Manager by selecting a package in the tree and then clicking the **Distribute** button in the **Home** tab of the ribbon or by right-clicking on a package and selecting **Distribute Package** from the shortcut menu. You can also launch the Package Distribution Wizard from the Windows Start menu.

Click **Next** to continue.

Distribution Type Panel

From the **Distribution Type** panel, you can select the distribution method you want to use. You can choose one of the following distribution methods:

Table 21-6 • Package Distribution Wizard Distribution Types

Method	Description
ConfigMgr	The Package Distribution Wizard will publish the selected package to ConfigMgr. See Publishing Packages to ConfigMgr (Formerly called as System Center Configuration Manager).
ZENworks Configuration Management Distribution	Create an MSI distribution object to distribute on ZENworks Configuration Management. See Preparing for ZENworks Configuration Management Distribution.
Altiris 6.5	Create a package in Altiris 6.5 Notification Server. See Preparing for Altiris 6.5 Distribution.
FTP Location	Distribute to an FTP server, providing both your user name and password. See Distributing Packages to FTP Servers.
Network Location	Distribute into a network directory. See Distributing Packages to Network Locations.
Administrative Install	The installation is copied to a network directory using the administrative install option provided by Windows Installer. See Creating Administrative Installations for Packages.

Click **Next** to proceed to the associated distribution method panel.

Administrative Install Panel

This panel is displayed if you selected the **Administrative Install** distribution method from the **Distribution Type** panel. The installation will be copied to the network directory using the Windows Installer administrative install option.

The **Administrative Install** panel includes the following properties:

Table 21-7 • Administrative Install Panel

Property	Description
Network Directory	Specify or browse to the network location where you want to perform the installation.
Use short file names	Select this option to force the administrative installation to use the 8.3 file name convention (using the SHORTFILENAMES property).

Connect to a ConfigMgr Server Panel

On this panel, which is displayed if you selected **ConfigMgr** on the **Distribution Type** panel, you are prompted to enter connection information for your ConfigMgr server.

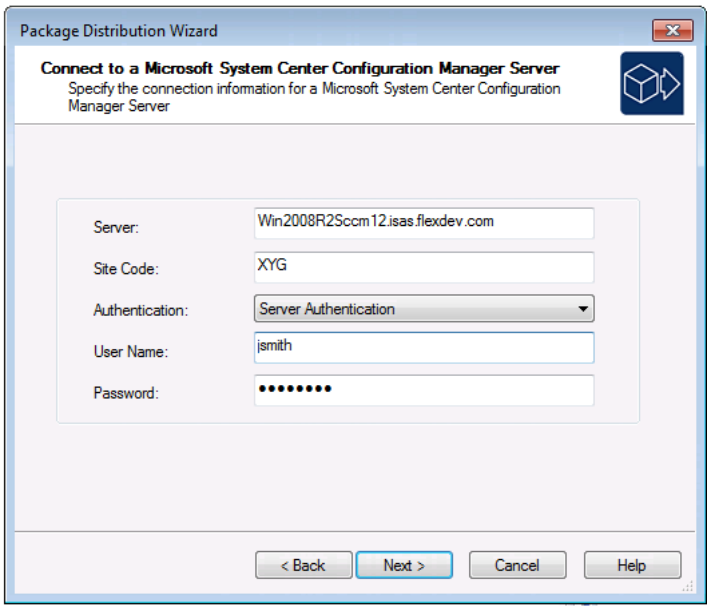


Figure 21-3: Package Distribution Wizard Connect to a ConfigMgr Panel

Enter the following information and click **Next** to continue.

Table 21-8 • Connect to a ConfigMgr Server Panel

Property	Description
Server	Enter the name of the ConfigMgr server that you want to connect to. ***Note*** • *Using the Package Distribution Wizard, you can publish Windows Installer packages and legacy setups to both System Center 2007 Configuration Manager and System Center 2012 Configuration Manager. However, the Package Distribution Wizard can only publish App-V 4.x packages to System Center 2007 Configuration Manager* ***Important*** • *To publish **applications** to System Center 2012 Configuration Manager, you need to use the application-based Distribution Wizard, as described in Distributing Applications Using the Distribution Wizard.*
Site Code	Enter the code that identifies the ConfigMgr site you want to connect to.
Authentication	From this list, select one of the following options: • **Windows Authentication**—Select if you want to use the credentials of the logged in user to login to the server. • **Server Authentication**—Select if you want to connect to the server using the specified **User Name** and **Password**.

Select Destination Folder

On this panel, which is displayed if you selected **ConfigMgr** on the **Distribution Type** panel, you are prompted to select a location that the ConfigMgr server has access to where you want to publish the selected packages.

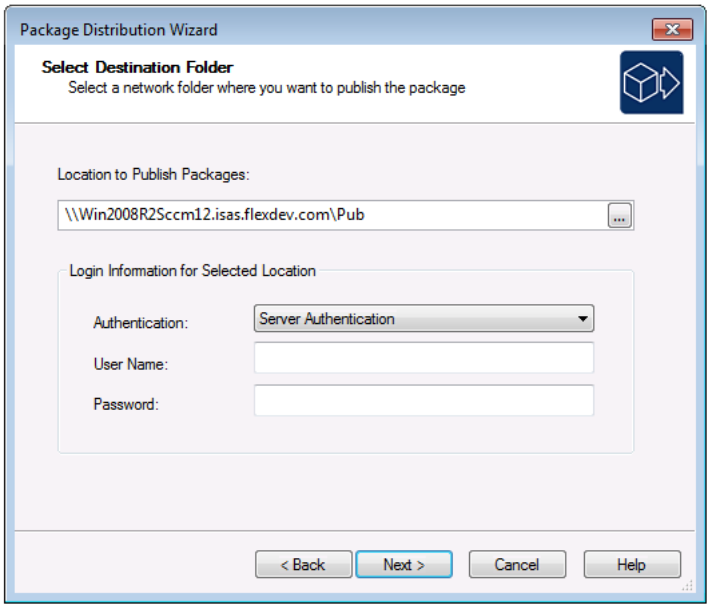


Figure 21-4: Package Distribution Wizard Select Destination Folder Panel

Enter the following information and click **Next** to continue.

Table 21-9 • Select Destination Folder Panel

Property	Description
Location to Publish Packages	Enter a target path, in UNC format (\\Server\Share), of the location where you want to publish the selected packages. Make sure that you enter a location that the ConfigMgr server has access to.
Authentication	From the **Authentication** list, select one of the following options: • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this location. • **ConfigMgr Authentication**—Choose this option if you want to use ConfigMgr server authentication (your ConfigMgr server login ID) to log into this location. • **Server Authentication**—Choose this option if you are publishing to an alternate file server that requires credentials. Then enter the appropriate **User Name** and **Password**.

Select Group

On this panel, which is displayed if you selected **ConfigMgr** on the **Distribution Type** panel, select the **Target Group** on the ConfigMgr Server where you want to publish the package and click **Next** to continue.

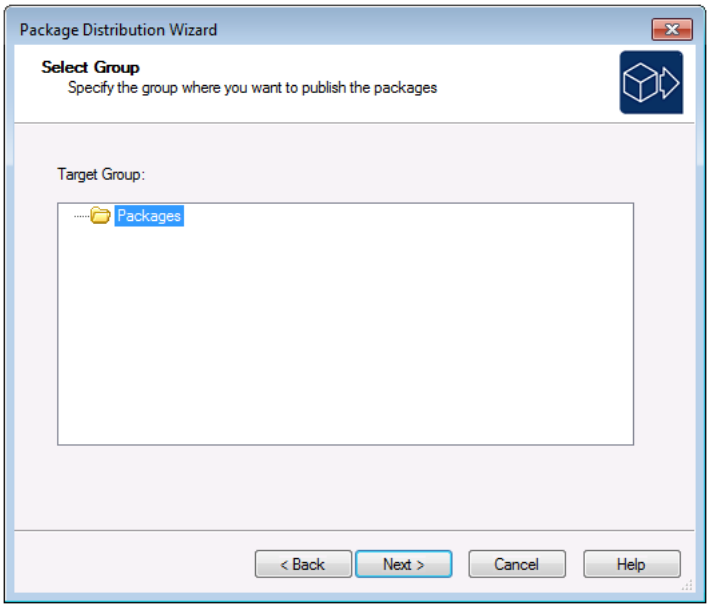


Figure 21-5: Package Distribution Wizard Select Group Panel

FTP Location Panel

This panel is displayed when you select FTP Location as the distribution method from the **Distribution Type** panel. The installation will be upload to the FTP server specified in the **FTP Location** field. If necessary, provide a **User Name** and **Password** for the FTP server.

Altiris Integration Panel

This panel is displayed if you selected the **Altiris 6.5** distribution method from the **Distribution Type** panel.

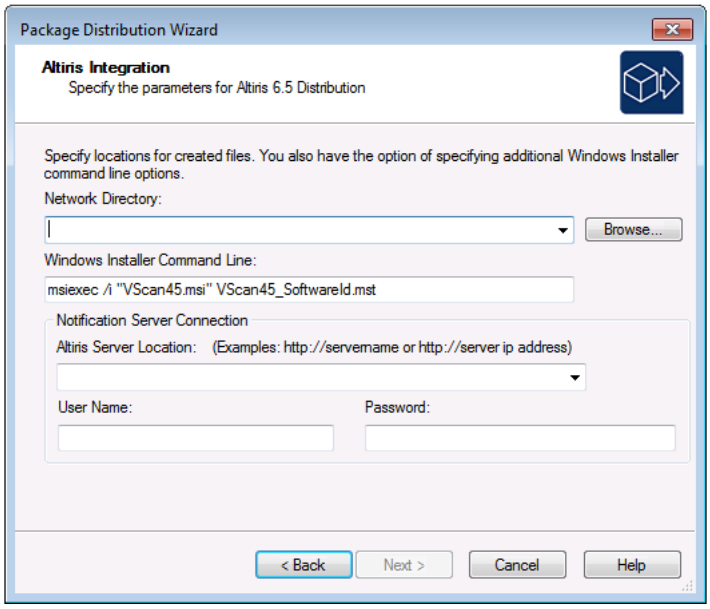


Figure 21-6: Distribution Wizard / Altiris Integration Panel

You would choose the Altiris method to create a package in the Altiris Notification Server.

Table 21-10 • Altiris Integration Panel

Field	Description
Network Directory	Specify or browse to the network location where you want to store the installation package. The Distribution Wizard remembers the last Network Location that is entered and displays it the next time this panel is accessed. The Distribution Wizard will copy the Windows Installer package along with any transforms and files to the UNC path specified. Also, the Distribution Wizard will use an XML template file (AltirisTemplate.config) to create a custom script file in this location named <packageName>.Config. ***Note*** • *You can edit AltirisTemplate.config to customize it for your organization. The file, which is installed with AdminStudio, is located in the Templates folder of the AdminStudio Shared directory. See Altiris XML Template for more information.*
Windows Installer Command Line	Enter any additional properties that you want to pass to the Windows Installer. See the Windows Installer Property Reference for more information.
Altiris Server Location	Enter the http: address for the location of the Altiris Server. The Distribution Wizard remembers the last Altiris Server Location that is entered and displays it the next time this panel is accessed.
User Name	Enter a User Name to log onto the server entered in the Altiris Server Location field. The Distribution Wizard remembers the last User Name that is entered and displays it the next time this panel is accessed.

Table 21-10 • Altiris Integration Panel

Field	Description
Password	Enter the password to log onto the server entered in the Altiris Server Location field.

Click **Next** to proceed to the **Package Information** panel.

Altiris XML Template

When using the Altiris distribution method, a custom script file is required. If you select the Altiris method on the Distribution Type Panel, the **Distribution Wizard** uses an XML Template file (`AltirisTemplate.config`) to create a custom script file named `<packageName>.Config`. The Distribution Wizard copies this configuration file along with the Windows Installer package with any transforms and files to the Network Directory specified on the Altiris Integration Panel.

You can edit `AltirisTemplate.config` to customize it for your organization. The file, which is installed with AdminStudio, is located in the `Templates` subdirectory of the following directory:

`[AdminStudioInstallDirectory]\AdminStudio Shared`

The following variables are used in the `AltirisTemplate.Config` file:

Table 21-11 • AltirisTemplate.config Variables

Variable	Value
%DIST.ASVERSION%	AdminStudio version number
%DIST.COMMANDLINE%	Command line specified on the Altiris Integration Panel of the Distribution Wizard. See the Windows Installer Property Reference for more information.
%DIST.NETWORKLOCATION%	Network location of the MSI package as specified on the Altiris Integration Panel of the Distribution Wizard
%ProductCode%	ProductCode property from the MSI Property Table
%ProductName%	ProductName property from the MSI Property Table
%ProductVersion%	ProductVersion property from the MSI Property Table
%SUMMARYSTREAM.Id%	Comments property from the MSI Summary Stream. Any property with the SUMMARYSTREAM prefix will be populated based on the MSI Summary Information Stream Property as specified by the Id, in the format of: `<description>%SUMMARYSTREAM.4%</description>` Summary Stream Ids range from 1 to 19. For a complete list of Summary Information Stream Ids, see Summary Information Stream Property Set. In the example above, "4" indicates that the value of the Author property should be inserted.

Please note the following:

- Any property with the DIST prefix will be custom populated by the Distribution Wizard.
- Any other property will be populated based on the MSI Property Table.
- Typically, all variables are enclosed within '%' characters, as shown above.

Network Location Panel

This panel is displayed when you choose the **Network Location** distribution method from the **Distribution Type** panel. The installation files will be copied to the network directory you specify (or browse to) in this panel.

Package Information Panel

On the **Package Information** panel, you select the package that is ready for distribution.

Table 21-12 • Package Information Panel Options

Option	Description
Windows Installer Package (.msi) OR **Windows Installer Package (.msi) / Microsoft App-V Package (.sft) / Legacy Setup Package (.exe)**	Specify or browse to the package that you want to distribute: • **If distributing to ConfigMgr**—You can select a Windows Installer (`.msi`), App-V 4.x (`.sft`), or legacy setup (`.exe`) package. • **All other distribution types**—You can select only a Windows Installer (`.msi`) package. If you launched the Package Distribution Wizard from the Application Manager by right-clicking on a package and selecting **Distribute Package** from the shortcut menu, the package name in this field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository: • **Not in the Software Repository**—The full name and path of the file is displayed, and you can edit this entry or click **Browse** and select a different package. • **In the Software Repository**—Only the name of the file is displayed (not the full path) and this entry cannot be edited or changed. ***Edition*** • *The Software Repository is included in AdminStudio Enterprise Edition.*
Windows Installer Transform Files (*.mst)	(Windows Installer packages only) In this area: • If there are transforms associated with the package, click the New button ( ) and navigate to the transform you want to add. • Use the Up and Down arrows ( ) to set the order in which the transforms are applied to the package. • Use the Delete button ( ) to delete a transform from the list.

Table 21-12 • Package Information Panel Options (cont.)

Option	Description
Specify Additional MSI Properties	(Windows Installer packages only) If desired, enter additional Windows Installer properties.

Click **Next** to proceed.

Distribution Summary Panel

The **Distribution Summary** panel displays a summary of all settings configured in the previous panels. When you click **Next**, the distribution begins and the **Distribution Output** panel is displayed.

Caution • *The distribution will overwrite the contents of the distribution folder.*

Distribution Output Panel

The **Distribution Output** panel displays a progress bar and status messages during distribution. When distribution is successful, a message appears and the background color of the output window turns green. If errors are encountered, the background color of the output window turns red.

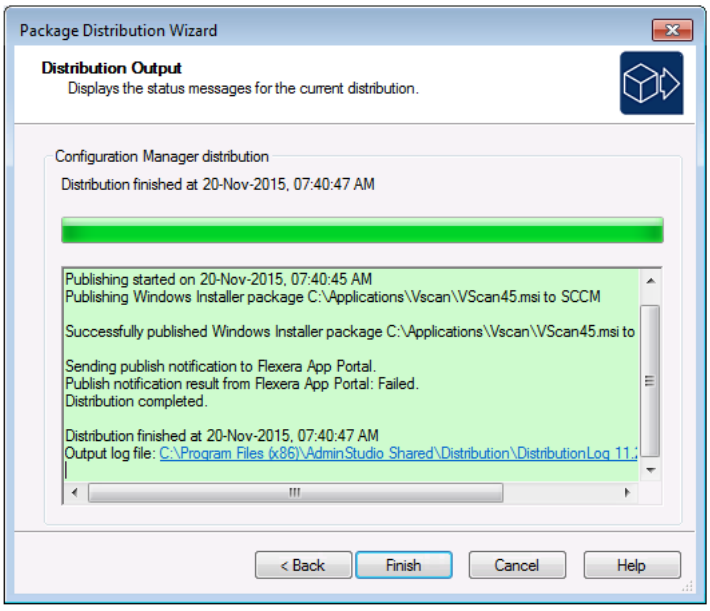


Figure 21-7: Distribution Output Panel

Click **Finish** to exit the Wizard.

Note • *For all distribution types, the Package Distribution Wizard will create a Distribution log file in the* `Distribution` *folder of the* `AdminStudio Shared` *directory.*

Distribution Wizard for ZENworks Configuration Management

Microfocus ZENworks Configuration Management 10 and 11 customers can use the Distribution Wizard for ZENworks Configuration Management to distribute a Windows Installer package (`.msi`)—including any associated transforms—to ZENworks Configuration Management.

The Distribution Wizard for ZENworks Configuration Management consists of the following panels:

- Welcome Panel
- Login Panel
- Windows Installer Package Information Panel
- Bundle Creation Options Panel
- Bundle Information Panel
- Summary Panel
- Publishing Process Panel

Welcome Panel

You can use the Distribution Wizard for ZENworks Configuration Management to prepare a Windows Installer package (`.msi`)—including any associated transforms—for distribution on ZENworks Configuration Management.

If you do not want this panel to be displayed each time you open this wizard, select the **Do not show the Welcome panel again** option. If this option is selected, the **Login** panel will be the first panel opened for this wizard.

Click **Next** to continue.

Login Panel

On the **Login** panel, enter the login information for the ZENworks Configuration Management server that you want to distribute packages on, and click **Login** to proceed with the distribution process.

Enter the following information:

Table 21-13 • ZENworks Configuration Management Server Login Information

Property	Description
User Name	Enter a valid User Name for the ZENworks Configuration Management server where you want to distribute packages
Password	Enter a valid Password for the ZENworks Configuration Management server where you want to distribute packages

Table 21-13 • ZENworks Configuration Management Server Login Information (cont.)

Property	Description

Server URL	Enter the server URL, machine name, or IP address of the ZENworks Configuration Management server using the following format:
	`http://www.servername.com or http://111.22.333.44`
	If you need to specify a specific port number, append the port number to the end of the URL, such as:
	`http://www.servername.com:123`
	If you are using SSL and you want a secure connection, change the `http` prefix to `https`. For example:
	`https://www.servername.com`

Windows Installer Package Information Panel

On the **Windows Installer Package Information** panel, enter the information that will be referenced by this ZENworks server bundle, and click **Next** to continue.

Enter the following information:

Table 21-14 • Windows Installer Package Information Panel Properties

Property	Description
Windows Installer Package file (.msi)	Click **Browse** and select the Windows Installer (`.msi`) package that you want to distribute.
Windows Installer Transform Files (.mst)	All of the `.mst` files that are in the same directory as the selected Windows Installer package are automatically listed in this list. To include transforms with the Windows Installer package, click the New button ( ) and select a transform. If the package requires multiple transforms, you can repeat the procedure as necessary. Use the Delete button ( ) to delete the selected transform from the list. ***Note*** *• All of the `.mst` files that are in the same directory as the selected Windows Installer package are automatically listed in this list even if they are not applicable to the selected package. To prevent the inclusion of non-applicable transform files, delete those transforms from the list.*

Table 21-14 • Windows Installer Package Information Panel Properties (cont.)

Property	Description
Install Parameters	You can customize how this package is installed by entering parameters in this field. These parameters are applied to `msiexec.exe` to perform the desired action. Any actions that you enter here will be performed whenever the bundle is installed. The root parameter, which should not be edited or deleted, is: `/i packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** *• If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* ***Note*** *• For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*
Uninstall Parameters	Enter an action that will be performed whenever the bundle is uninstalled. The root parameter, which should not be edited or deleted, is: `/x packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** *• If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* ***Note*** *• For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*

Table 21-14 • Windows Installer Package Information Panel Properties (cont.)

Property	Description
Repair Parameters	Enter an action that will be performed whenever the user chooses to repair the bundle by repairing or reinstalling missing or corrupted files. The root parameter, which should not be edited or deleted, is: `/f packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** *• If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* You can also apply any of the following additional parameters after the package name: ● **p** – Reinstalls a file if it is missing ● **o** – Reinstalls a file if it is missing or if an older version of the file is present on the user's system ● **e** – Reinstalls a file if it is missing or if an equivalent or older version of the file is present on the user's system ● **c** – Reinstalls a file if it is missing or if the stored checksum of the installed file does not match the new file's value ● **a** – Forces a reinstall of all files ● **u** or **m** – Rewrite all required user registry entries ● **s** – Overwrites any existing shortcuts ● **v** – Runs your application from the source and re-caches the local installation package ***Note*** *• For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*

Additional Install, Uninstall, and Repair Parameters

The following additional parameters can be entered in the Parameters fields.

Table 21-15 • Additional Parameters

Parameter	Description

`/j [u|m] packagename.msi`
`/j [u|m] packagename.msi /t <transform list>`
`/j [u|m] packagename.msi /g`
`/j <language ID>`
	Building with the `/j <package>` option advertises the components of your application on the end user's computer
	• **u** – Advertises components only to the current user
	• **m** – Advertises components to all users of the computer
	• **g** – Specifies language ID
	• **t** – Applies a transform to your advertised product
	Transforms allow the synchronization of applications across different languages. For example, if you upgrade the English version of your product, you could apply a transform to automatically upgrade the French version of your product.

`/L [i|w|e|a|r|u|c|m|p|v|+] <log file>`	Building with the `/L` option specifies the path to the log file. These flags indicate which information to record in the log file: • **i** – Logs status messages • **w** –Logs non-fatal warning messages • **e** – Logs any error messages • **a** – Logs the commencement of action sequences • **r** – Logs action-specific records • **u** – Logs user requests • **c** – Logs initial user interface parameters • **m** – Logs out-of-memory messages • **p** – Logs terminal settings • **v** – Logs the verbose output setting • **+** – Appends to an existing file • ***** – Is a wildcard character that allows you to log all information (excluding the verbose output setting)

Table 21-15 • Additional Parameters (cont.)

Parameter	Description
`/q [n|b|r|f]`	The `/q` option is used to set the user interface level in conjunction with the following flags: • **q** or **qn** – Creates no user interface • **qb** – Creates a basic user interface The user interface settings below display a modal dialog box at the end of installation: • **qr** – Displays a reduced user interface • **qf** – Displays a full user interface • **qn+** – Displays no user interface • **qb+** – Displays a basic user interface
`/y <filename>`	This command calls the `DllRegisterServer` entry-point function of the DLL or OCX file specified in `<filename>`.
`/z <filename>`	This command calls the `DllUnregisterServer` entry-point function of the DLL or OCX file specified in `<filename>`.
`TRANSFORMS`	Use the `TRANSFORMS` command-line parameter to specify any transforms that you would like applied to your base package. Your transform command-line call might look something like this: `msiexec /i "C:\Directory\ProductName.msi" TRANSFORMS="New Transform 1.mst"` You can separate multiple transforms with a semicolon. Because of this, it is recommended that you do not use semicolons in the name of your transform, as the Windows Installer service will interpret those incorrectly.
`Properties`	All public properties can be set or modified from the command line. Public properties are distinguished from private properties by the fact that they are in all capital letters. For example, `COMPANYNAME` is a public property. To set a property from the command line, use the following syntax: `PROPERTY=VALUE`. If you wanted to change the value of `COMPANYNAME`, you would enter: `msiexec /i "C:\Directory\ProductName.msi" COMPANYNAME="YourCompany"`

Bundle Creation Options Panel

On the **Bundle Creation Options** panel, specify whether you want to create a new bundle or overwrite an existing bundle. After making your selection, click **Next** to proceed.

You have the following options:

Table 21-16 • Bundle Creation Options

Option	Description
Create a new bundle from these Windows Installer package files	To create a new bundle to reference this Windows Installer package, select this option.
Update an existing bundle using these Windows Installer package files	If you want to overwrite an existing bundle to reference this Windows Installer package, select this option, and then select an existing bundle in the tree: • **Recommended Bundles**—This group lists the bundles that contain the same Windows Installer package as the one you selected on the Windows Installer Package Information Panel. • **All Other Bundles**—This group lists the rest of the existing bundles on the server.

Bundle Information Panel

On the **Bundle Information** panel, enter information to specify attributes for this bundle on ZENworks Configuration Management, and click **Next** to continue.

Enter the following properties:

Table 21-17 • Bundle Information Panel Properties

Property	Properties
Bundle Name	Enter the bundle's name as you want it to appears in ZENworks® Control Center (ZCC) and the ZENworks Application Launcher (on managed devices).
Version Number	Enter the bundle's version number. If you are overwriting an existing bundle, and you enter a higher version number than the bundle's original version number, the bundle will be redeployed.
Icon	Click **Browse** and select a shortcut icon graphic (in `.ico`, `.gif`, or `.jpg` format) that ZENworks Application Launcher will display on managed devices. If you do not select an icon file, the standard ZENworks bundle icon will be used.

Table 21-17 • Bundle Information Panel Properties (cont.)

Property	Properties
Folder	From the **Folder** list, select the folder path that will be used by ZENworks Application Launcher when displaying the bundle on either the device's desktop or **Start** menu. All of the folders defined on the ZENworks server are listed. For example: • **Start Menu**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the **Start** menu, ZENworks Application Launcher creates an `Application\Accounting` folder on the root of the **Start** menu and adds the bundle to it. • **Desktop**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the desktop, ZENworks Application Launcher creates an `Applications\Accounting` folder on the desktop and adds the bundle to it. You can place multiple bundles in a single folder by specifying the same folder path for each of the bundles.
Description	Enter a description of the bundle. This description will be displayed in ZENworks® Control Center and the ZENworks Application Launcher (on managed devices).

Summary Panel

The **Summary** panel displays the options you have selected for distributing this Windows Installer package on ZENworks Configuration Management.

Click **Publish** to complete the distribution process or **Back** to change the listed options.

Publishing Process Panel

The **Publishing Process** panel lists the progress messages while the bundle is being published on ZENworks Configuration Management.

- **ZENworks error messages**—Any error messages with a numeric prefix that appear on this panel are generated by ZENworks Configuration Management. To resolve these errors, contact your ZENworks Configuration Management System Administrator.
- **Canceling publication**—If you want to cancel the publication of the bundle on ZENworks Configuration Management, click **Cancel**.
- **Exiting the wizard**—When processing is complete, the **Finish** button becomes enabled. Click **Finish** to exit this wizard.

22

Generating and Viewing Reports

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

Reports provides reporting capability for both AdminStudio and Workflow Manager. You can use Reports to generate reports on packages stored in the Application Catalog, and on Workflow Manager projects and workflow requests, using customized SQL queries or stored procedures.

Table 22-1 • AdminStudio and Workflow Manager Reports in Reports

Product	Available Reports
AdminStudio Reports	Reports provides a centralized view of all of the information regarding packages in your Application Catalog. See Generating and Viewing AdminStudio Reports. • **Package Reports**—Includes detailed information on individual packages in the Application Catalog. See Viewing Package Reports. • **Custom SQL Query Report**—A custom report defined by entering an SQL query in the Report Wizard. See Generating a Custom SQL Query Report for AdminStudio. • **Custom Stored Procedure Report**—A custom report on data generated by AdminStudio or Workflow Manager that is defined by specifying a stored procedure in the Report Wizard. See Generating a Custom Stored Procedure Report for AdminStudio. • **AdminStudio Application Catalog Reports**—View a wide array of reports containing summary information on the Windows Installer, App-V, and iOS and Android applications in your Application Catalog. See Viewing AdminStudio Application Catalog Reports.

Table 22-1 • AdminStudio and Workflow Manager Reports in Reports (cont.)

Product	Available Reports
Workflow Manager Reports	You can view System Reports that include information on projects and requests. You can also define custom reports that include information about the status of projects and requests. See Generating and Viewing Workflow Manager Reports. • **System Reports**—Includes detailed summary information on a company's projects and requests. See Generating Standard Reports. • **Custom Report**—A report defined by using the Report Wizard. See Creating a Custom Report. • **Custom Activity Report**—Every time an activity or event occurs during the completion of a request, Workflow Manager records that activity. You can view a listing of these activities in the Activity Report, a custom report which you define using the Report Wizard. See Creating an Activity Report • **Custom SQL Query Report**—A custom report defined by entering an SQL query in the Report Wizard. See Generating a Custom SQL Query Report. • **Custom Stored Procedure Report**—A custom report on data generated by Workflow Manager that is defined by specifying a stored procedure in the Report Wizard. See Generating a Custom Stored Procedure Report.

Generating and Viewing AdminStudio Reports

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use Reports to obtain a centralized view of all of the information regarding packages in your AdminStudio Application Catalog. Because Reports is a Web application, it can be easily accessed by a geographically dispersed workforce without requiring any software installation or data transfer. Reports makes it easy to get the application data you need to diagnose and repair software problems and to manage applications across your organization.

A catalog-level search tool enables you to generate detailed, custom reports on packages with particular characteristics. These reports are accessible anywhere via a Web interface and can be exported to PDF or Excel format for sharing and archiving.

Information on generating and viewing AdminStudio reports in Reports is presented in the following sections:

Table 22-2 • Information About Generating AdminStudio Reports

Section	Description
Viewing Package Reports	Explains how to generate a Package Report on a selected package. Also explains how to filter the package tree by specified criteria in order to find a specific package in the Application Catalog. This section also lists the contents of all of the sections of a Package Report.

Table 22-2 • Information About Generating AdminStudio Reports

Section	Description
Generating a Custom SQL Query Report for AdminStudio	Explains how to enter an SQL query to specify the data to be displayed in a custom report.
Generating a Custom Stored Procedure Report for AdminStudio	Explains how to generate a custom report on data generated by AdminStudio or Workflow Manager that is defined by specifying a stored procedure in the Report Wizard.
Viewing AdminStudio Application Catalog Reports	Explains how to view a wide array of reports containing Application Catalog summary information on Windows Installer, App-V, and iOS and Android applications in the Application Catalog.

Viewing Package Reports

Edition *• Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

Note *• In Workflow Manager, Package reports can be viewed by users in both Administrator and Consumer companies.*

You can generate AdminStudio Package Reports on the **Search Packages** page, which is opened by clicking **Search Packages** on the **Reports** menu of the navigation bar.

On the **Search Packages** page you can perform a search of all of the applications in the Application Catalog to locate the package you would like to generate a report for.

- Searching for a Package on the Search Packages Page
- Information Included in Package Reports
- Navigating Through a Package Report
- Archiving a Package Report
- Exporting a Package Report

Searching for a Package on the Search Packages Page

Edition *• Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

By default, all of the packages in the connected AdminStudio Application Catalog are listed on the **Search Packages** page. However, you can filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria. The search criteria are grouped into three categories:

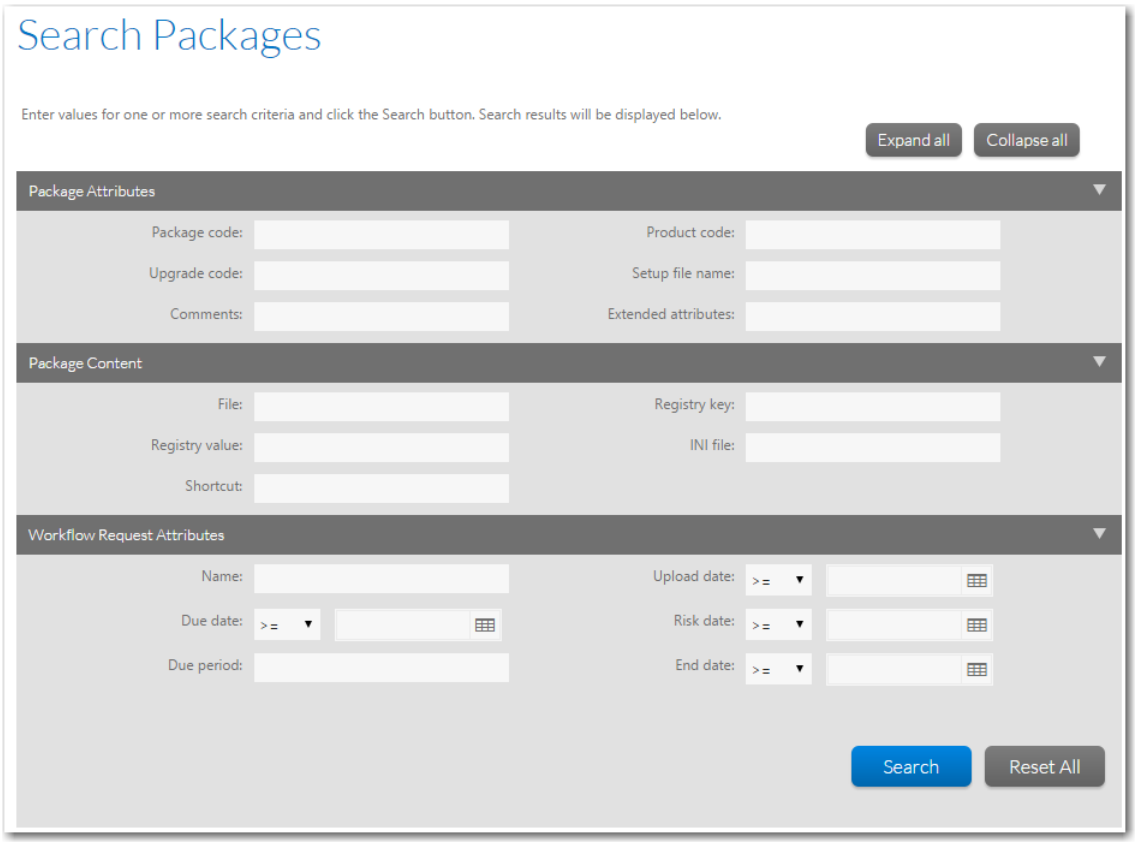
- **Package Attributes**—Search by common properties assigned to packages. See Package Attributes.

- **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the package. See Package Content.
- **Workflow Request Attributes**—Search by information related to a package's associated workflow request. See Workflow Request Attributes.

To filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, perform the following steps.

Task ***To search for a package on the Search Packages page:***

1. In the **Search Packages** area of the **Search Packages** page, expand the criteria category that you want to use by clicking the arrow. When all three categories are expanded, the following fields are available:



2. Enter values in the criteria fields that you want to search on. You can search for packages in the Application Catalog based on metadata in three categories:
   - **Package Attributes**—Search by properties assigned to the package. See Package Attributes.
   - **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the package. See Package Content.
   - **Workflow Request Attributes**—Search by information related to a package's associated workflow request. See Workflow Request Attributes.
3. After you have entered the search criteria, click **Search**. The packages that meet the criteria are now listed.

Package Attributes

You can search for packages in a catalog based on one or more of any of the following package attribute metadata:

Table 22-3 • Package Attribute Search Fields

Metadata	Description
Package Code	Enter the GUID that identifies a particular Windows Installer `.msi` package. The Package Code associates an `.msi` file with an application or product and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F).
Product Code	Enter the GUID that uniquely identifies the particular product release of a package. The ProductCode is a Windows Installer property and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F).
Upgrade Code	Enter the GUID that identifies the family of products that are in the same upgrade path. The **UpgradeCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXsXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F). ***Note*** • *Each stand-alone product usually has its own **UpgradeCode** GUID. Every version of XYZ Product typically uses the same GUID for the **UpgradeCode**. In other words, Product A Version 1.0 has the same UpgradeCode as Product A Version 2.0, but has a different UpgradeCode than Product B.*
Setup File Name	Name of the file that was imported into the Application Catalog.
Comments	Enter the text of any comments associated with the package.
Extended Attributes	Enter the value of any of the Extended Attributes associated with the package.

Package Content

You can search for packages in a catalog based on one or more of any of the following Package Content metadata

Table 22-4 • Package Content Search Fields

Metadata	Description
File	Enter the file name of one of the files in the package.

Table 22-4 • Package Content Search Fields (cont.)

Metadata	Description
Registry Key	Enter a registry key to search on.
Registry Value	Enter a registry value to search on.
INI File	Enter any changes to an `.ini` file that are made when the product is installed.
Shortcut	Enter the name of a shortcut that is created when the product is installed.

Workflow Request Attributes

You can search for packages in a catalog based on one or more of any of the following attributes of the package's associated workflow request:

Table 22-5 • Workflow Request Attributes Search Fields

Metadata	Description
Name	Enter the name of the package's associated workflow request.
Upload Date	Enter the date the workflow request was created.
Due Date	Enter the date the workflow request is scheduled to be completed, based upon its value for **Application Due Period**.
Risk Date	Enter the date at which the workflow request's status will change to **At Risk**, which is based upon its value for **Application At Risk Period**.
Due Period	Enter, in days, the length of time this workflow request needs to be completed in order to meet its project's Service Level Agreement (SLA) requirements.
End Date	Enter the date the workflow request was completed.

Information Included in Package Reports

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

A Package Report lists detailed package information for packages of the following deployment types:

- Microsoft Windows Installer packages
- Microsoft App-V virtual packages
- Apple iOS mobile apps (local and public store)
- Google Android mobile apps (local and public store)

In a Package Report, the information is presented in a tabbed interface, as described in Navigating Through a Package Report. A Package Report includes the following major sections:

- Package Summary Information View
- Files View
- Registry View
- Shortcuts View
- ODBC Drivers View
- ODBC DS View
- Extended Attributes View
- Validation View
- Conflicts View
- History View
- Dependencies View
- Properties View

Package Reports for mobile apps only include the Files View, Properties View, and History View.

Package Summary Information View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The initial view (Page 1) of a Package Report is the **Package Summary Information** view.

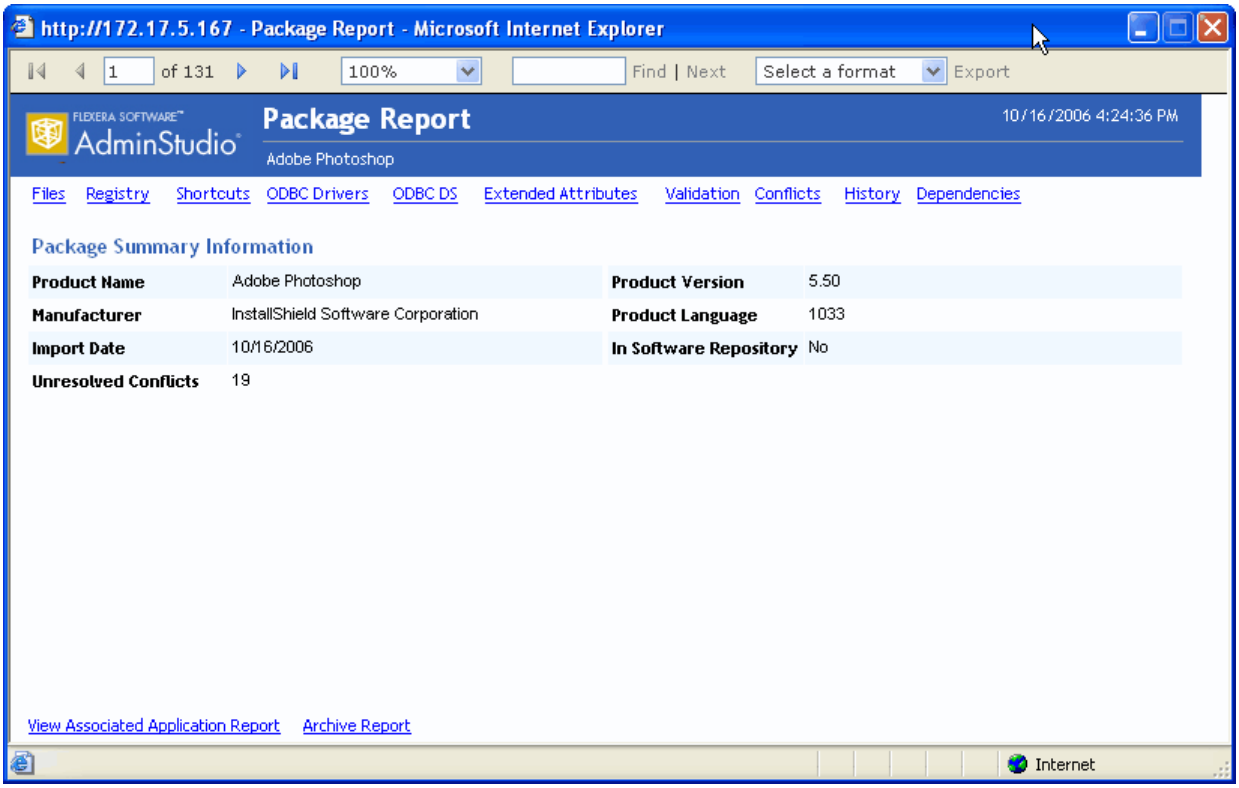


Figure 22-1: Package Report / Package Summary Information View

The Package Summary Information View lists the following information:

Table 22-6 • Package Report / Package Summary Information

Item	Description
Product Name	Name assigned to the package.
Manufacturer	Company that authored the package.
Import Date	The date and time the package was imported into the Application Catalog.
Unresolved Conflicts	The number of detected conflicts, generated during conflict analysis of this package, which have not yet been resolved—either automatically or manually.
Product Version	Version of package that is recorded in the package's Windows Installer file.
Product Language	Decimal-based code identifying the language that this software package was authored for. For example, English is 1033, German is 1031, and Japanese is 1041.
In Software Repository	Indicates whether or not this package and its associated files are managed by the Software Repository.

Files View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Files** view lists all of the files included in the selected package, and the location where these files will be installed.

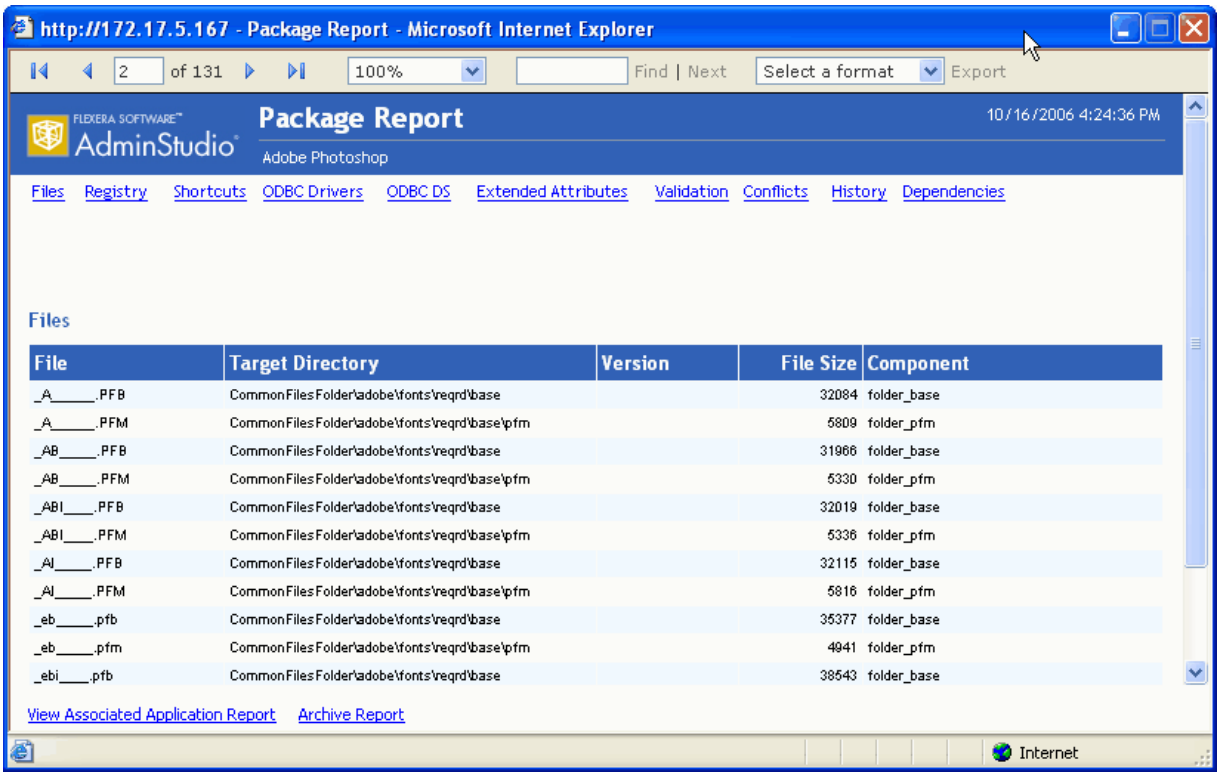


Figure 22-2: Package Report / Files View

For each file, the following information is listed:

Table 22-7 • Package Report / Files Information

Item	Description
File	Name of file included with this package.
Target Directory	Name of directory where the file is installed.
Version	Version number of the file.
File Size	Size of the installed file.
Component	Component that the file is associated with.

Registry View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Registry** view lists the registry entries that will be created when this package is installed.

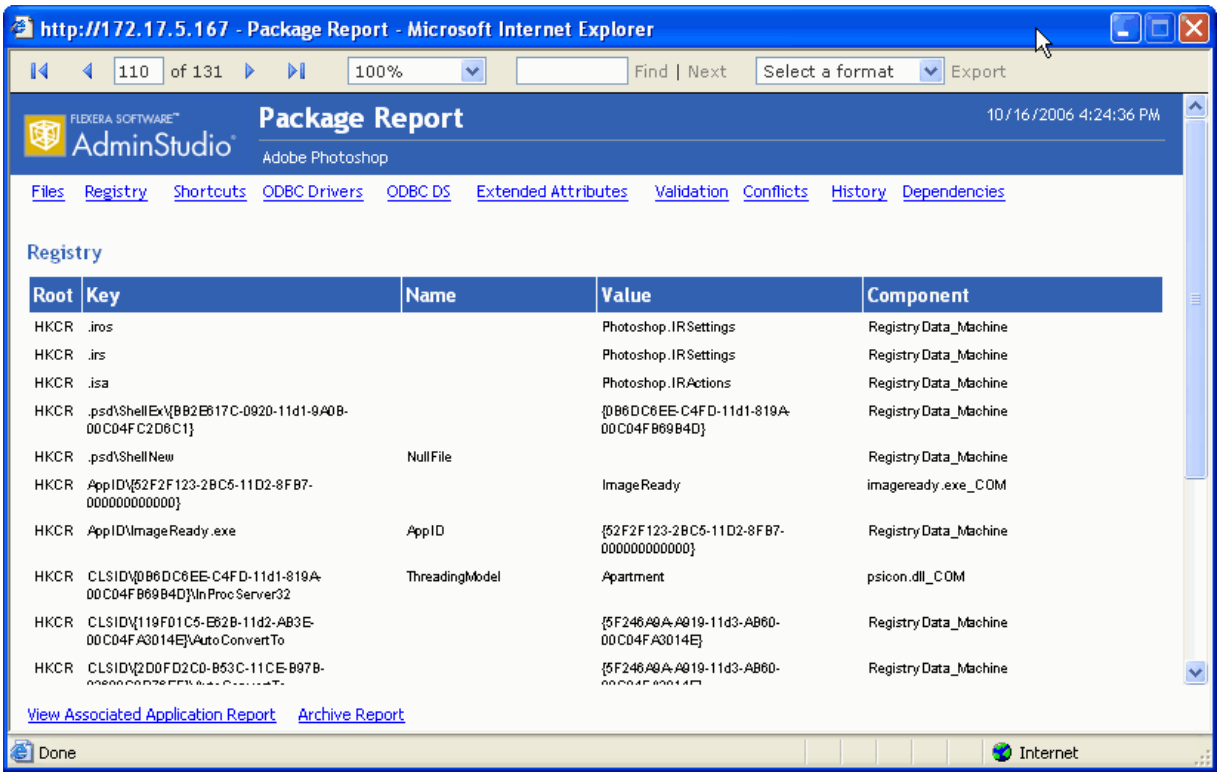


Figure 22-3: Package Report / Registry View

For each registry entry, the following information is listed:

Table 22-8 • Package Report / Registry Information

Item	Description
Root	Identifies the predefined “root” key that contains the registry entry.
Key	A registry key.
Name	Name identifying the registry entry.
Value	The string of data that defines the value of the key.
Component	Package component that the registry entry is associated with.

Shortcuts View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Shortcuts** view lists all of the shortcuts that will be created when this package is installed.

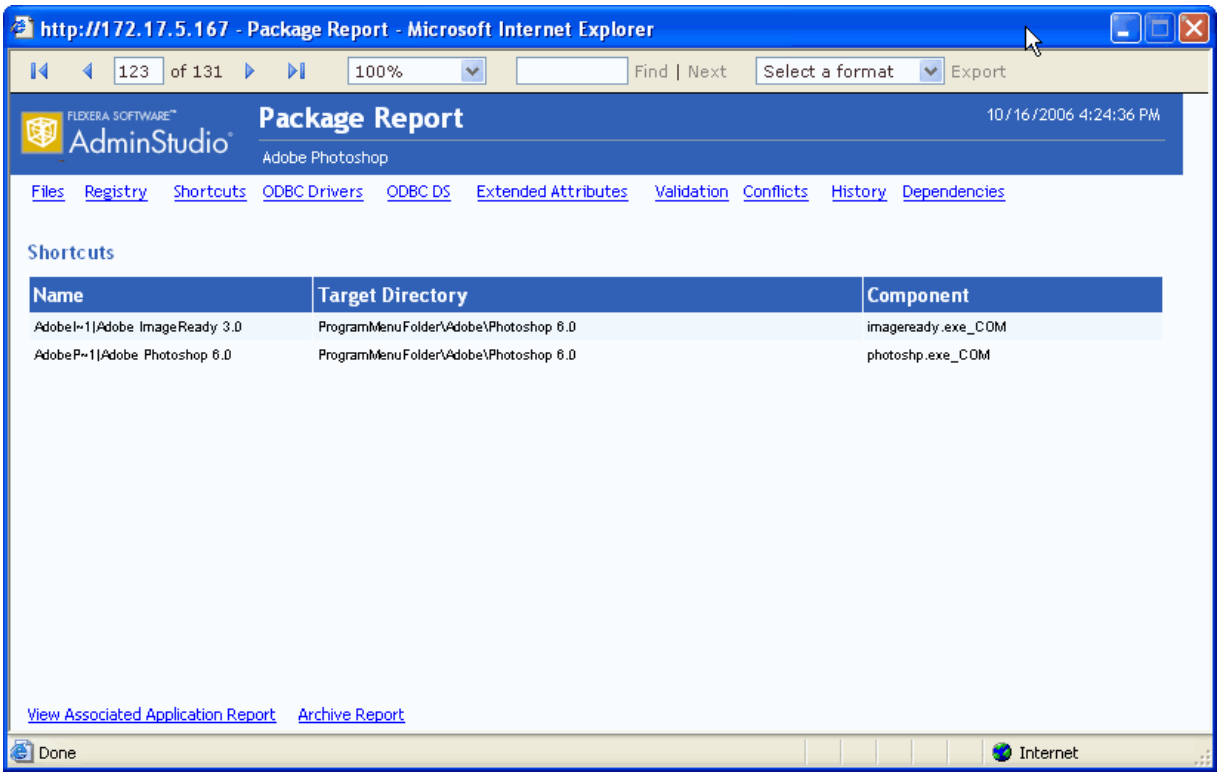


Figure 22-4: Package Report / Shortcuts View

For each shortcut, the following information is listed:

Table 22-9 • Package Report / Shortcuts Information

Item	Description
Name	Name identifying the shortcut.
Target Directory	Directory and executable that the shortcut invokes.
Component	Component associated with the shortcut.

ODBC Drivers View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **ODBC Drivers** view lists all of the Open Database Connectivity (ODBC) drivers in the package.

ODBC Resources are ones that involve interaction with databases. ODBC drivers are libraries that implement functions involving ODBC. Each database type has its own ODBC driver.

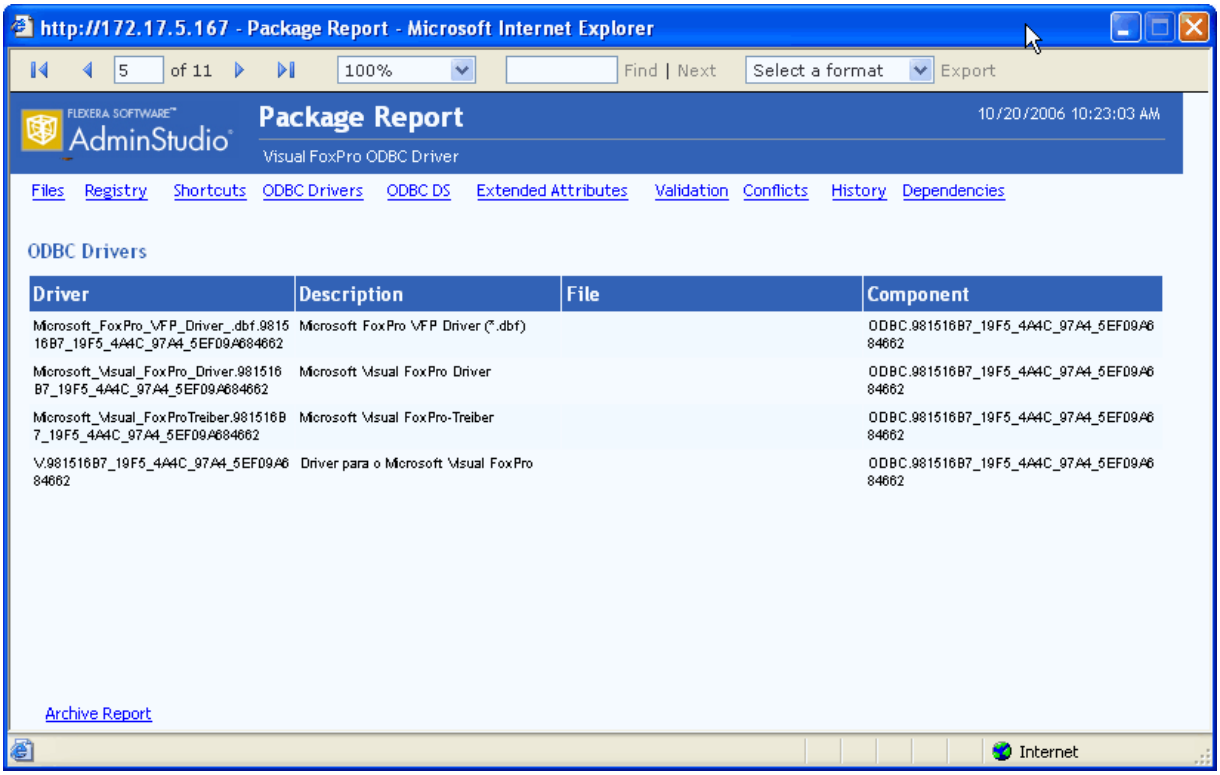


Figure 22-5: Package Report / ODBC Drivers View

For each ODBC driver, the following information is listed:

Table 22-10 • Package Report / ODBC Drivers Information

Item	Description
Driver	Name of an Open Database Connectivity (ODBC) driver in the package. Each database type has its own ODBC driver.
Description	Description of the ODBC driver identifying its associated database type.
File	File associated with the ODBC driver.
Component	Component associated with the ODBC driver.

ODBC DS View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **ODBC DS** view lists all of the Open Database Connectivity (ODBC) data sources in the package. An ODBC data source identifies the source database type and provides information on how to connect to that database.

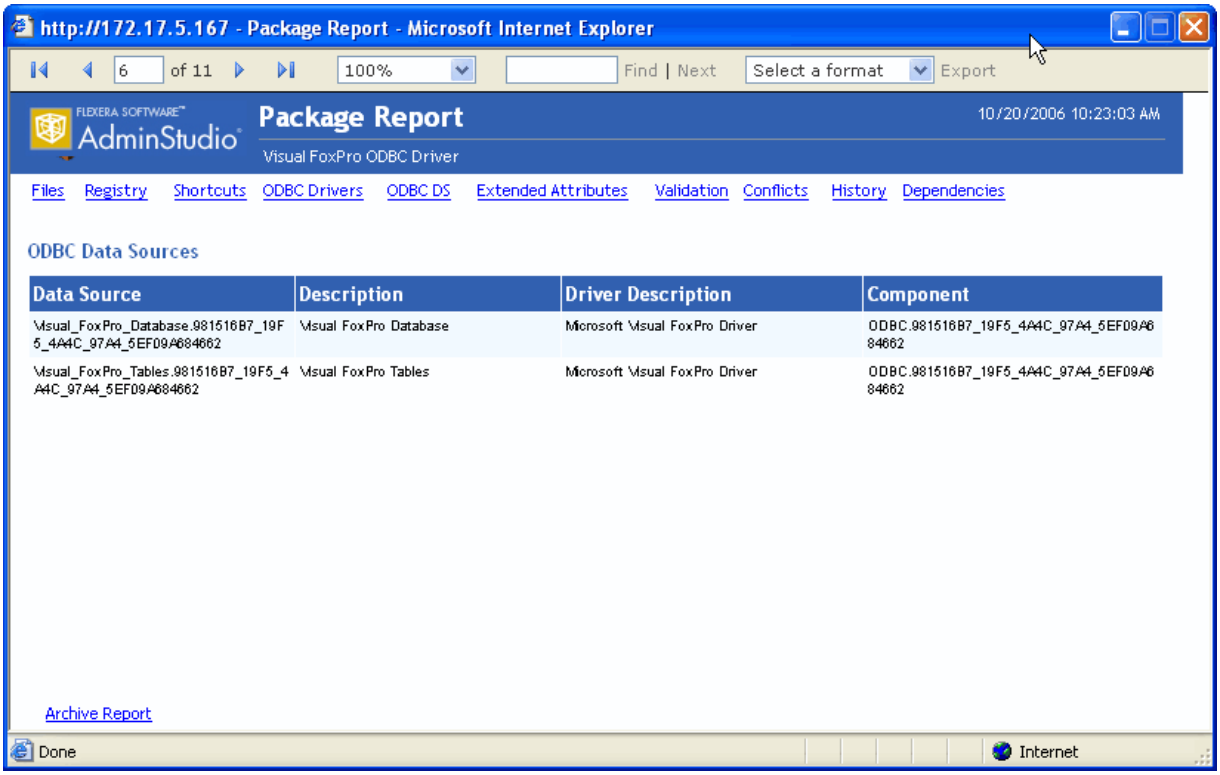


Figure 22-6: Package Report / ODBC Data Sources View

For each ODBC DS, the following information is listed:

Table 22-11 • Package Report / ODBC DS Information

Item	Description
Data Source	Name of the ODBC data source, which identifies the source database type and provides information on how to connect to that database.
Description	Identifies the database type.
Driver Description	Name of this ODBC data source's associated ODBC driver.
Component	Component that this ODBC data source is affiliated with.

Extended Attributes View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Extended Attributes** view lists all of the extended attribute metadata that has been entered for this package.

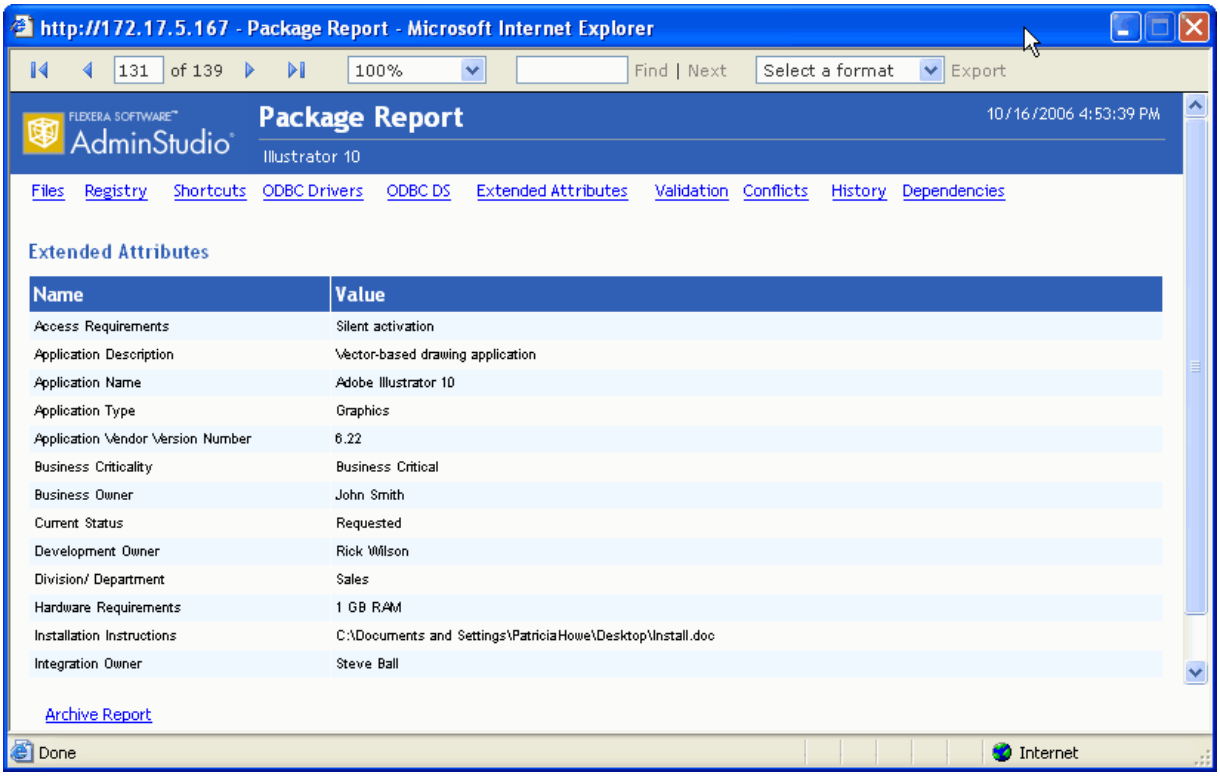


Figure 22-7: Package Report / Extended Attributes View

For each Extended Attribute, the following information is listed:

Table 22-12 • Package Report / Extended Attributes Information

Item	Description
Name	Name identifying the attribute.
Value	Content entered for the attribute.

Validation View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Validation** view lists all of the ICE rule errors and warnings that were generated when the package was validated against Microsoft ICEs (Internal Consistency Evaluators)—custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards.

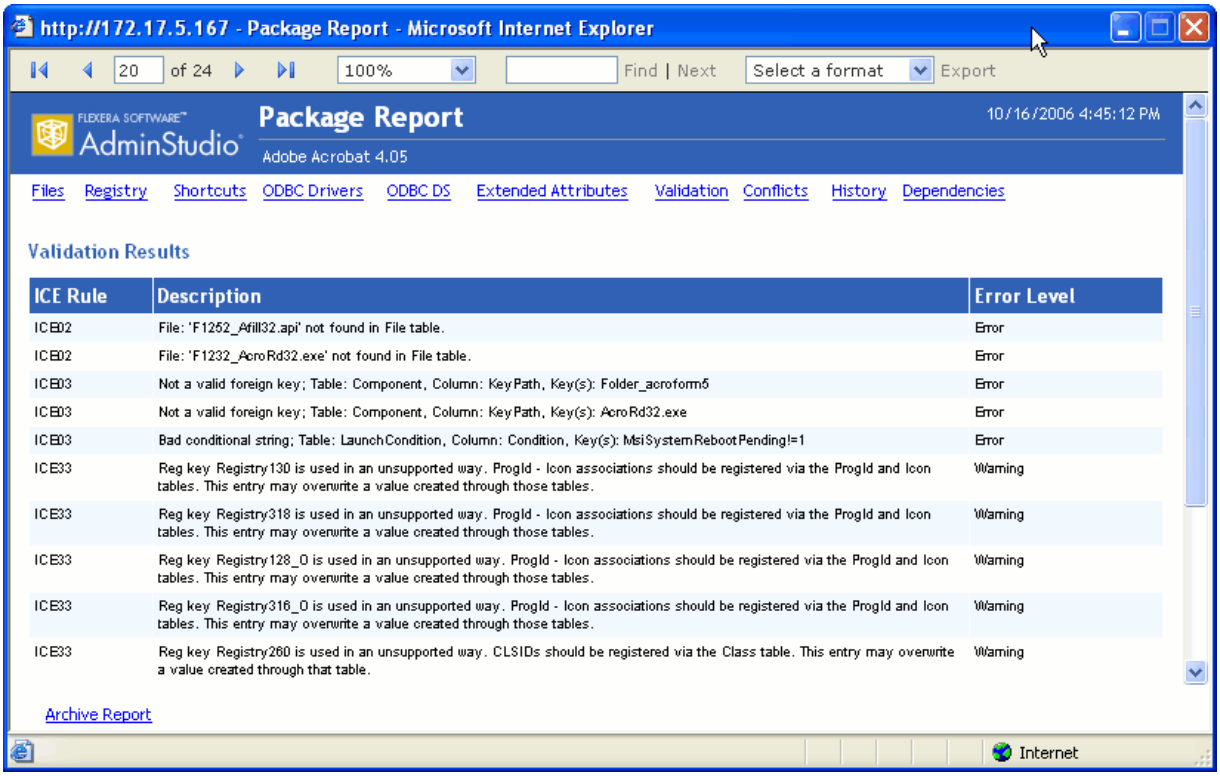


Figure 22-8: Package Report / Validation View

For each error or warning, the following information is listed:

Table 22-13 • Package Report / Validation Information

Item	Description
ICE Rule	Name of ICE Rule that generated an error or warning message.
Description	Error or warning message.
Error Level	Indicates the severity of the message as either being a Warning or an Error. • **Errors**—Package authoring that will cause incorrect behavior. • **Warnings**—Package authoring that could possibly cause incorrect behavior. Warnings can also report unexpected side-effects of package authoring.

Conflicts View

Edition *• Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Conflicts** view lists all of the unresolved errors that were found when conflict analysis was performed on this package.

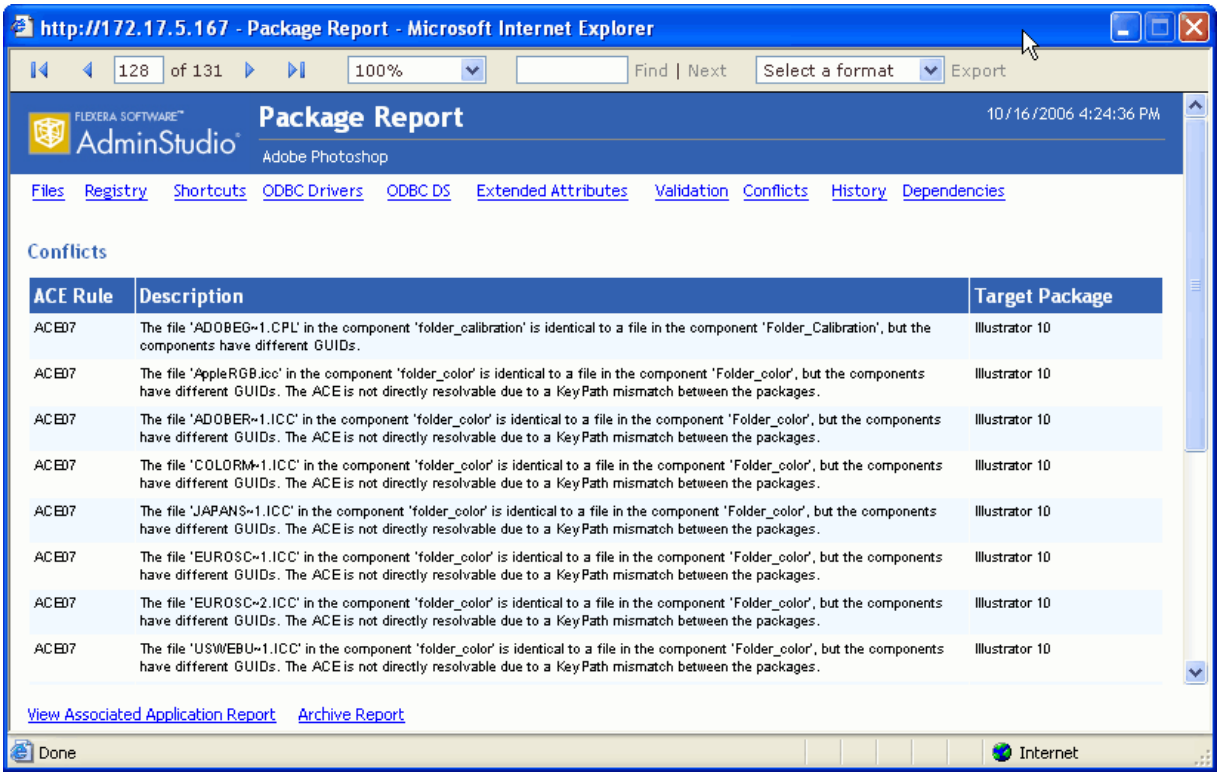


Figure 22-9: Package Report / Conflicts View

For each error, the following information is listed:

Table 22-14 • Package Report / Conflicts Information

Item	Description
ACE Rule	Name of ACE Rule that generated the message.
Description	Message generated during conflict analysis.
Target Package	Package that conflicted with this package.

History View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **History** view lists all of the actions that have been performed on this package since it was imported into the Application Catalog.

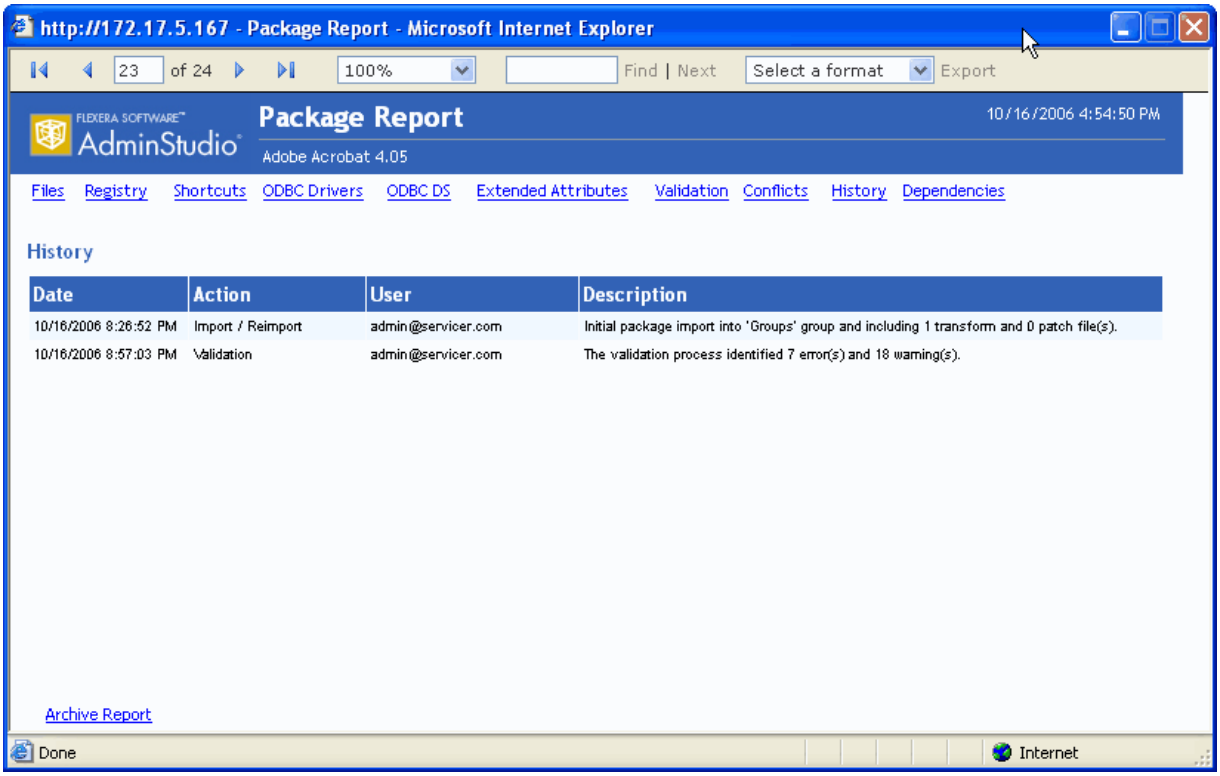


Figure 22-10: Package Report / History View

For each action, the following information is listed:

Table 22-15 • Package Report / History Information

Item	Description
Date	Day and time the event occurred.
Action	Identifies the event that occurred.
User	Identifies the user who executed the event.
Description	Description of the event that occurred.

Dependencies View

Edition *• Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Dependencies** view lists all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog.

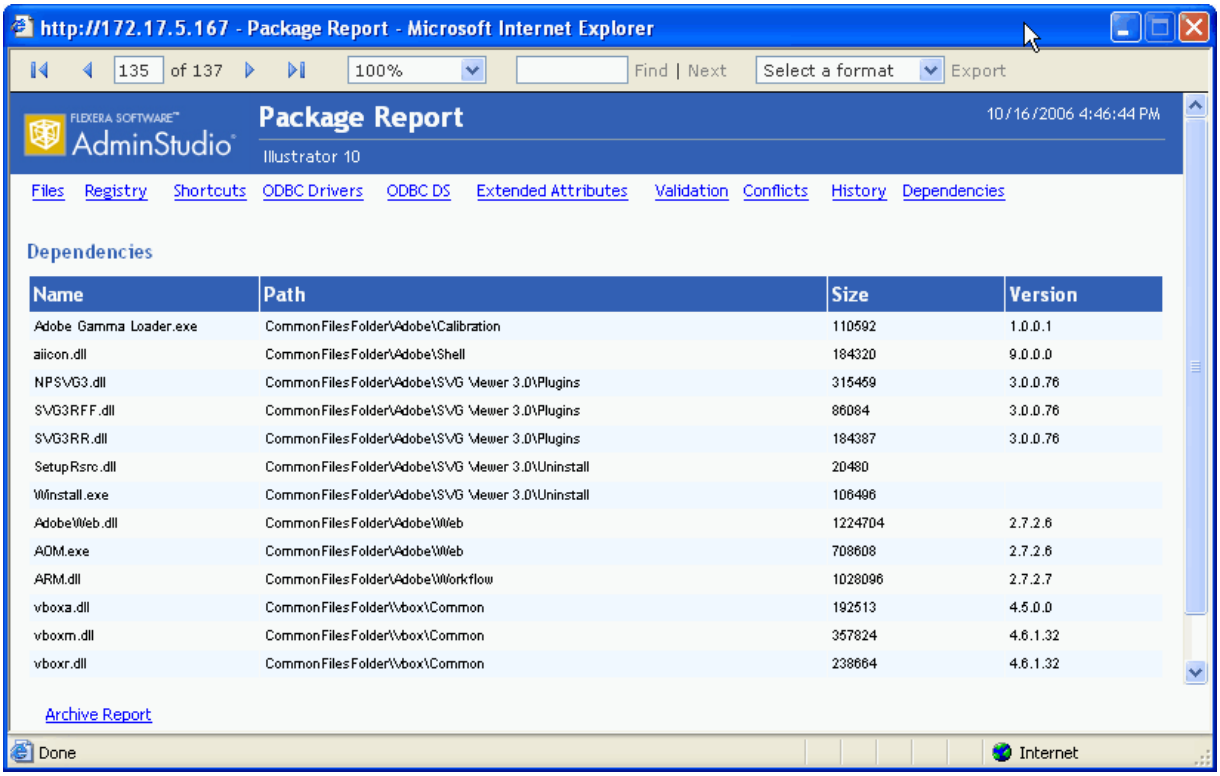


Figure 22-11: Package Report / Dependencies View

For each dependency, the following information is listed:

Table 22-16 • Package Report / Dependencies Information

Item	Description
Name	Name of a file associated with this package that has dependencies with files used by other products or operating systems in the Application Catalog.
Path	Location where this dependent file is installed.
Size	Size of the dependent file.
Version	Version of the dependent file.

Properties View

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The **Properties** view of the Package Report, which is only displayed for mobile apps, lists various attributes of the selected mobile application.

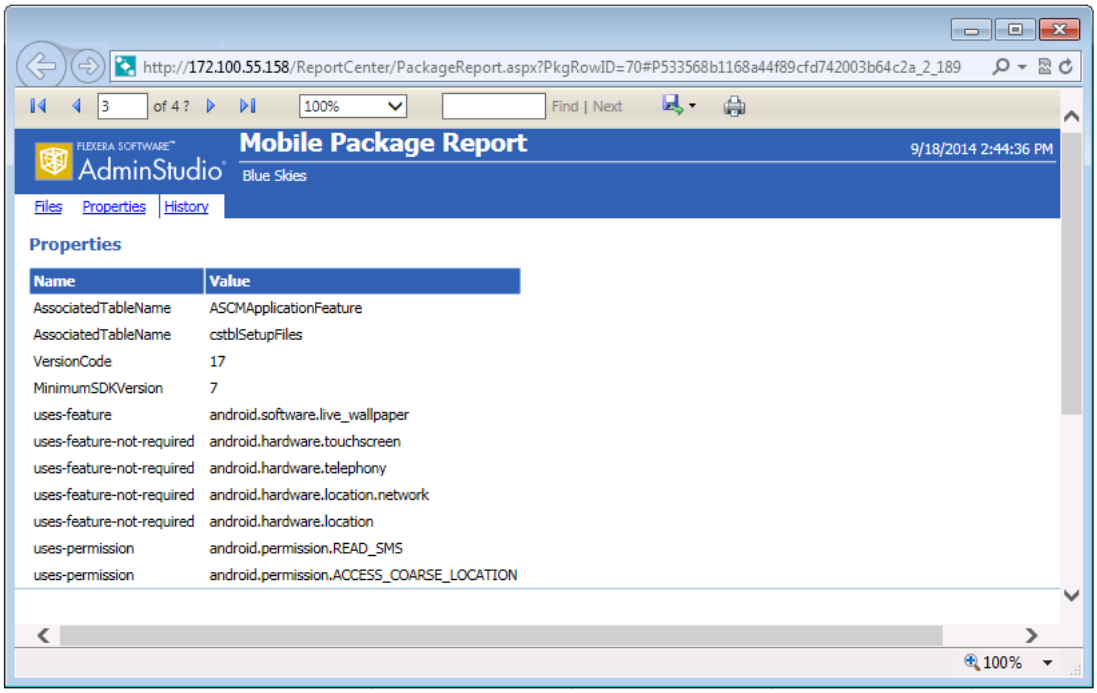


Figure 22-12: Package Report / Properties View

Navigating Through a Package Report

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

The Package Report consists of the initial Package Summary View and 10 other multi-page views which are accessed by clicking the links at the top of the report:

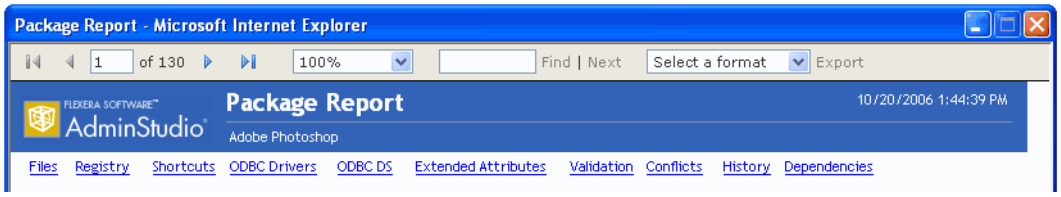


Figure 22-13: Navigation Links on the Package Report

Scrolling Through Pages of a View

Each of the Package Report views can be either a single page or multi-page, depending upon the content. The Package Report window is not resizable, so you cannot enlarge the window to display more items. Instead, you can use the Page Scrolling controls in the toolbar.

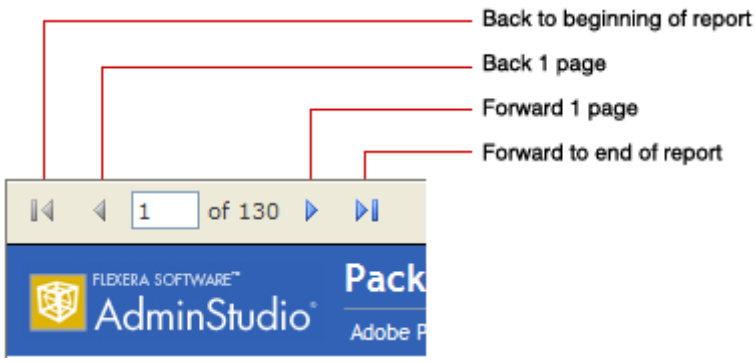


Figure 22-14: Page Scrolling Controls on Package Report

The total number of pages of the Package Report is listed in the toolbar, along with the number of the page that you are currently viewing. To jump to a specific page, enter a number in the box and click **Enter**.

Page 1 of the Package Report is the **Package Summary Information** view. Following this view, the rest of the views follow in the order in which they appear in the navigation links. The total number of pages in a Package Report is determined by adding the number of pages of all of the different views together.

Using Zoom Capability to Modify the Report Size

You can make selections from the Zoom list in the tool bar to enlarge or decrease the size of the report.

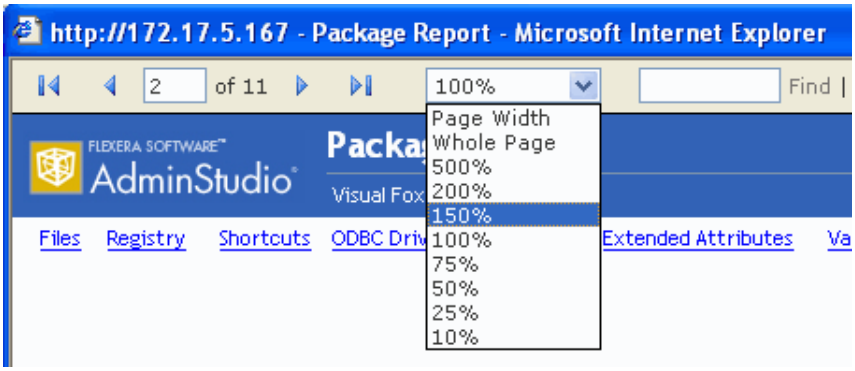


Figure 22-15: Zoom List on the Package Report

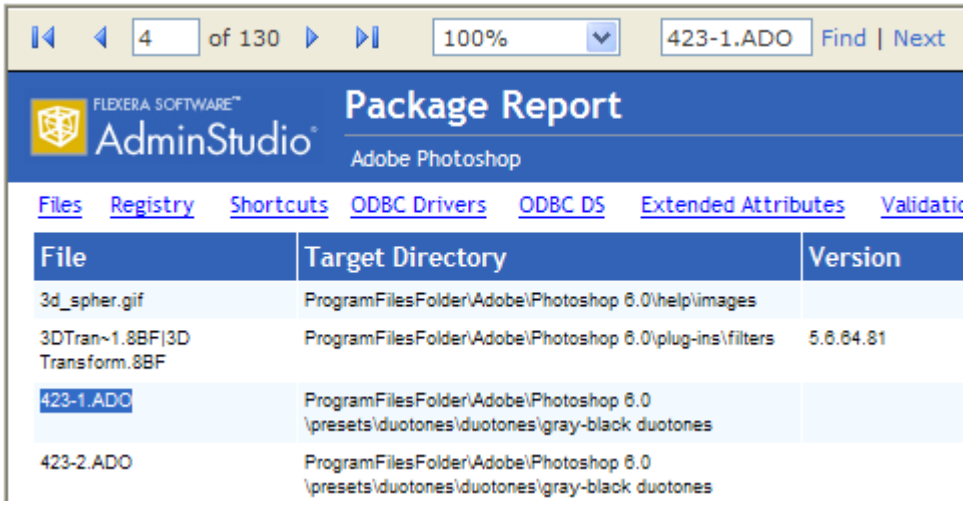
Note • *When you use the Zoom list to change the size of a Package Report, the size of the font used in the text is increased or decreased; however, the amount of information displayed on one page does not change.*

Searching for Information in a Package Report

You can use the **Find** box in the Package Report tool bar to search for specific information in the Package Report.

Task ***To search a Package Report:***

1. In the Package Report toolbar, enter the text you want to search for in the **Find** box and click **Find**. The page containing the first instance of that text is opened, and the text you searched for is highlighted.



2. Click **Next** in the tool bar to find the next instance of the text.

Archiving a Package Report

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

You can archive a Package Report to document a snapshot of a package's information as of a specific date and time.

Package Reports are saved in PDF format, and therefore can be easily distributed. An archived report looks very similar to the original report, except that it is a multiple-page PDF:

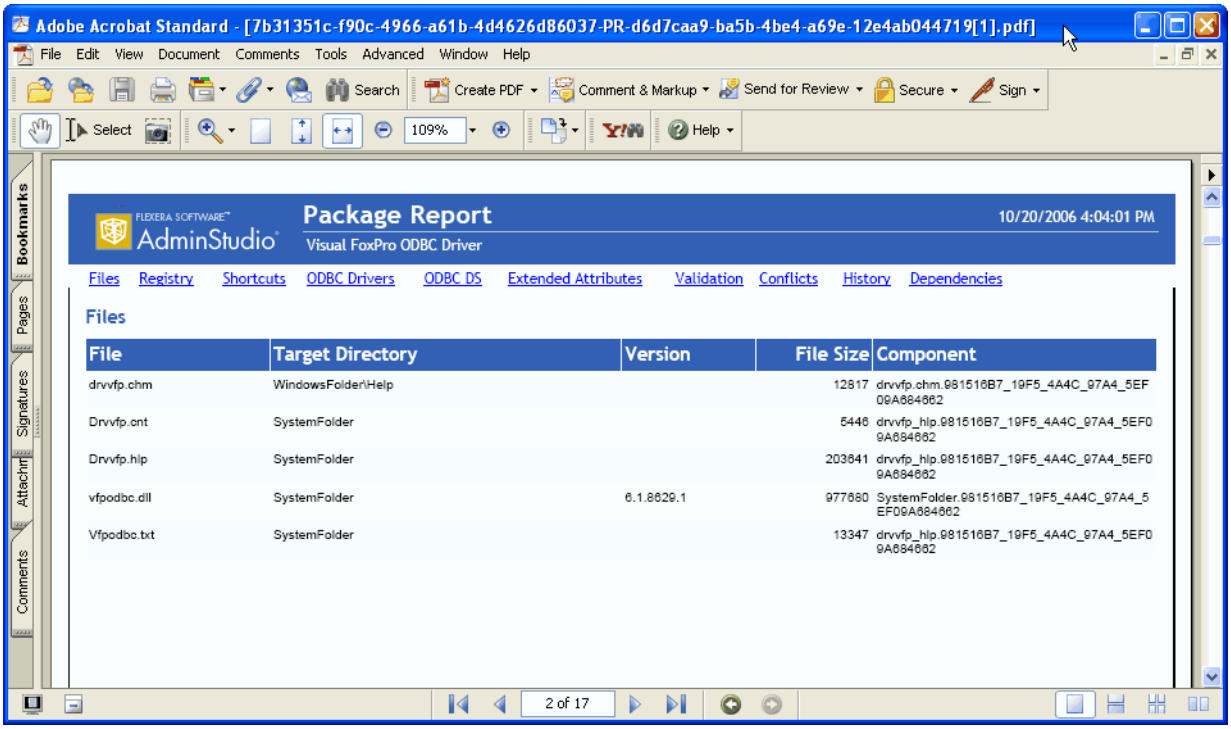


Figure 22-16: Archived Package Report

Note • *In an archived Package Report PDF, the navigation links at the top of the report (**Files**, **Registry**, **Shortcuts**, etc.) are not active. To scroll through the PDF, use the standard Adobe Reader controls.*

To archive a Package Report, perform the following steps.

Task ***To archive a Package Report:***

1. Open a Package Report.
2. Click the **Archive Report** link in the lower left corner any of the Package Report pages. The report is archived in PDF format and the following message is displayed:

   `The report has been archived.`
3. Click the Reports **All Reports** tab. The **All Reports** page opens, and the report that you just archived is listed.

   Note • *Each user's **Archived Reports** list only includes those reports that they archived. If you want others in your organization to view an archived report, you need to distribute the PDF via email or other delivery method.*
4. Click **View** next to the Package Report that you want to view. The report is opened in a PDF browser.

Deleting an Archived Package Report from the Archived Reports List

To delete an archived Package Report, perform the following steps.

Task ***To delete an archived Package Report from the Archived Reports list:***

1. In the Archived Reports list on the **All Reports** page, right-click on the archived report you want to delete, and then click **Delete**. You are prompted to confirm the deletion.
2. Click **OK**. The archived report is deleted.

Exporting a Package Report

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

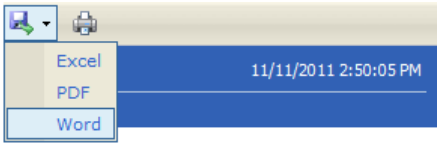
You can export the contents of a Package Report to an Excel (`.xls`) or Acrobat (`.pdf`) file, or Microsoft Word (`.doc`) file.

- **Excel .xls file**—When a Package Report is exported to Microsoft Excel format, each of the Package Report views are displayed on a different worksheet.
- **Acrobat .pdf file**—An exported Package Report in PDF format is the same as the PDF created when a Package Report is archived. See Archiving a Package Report.
- **Word (.doc) file**—The Package Report is exported in Microsoft Office Word 97 - 2003 format.

To export a Package Report, perform the following steps.

Task ***To export a Package Report:***

1. In the Package Report tool bar, click on the Export icon.
2. Select **Excel** or **PDF**, or **Word** from the list.



One the following occurs:

- If you selected **Excel** or **Word**, the **File Download** dialog box opens. Click **Save** and select a location for the exported file on the **Save As** dialog box.
- If you selected **PDF**, the PDF will open in a new browser window.

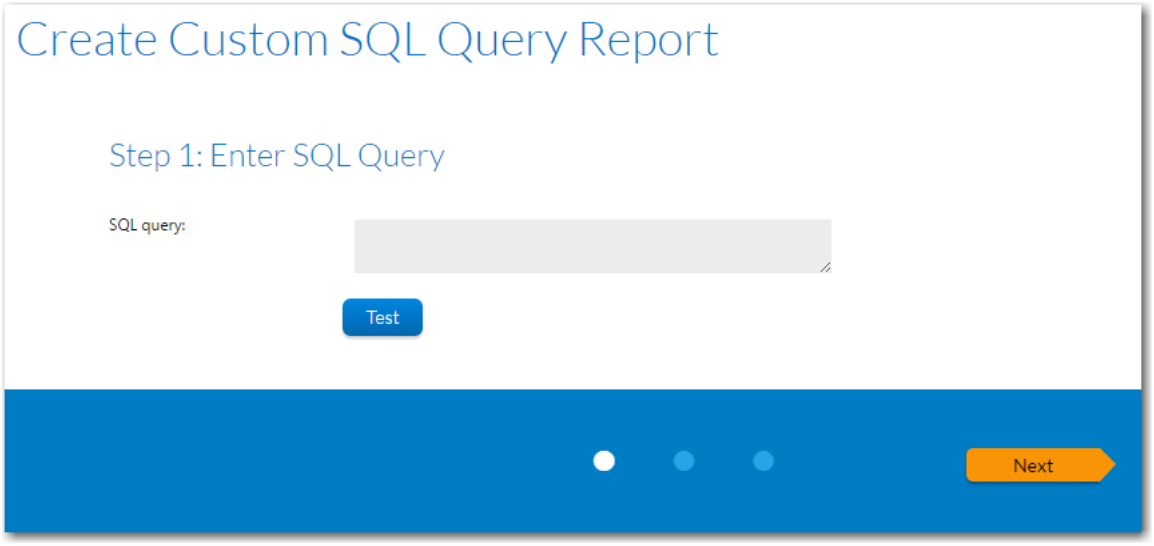
Generating a Custom SQL Query Report for AdminStudio

Edition • *Reports is included with AdminStudio Enterprise Edition and with Workflow Manager.*

You can generate a Custom SQL Query Report to include data generated by both AdminStudio and Workflow Manager. To generate a Custom SQL Query Report, perform the following steps.

Task ***To generate a new Custom SQL Query report:***

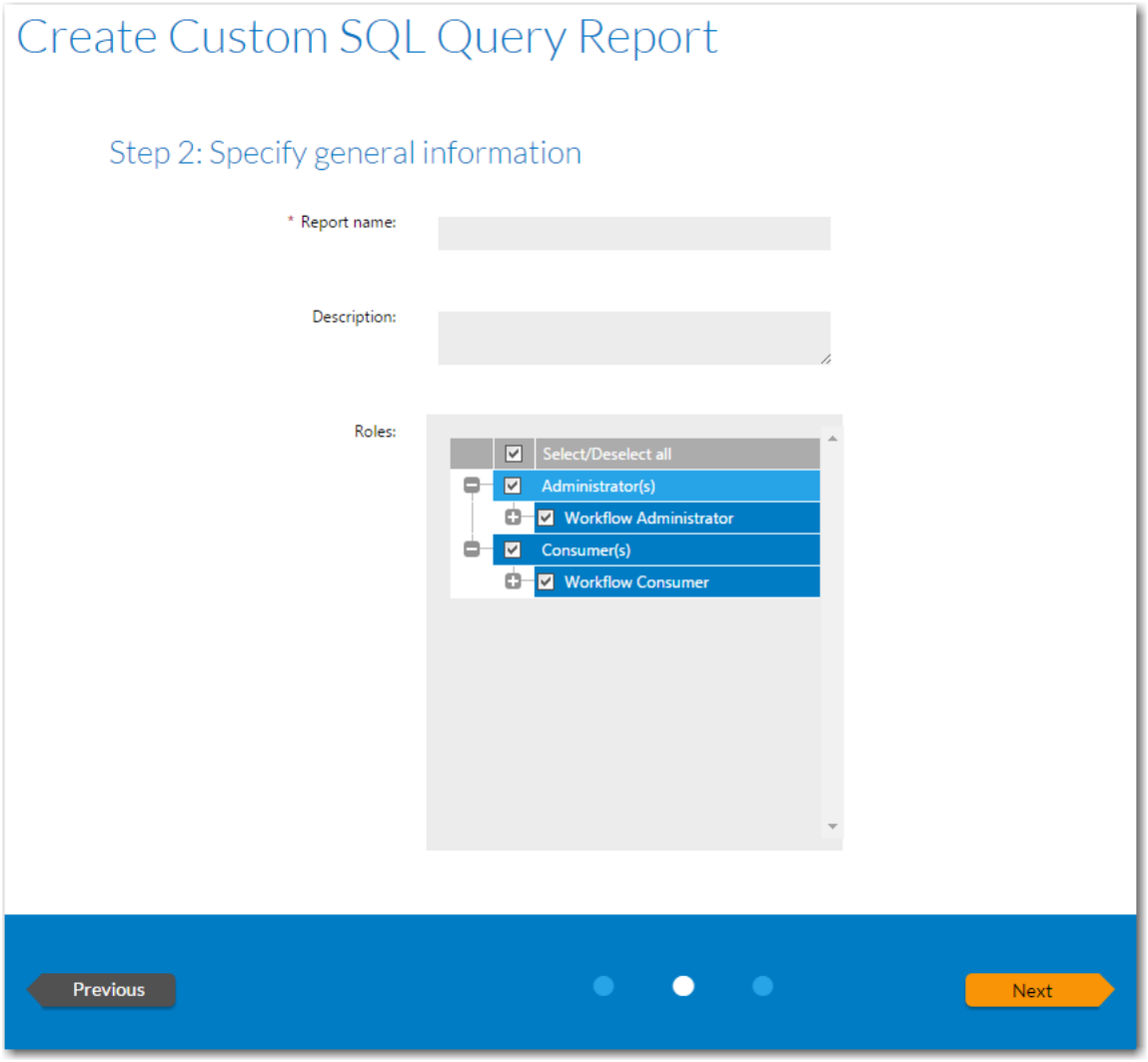
1. In the navigation bar, click **Create Custom SQL Query Report** on the **Reports** menu. The **Step 1: Enter SQL Query** panel of the **Create Custom SQL Query Report** page opens.



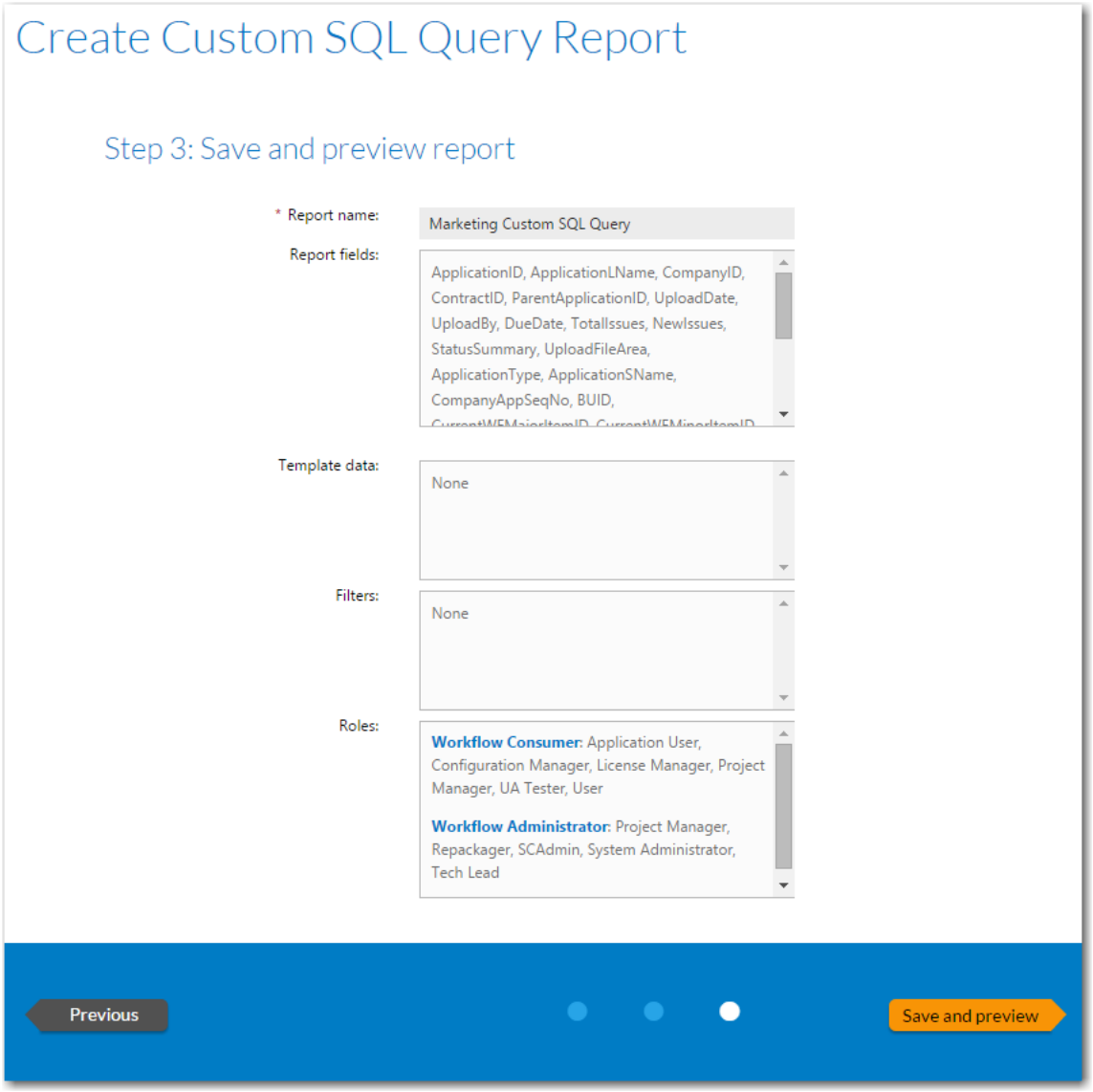
2. Enter an SQL query to retrieve the data for this report. Click the **Test Query** button to verify the query syntax.

Tip • *To assist you in writing queries to retrieve data, see Wildcard Support in Reports SQL Queries.*

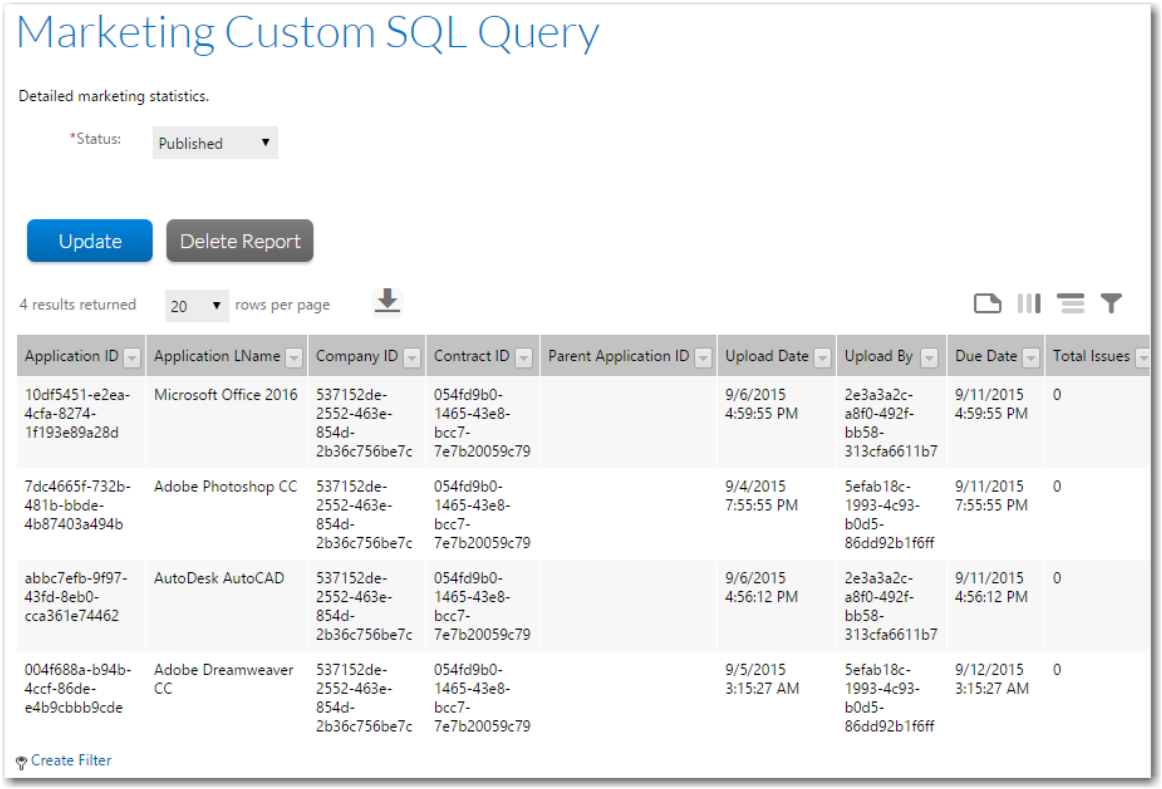
3. Click **Next**. The **Step 2: Specify general information** panel opens.



4. Enter a **Report name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.
5. Select the roles that you want to have permission to view this report.
6. Click **Next**. The **Step 3: Save and preview report** page opens, which displays all the information needed to create the report.



7. Click **Save and preview**. The report is generated. This report is also saved and now appears in the list on the **All Reports** page.



Wildcard Support in Reports SQL Queries

In Reports searches, the `LIKE` operator is always used. You can combine the `LIKE` operator with a wildcard character, and the following rules apply:

Table 22-17 • Wildcard Support in Reports Queries

Situation	Rule

When no wildcards are used	If you do not enter a wildcard character in the **Search** box, then Reports performs a "LIKE" search, which searches for any occurrence of that text anywhere in the item that is being searched for.
	
	For example, if you are searching for a file name that has the word `test` anywhere in the file name, and you entered **`test`** in the **Search** box, it would be interpreted by Reports as:
	
	`*test*`
	
	And the following files would be found:
	
	`MyTestFile and TestFile`

When wildcards are used	You can specify a * wildcard in the **Search** box to narrow the search results. For example, if you are searching for a file name that includes the word `test`, but does not begin with it, and you entered `*test` in the **Search** box, `MyTest` would be returned, but not `TestFile`.

Generating a Custom Stored Procedure Report for AdminStudio

You have the option of generating an AdminStudio Enterprise Server report using a stored procedure. A **Custom Stored Procedure Report** is a report on data generated by AdminStudio that is defined by specifying a stored procedure.

- Generating a New Custom Stored Procedure Report
- Sample Custom Stored Procedure Report

Generating a New Custom Stored Procedure Report

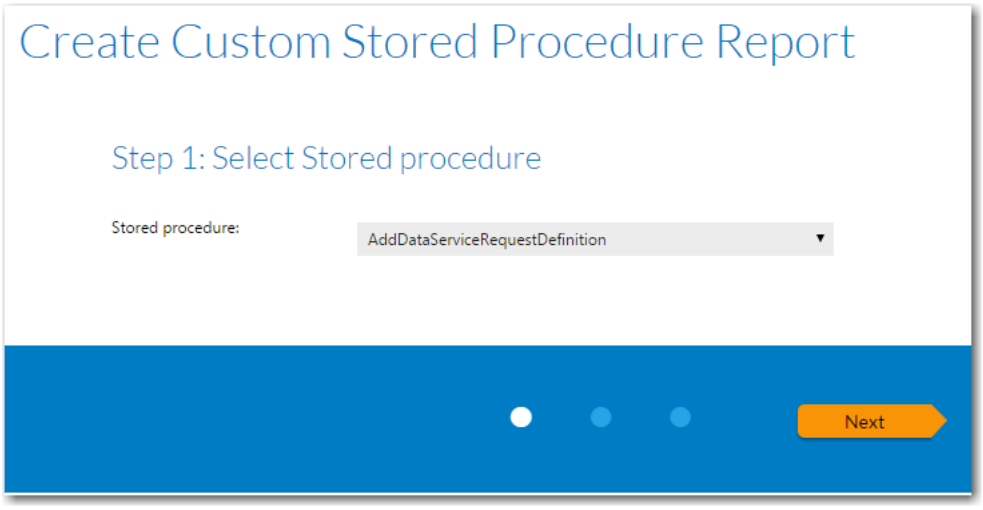
To generate a Custom Stored Procedure report, perform the following steps.

Task ***To generate a Custom Stored Procedure report:***

1. Open the `AMS_CustomReports` table and enter the names of the stored procedures you want to use to generate reports.

   Note • *For more information on stored procedures, see SQL Stored Procedures in Microsoft TechNet.*

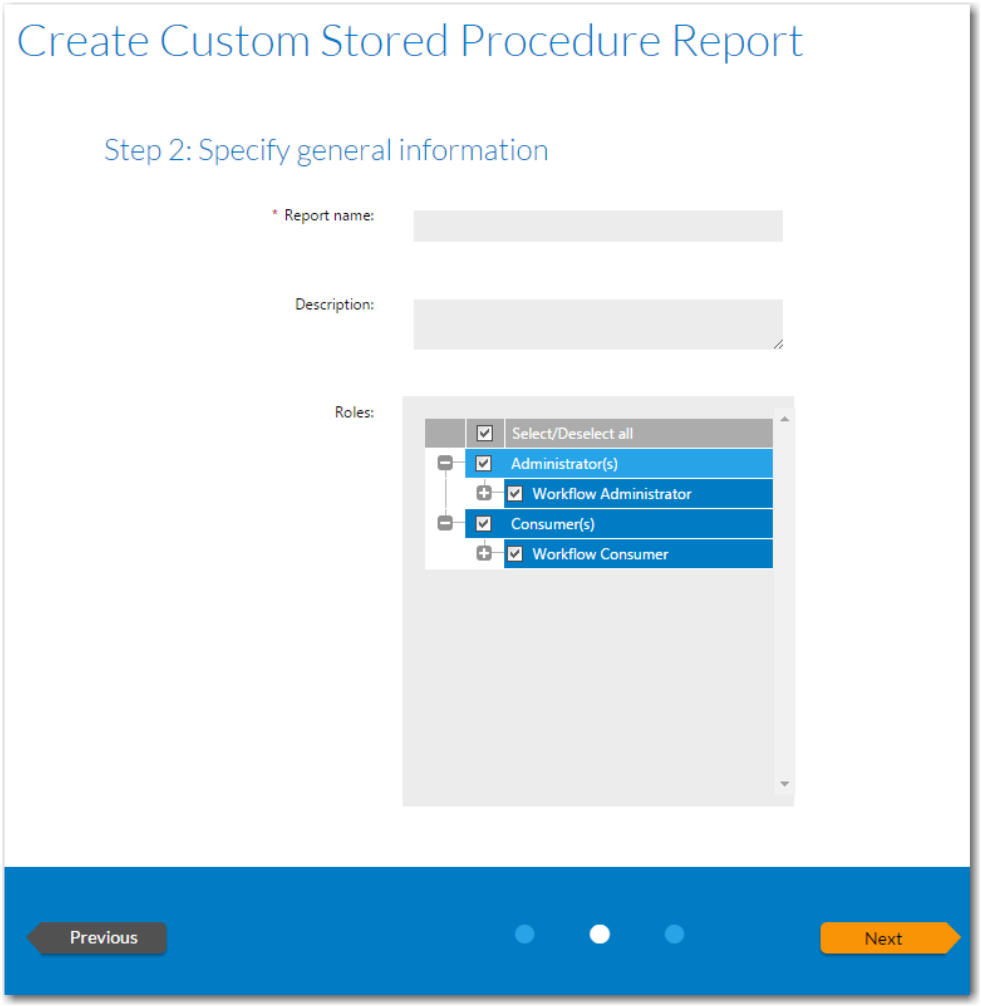
2. In the navigation bar, click **Create Custom Stored Procedure Report** on the **Reports** menu. The Step 1: Select Stored procedure of the Create Custom Stored Procedure Report page opens.

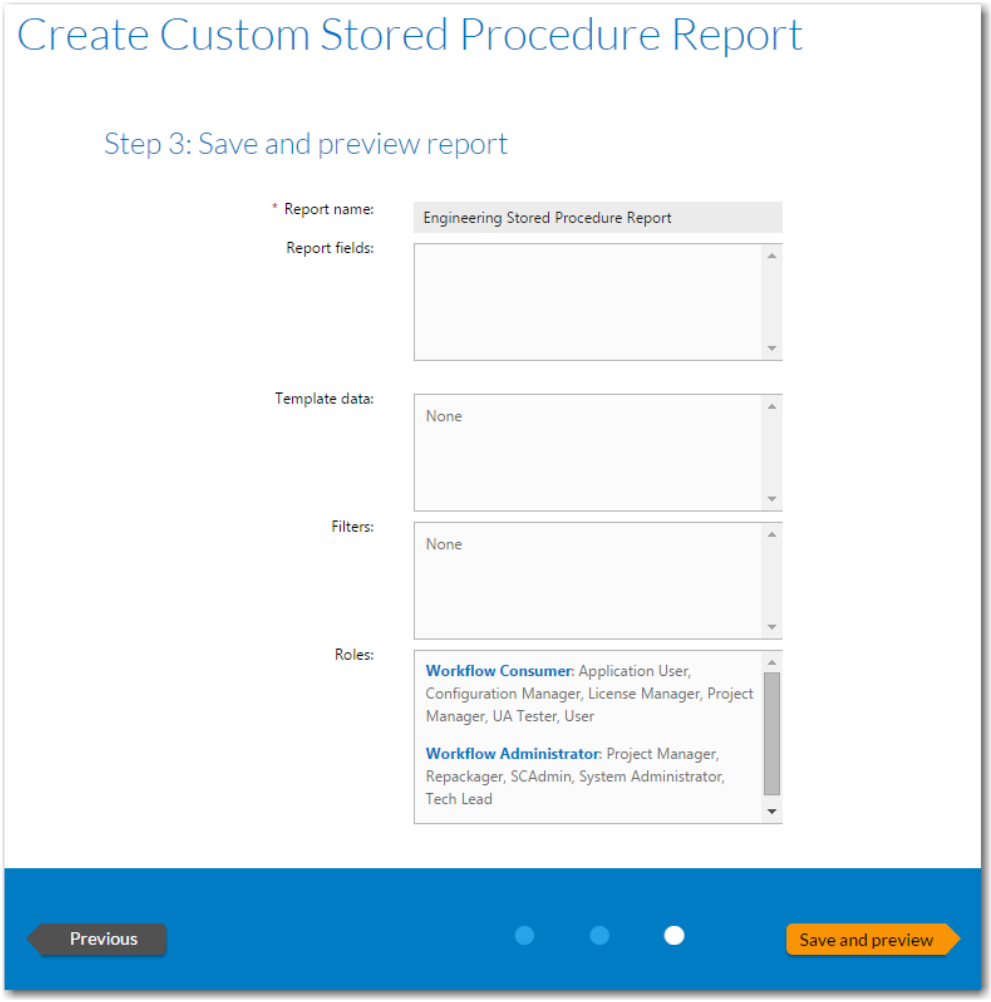
   The contents of this panel is determined by the selected stored procedure.

   Note • *This panel is customizable per customer need. The filters shown on the screen are based up on the parameters required by the stored procedure.*

3. Select a stored procedure from the list and specify any other requested information.
4. Click **Next**. The **Step 2: Specify general information** panel opens.



5. In the **Report name** field, enter a name to identify this report. This name will be listed on the **All Reports** page.
6. Enter a **Description** to identify the purpose of this report.
7. In the **Roles** section, select those roles that you want to assign permission to view this report.
8. Click **Next**. The **Step 3: Save and preview report** panel opens.



9. Click **Save and preview**. The report is displayed.

   The report is now saved and available to view by users with appropriate permission.

Sample Custom Stored Procedure Report

A sample custom stored procedure report is shipped with Workflow Manager named `usp_GetReadOnlyWorkflows`. This report lists all of the workflow requests that are in “read-only” status due to a stopped SLA clock or having an open critical issue.

You can view this report by selecting **Create Custom Stored Procedure Report** from the **Reports** menu, and then selecting the `usp_GetReadOnlyWorkflows` from the **Stored procedure** list.

Specifying Predefined Parameters in a Custom Stored Procedure Report

You can specify known predefined parameters in a stored procedure. When the report is displayed, it will include drop down lists that users can use to filter the data displayed in the custom report.

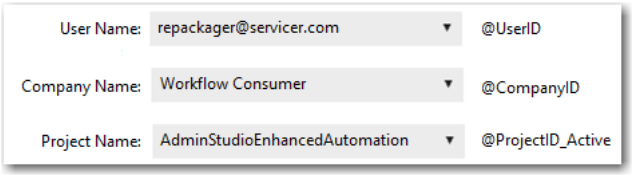


Figure 22-17: Predefined Parameters Displayed in Custom Stored Procedure Report

Workflow Manager ships with a default stored procedure named `usp_getreadonlyworkflows`. In this stored procedure, the following predefined parameters are supported:

Table 22-18 • Predefined Parameters

Parameters	Description
`@ApplicationID`	List of all the workflow request names.
`@ApplicationID_Completed`	List of all the workflow request names which are already completed.
`@ApplicationID_InProgress`	List of all the workflow request names which are in progress.
`@CompanyID`	For a consumer company, this will default to the company ID of the user's company. For an administrator company, this will default to the company ID of the user's company, but will also include the company IDs of corresponding client companies as well.
`@ProjectID`	List of all projects.
`@ProjectID_Active`	List of all active projects.
`@TemplateID`	List of all templates.
`@UserID`	List of all users based on CompanyID.
`@WorkflowStatus`	Values from `[AMS_ApplicationStatus]` table status.

You can also edit an XML file named `StoredProcParams.xml` in the Workflow Manager installation directory to add additional parameters.

Figure 22-18: StoredProcParams.xml File

This file installed in the following location:

`C:\AdminStudioWebComponents_2016\wwwroot\App_Data`

In the `StoredProcParams.xml` file, the `<Value>` element is a select statement that will be used to populate the drop down list for that parameter.

Viewing AdminStudio Application Catalog Reports

Edition • *Reports is included with AdminStudio Enterprise Edition and with Work flow Manager.*

Note • *In Workflow Manager, AdminStudio Application Catalog reports can be viewed by users in both Administrator and Consumer companies.*

On the **Application Catalog Dashboard** page, you can view a wide array of reports containing summary information on the applications in your Application Catalog. These reports give you insight into the readiness of those packages for distribution and for conversion to virtual packages.

You open the **Application Catalog Reports** page by selecting **Application Catalog Dashboard** on the **Reports** menu of the navigation bar. You switch between reports by selecting the report name from the **Select Report** list.

For example, **Application Readiness Dashboard** report is opened and shown in below figure.

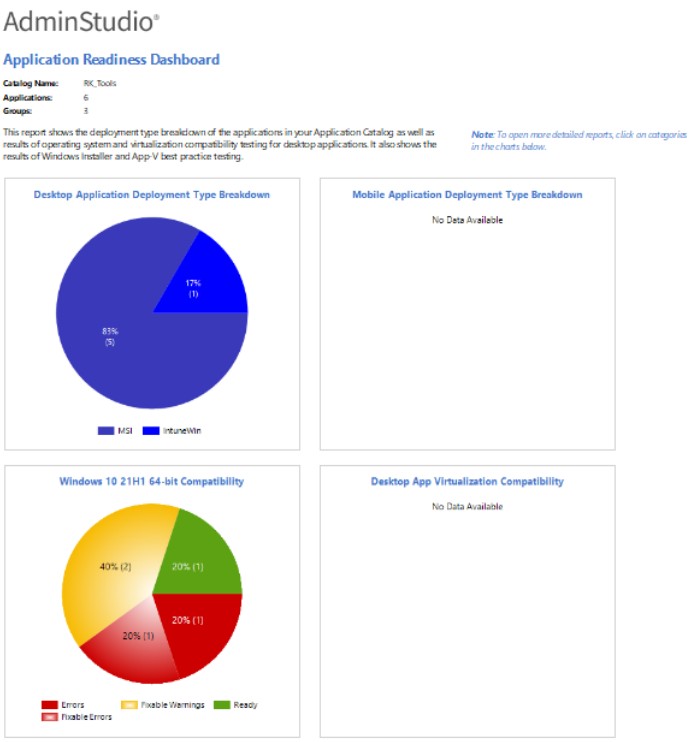


Figure 22-19: Application Readiness Dashboard Report

The available reports include test results from operating system compatibility, virtualization compatibility, remote application publishing compatibility, best practices testing, and application conflict testing. For Mac OS, Apple iOS, Google Android, and Microsoft Windows Phone mobile apps, reports on feature use, risk assessment, device compatibility, and/or policy compatibility are available. Reports are also included on App-V packages in your Application Catalog, as well as ConfigMgr (Formerly called as System Center Configuration Manager) deployment information.

For most reports, detailed sub-reports are available by clicking on one of the categories of the pie bar chart, on one of the numbers in an issue count column, or on a package name. Click on the available hyperlinks until you have explored all of the levels of the report.

For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab in the AdminStudio Help Library.

Generating and Viewing Workflow Manager Reports

Edition • *This feature is available in Workflow Manager only.*

Both workflow consumers and administrators can generate reports. All reports can be filtered by many common fields (such as project, company, and so on), and you can also export reports to many different formats, including PDF, RTF, XLSX, and CSV.

This section includes topics on the following:

- Generating Standard Reports
- Creating Custom Reports
- Exporting Report Data from Reports

Note • *You must be a Workflow Administrator with Administrative permissions to create a Report.*

Generating Standard Reports

Edition • *This feature is available in Workflow Manager only.*

Workflow administrators can open five system reports from the **All Reports** page. These reports provide you with detailed summary information about a company's projects and workflow requests.

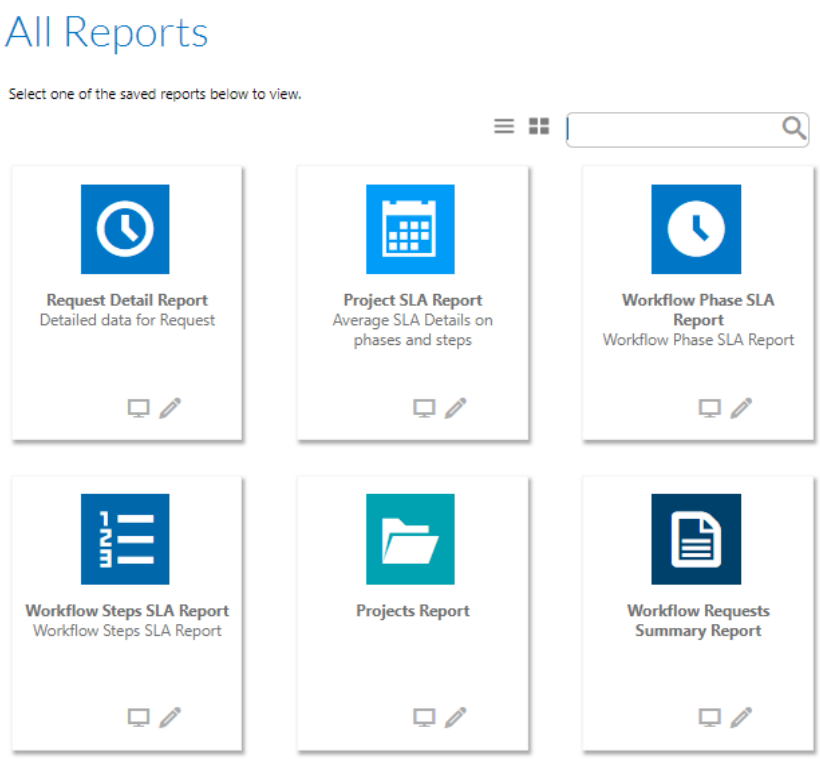


Figure 22-20: System Reports on the All Reports Page

The following reports are available:

- **Projects Report**—A report that groups projects by customer and returns summarized information including the progress and Service Level Agreement (SLA) status of workflow requests. You can choose to return information about one project or all of a company's projects. See Generating a Projects Report.

- **Workflow Requests Summary Report**—A report that lists all of a company's workflow requests, displaying the SLA status and workflow progress of each request. You can filter this report by SLA status. See Generating a Workflow Requests Summary Report.
- **Request Detail Report**—A report that lists information on workflow requests for a specific project. See Generating a Request Detail Report.
- **Project SLA Report**—A report that measures and reports on the SLA status for a specific project, or for all projects, during a specific date range. See Generating a Project SLA Report.
- **Workflow Phases SLA Report**—A report that lists all workflow phases for which phase-level SLA tracking is being performed along with their **SLA Status**. See Generating a Workflow Phases SLA Report.
- **Workflow Steps SLA Report**—A report that lists all workflow steps for which step-level SLA tracking is being performed along with their **SLA Status**. See Generating a Workflow Steps SLA Report.

Opening a System Report

You can open all of these System Reports from either the **All Reports** page or by selecting them on the **Reports** menu of the navigation bar.

Task ***To open a System Report:***

1. Click on a report name on the **Reports** menu of the navigation bar.
2. Follow the instructions in one of the following topics:
   - Generating a Projects Report
   - Generating a Workflow Requests Summary Report
   - Generating a Request Detail Report
   - Generating a Project SLA Report
   - Generating a Workflow Phases SLA Report
   - Generating a Workflow Steps SLA Report

Setting View Permissions for the Projects or Workflow Requests Summary Reports

You can specify which roles at your company are able to view the Projects and the Workflow Requests Summary reports.

Task ***To set view permissions:***

1. On the **All Reports** page, click the Edit icon **Projects Report** or **Workflow Requests Summary Report**. The report's information panel opens, listing all of the roles for both the Workflow Consumer and Workflow Administrator company.
2. Select the roles that you want to be able to view the selected report.
3. Click **Save**.

Generating a Projects Report

Edition • *This feature is available in Workflow Manager only.*

A **Projects Report** groups projects by customer and returns summarized information including the progress and Service Level Agreement (SLA) status of workflow requests. You can choose to return information about one project or all of a company's projects.

Task ***To view a Projects Report:***

1. Click **Projects Report** on the **Reports** menu. The **Projects Report** page opens.
2. Select the company that you want to report on from the **Company** list.
3. Choose one of the company's projects from the **Projects** list, or select **** View for all **** to return information about all of a company's projects.
4. To return only workflow requests with specific SLA Status values, choose one or more of: **Completed On Time**, **Completed Late**, **On Time**, **At Risk**, or **Late**. Select all of these if you want to report on all workflow requests.
5. Click **View Report**. The **Projects Report** opens. See Projects Report for more information.

Generating a Workflow Requests Summary Report

Edition • *This feature is available in Workflow Manager only.*

A **Workflow Requests Summary Report** groups workflow requests by company, and presents information on their progress and Service Level Agreement (SLA) status.

You can choose to return information about one workflow request or about all of a company's workflow requests. You can also filter the report by SLA status.

Task ***To view a Workflow Requests Summary Report:***

1. Click **Workflow Requests Summary Report** on the **Reports** menu. The **Workflow Requests Summary Report** page opens.
2. From the **Company** list, select the company that you want to view a report on.
3. Select one of the following options to specify which requests to include in this report:
   - **Single Workflow Request**—Select this option to return information about only one workflow request, which you select from the associated combo box.
   - **Multiple Workflow Requests**—Select this option to return all workflow requests with specific SLA status values. Choose one or more of **Completed On Time**, **Completed Late**, **On Time**, **At Risk**, or **Late** to indicate which workflow requests you wish to return. Select all of these options to return all workflow requests associated with a project

4. If you chose to generate a report about a single workflow request, the **Additional Metadata Filter Conditions** check box becomes visible. Select this check box to return filter your report by the values provided by people as they complete the data elements in a workflow request. If you select the checkbox, the metadata filter fields appear, and you should do the following:

   a. Select the data element that you want to filter by from the **Metadata Field** list.

   b. Enter a value for the selected **Metadata Field** in the **Condition Value** box, or select a value from the **Values List** (when available).

   c. Select the appropriate **Operator** from the list (`AND` or `OR`).

   d. Continue adding **Metadata Fields**, if desired. You can filter by up to four fields.

5. Click **View Report**. The **Workflow Requests Summary Report** opens. See Workflow Requests Summary Report for more information.

Generating a Request Detail Report

Edition • *This feature is available in Workflow Manager only.*

A **Request Detail Report** groups workflow requests by project, and presents information on their progress and Service Level Agreement (SLA) status.

You can choose to return information about workflow requests for one project or for all of the projects of a company.

Task ***To view a Request Detail report:***

1. Click **Request Detail Report** on the **Reports** menu on the navigation bar. The **Request Detail Report** page opens.
2. From the **Company/Business Unit Name** list, select the company that you want to view a report on. You can also select **** View for all ****.
3. From the **Template** list, select a template. You can also select **** View for all ****.
4. From the **Project** list, select a template. You can also select **** View for all ****.
5. From the **Date Range From** and **Date Range To** lists, specify the date range that you want the report to cover.
6. Click **View Report**. The **Request Detail Report** opens. See Workflow Requests Summary Report for more information.

Generating a Project SLA Report

Edition • *This feature is available in Workflow Manager only.*

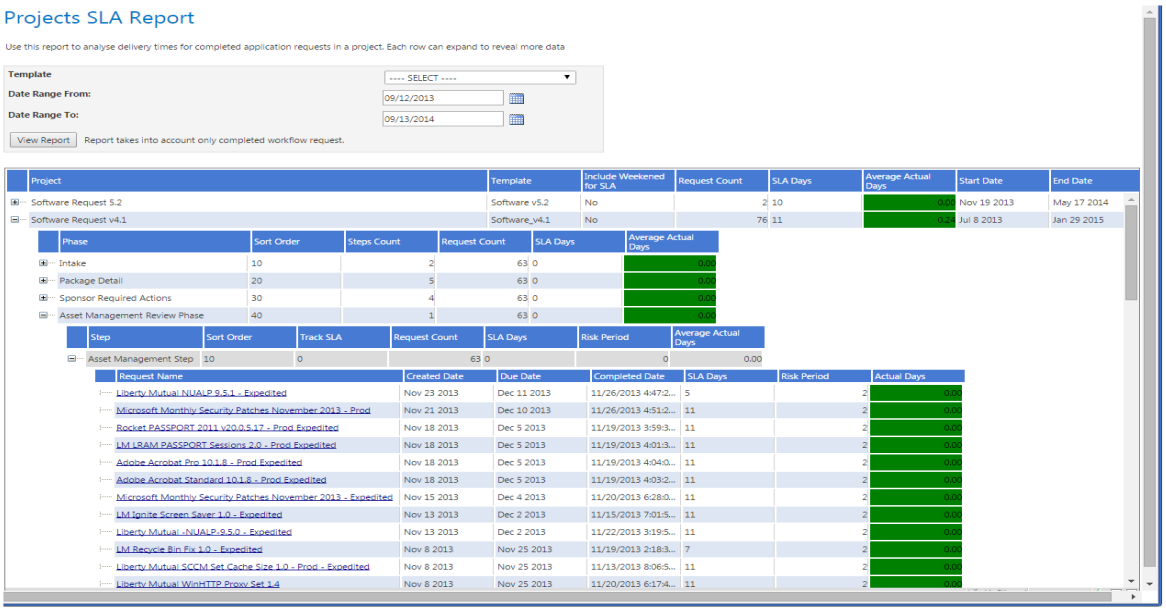
You can generate a **Project SLA Report** to measure and report on the SLA status for a specific project, or for all projects, during a specific date range. This helps you helps you analyze the delivery time for any completed project, and identify bottlenecks and weak points in your process.

Using this report, you can view projects within a specific date range, and then drill down from project level to workflow requests across both phases and steps to see the SLA status at each level.

Task ***To generate a Project SLA Report:***

1. Click **Project SLA Report** on the **Reports** menu on the navigation bar. The **Project SLA Report** page opens.
2. From the **Template** list, select the name of the Workflow Template used by the project or projects that you want to view SLA information for.

   Important • *To generate a report that lists SLA data for all projects during a specific date range, do not make a selection from the* ***Template*** *list.*
3. For the **Data Range From** and **Date Range To** fields, identify the date range for which you want to view project data.
4. Click **View Report** to generate the Project SLA Report.



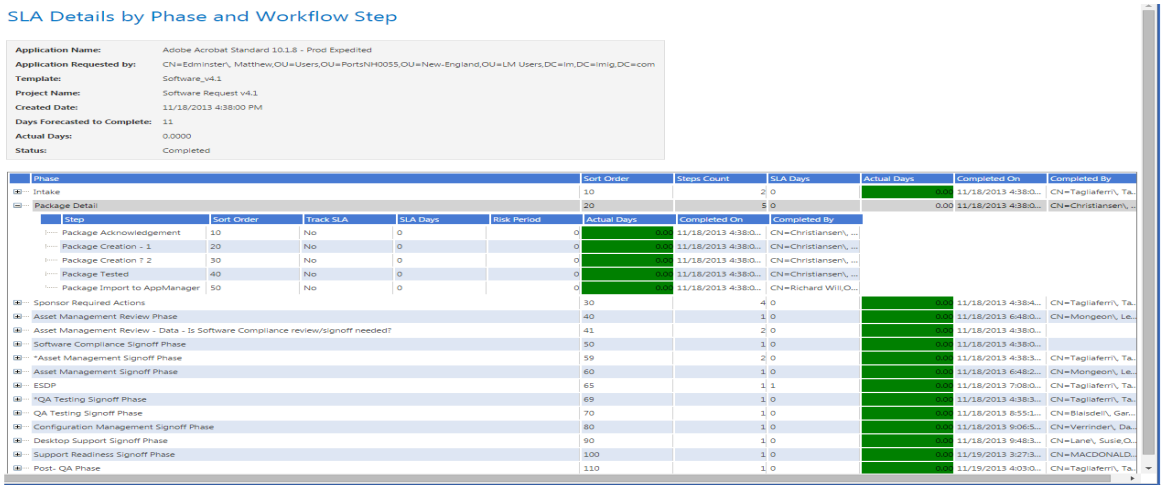
5. Click the plus signs to expand the listing to view SLA data across phases and steps for a specific project.
6. To view SLA information on a specific workflow request, click the hyperlinked **Request Name** to open the SLA Details by Phase and Workflow Step Subreport for that workflow request.

SLA Details by Phase and Workflow Step Subreport

The **SLA Details by Phase and Workflow Step** report lists SLA data for a specific workflow request.

Task ***To open an SLA Details by Phase and Workflow Step report:***

1. Open a Project SLA Report, as described in Generating a Project SLA Report.
2. Click the plus signs to expand the listing until you can view the SLA data for a specific workflow step. Workflow requests that contain that workflow step are listed.
3. Under the expanded workflow step, click the hyperlinked **Request Name** to open the **SLA Details by Phase and Workflow Step Subreport** for that workflow request.



4. To view the SLA data for other workflow steps in that workflow request, use the plus signs to expand the listing.

Generating a Workflow Phases SLA Report

Edition • *This feature is available in Workflow Manager only.*

A **Workflow Phases SLA Report** lists all workflow phases for which step-level SLA tracking is being performed along with their **SLA Status**. SLA (Service Level Agreement) time tracking is used to determine the status of a workflow phase (or workflow request) in relationship to its SLA due date as either: In Progress, On Time, At Risk, Late, Completed on Time, or Completed Late.

Note • *For information on enabling workflow-*phase *level SLA tracking, see "Tracking a Workflow Request or Workflow Phase's SLA Status" in the Workflow Manager Help Library.*

Task ***To generate a Workflow* Phases *SLA Report:***

1. Click **Workflow Phases SLA Report** on the **Reports** menu on the navigation bar. The **Workflow Phases SLA Report** page opens.
2. To display the SLA status of workflow steps from all workflow requests, even those that have been completed, clear the selection of the **Only include Workflow Phases in active Workflow Requests** option and click **Refresh Report**.

Generating a Workflow Steps SLA Report

Edition • *This feature is available in Workflow Manager only.*

A **Workflow Steps SLA Report** lists all workflow steps for which step-level SLA tracking is being performed along with their **SLA Status**. SLA (Service Level Agreement) time tracking is used to determine the status of a workflow step (or workflow request) in relationship to its SLA due date as either: In Progress, On Time, At Risk, Late, Completed on Time, or Completed Late.

Note • *For information on enabling workflow-step level SLA tracking, see "Tracking a Workflow Request or Workflow Step's SLA Status" in the Workflow Manager Help Library.*

Task ***To generate a Workflow Steps SLA Report:***

1. Click **Workflow Steps SLA Report** on the **Reports** menu on the navigation bar. The **Workflow Steps SLA Report** page opens.
2. To display the SLA status of workflow steps from all workflow requests, even those that have been completed, clear the selection of the **Only include Workflow Steps in active Workflow Requests** option and click **Refresh Report**.

Creating Custom Reports

Edition • *This feature is available in Workflow Manager only.*

You can create the following types of custom reports using the Reports Wizard:

- **Custom Report**—A custom report defined by using the Reports Wizard. See Creating a Custom Report.
- **Activity Report**—A custom report, which you define using the Report Wizard, that displays a listing of activities that occur during the completion of a request. See Creating an Activity Report.
- **Custom SQL Query Report**—A custom report defined by entering an SQL query in the Report Wizard. See Generating a Custom SQL Query Report.
- **Custom Stored Procedure Report**—A custom report defined by specifying a stored procedure in the Report Wizard. See Generating a Custom Stored Procedure Report.

Additional information is provided in this section that may help you generate custom reports:

- **Wildcard Support in Reports SQL Queries**—You can combine the SQL `LIKE` operator with wildcard characters to perform searches. See Wildcard Support in Reports SQL Queries.
- **Sample SQL Queries Used to Generate Project and Workflow Request Reports**—Sample SQL queries that are used to generate the built-in Project and Workflow Requests reports are provided. These sample queries might be helpful to refer to when you are creating your own custom reports. See Sample SQL Queries Used to Generate Project and Workflow Request Reports.

Creating a Custom Report

Edition • *This feature is available in Workflow Manager only.*

To create a new custom report, perform the following steps.

Task ***To create a new report:***

1. In the navigation bar, click **Create Custom Workflow Manager Report** on the **Reports** menu. The **Step 1: Select report objects** panel of the **Create Custom Workflow Manager Report** page opens.
2. Select the objects that you would like to include in the report and click **Next**. The **Select report fields** panel opens, listing all of the defined fields by object type. Only the objects that you selected in the previous step will be listed.
3. Select the report fields that you would like to include in the report and click **Next**. The **Select report filters** panel opens, where you can filter the data that you want to appear in the report.
4. Click on a field in the tree and set its filter on the right side using the drop-down boxes and the text box, selecting appropriate logical conditions which are populated according to the selected field. Each time you create a filter, click **Add** to add the filter to the current filter conditions.

   Note • *Even though you may not have included all of the available report fields in this report, you can still filter the data using all of these report fields.*
5. Click **Test** to test the created query for your report.
6. When you are satisfied with the filter conditions, click **Next**. The **Templates** panel opens, listing all available template data.
7. Expand the templates in the tree and select the data that you want to include in the report. All of the data groups and data elements associated with the selected template are listed.

   To display only those templates that are in use in the **Available Templates** list, select the **Templates in use only** option.
8. Click **Next**. The **Specify general information** panel opens.
9. Enter a **Report name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.
10. Select the Administrator and Consumer roles that you want to have permission to view this report.

11. Click **Next**. The **Save and preview report** panel opens, which displays all the information needed to create the report.
12. Click **Save and preview**. The report is generated. This report is also saved and now appears in the list on the **All Reports** page.

Creating an Activity Report

Edition • *This feature is available in Workflow Manager only.*

Every time an activity or event occurs during the completion of a request, Workflow Manager records that activity. You can view a listing of these activities in the Activity Report, a custom report which you define using the Report Wizard.

- Activities Displayed in the Activity Report
- Information that Can Be Included in an Activity Report
- How to Create an Activity Report

Activities Displayed in the Activity Report

The Activity Report lists a record for each time one of the following activities occurs during the completion of a request:

Table 22-19 • Activities Listed in the Activity Report

Activity	Description
Request Name Change	Occurs when a user edits the **Workflow Name** field on the **Properties** tab of the **Workflow Request** page and clicks **Update**.
Request Status Changed	Occurs when a user edits the **Status** of a request on the **Properties** tab of the **Workflow Request** page and clicks **Update**.
Data Acceptance Begins	Occurs when a user clicks the **Submit** button after they have entered all of the initial data that is required for a request (the Data Entry Step of the first Workflow Phase). ***Note*** • *This event occurs simultaneously with the Data Submission Complete event.*
Data Acceptance Cancel	Occurs when a Workflow Administrator clicks the **Reject Data** button to reject the data submitted during a request's Data Entry Step. Each time data is rejected, three activities are recorded: • Data Acceptance Cancel • Data Rejected • Data Submission Begins
Data Acceptance Complete	Occurs when a Workflow Administrator clicks the **Accept Data** button after reviewing the data submitted during a request's Data Entry Step.

Table 22-19 • Activities Listed in the Activity Report (cont.)

Activity	Description
Data Changed	Occurs when a user clicks **Update** after editing data that was submitted as part of a request.
Data Edit	Occurs when a user clicks the name of a Data Entry Workflow Step, and then clicks the **Edit Data** button.
Data Rejected	Occurs when a Workflow Administrator clicks the **Reject Data** button to reject the data submitted during a request's Data Entry Step. Each time data is rejected, three activities are recorded: • Data Acceptance Cancel • Data Rejected • Data Submission Begins
Data Submission Begins	Because the first Workflow Step of the first Workflow Phase of every Request is a Data Entry step, each time a Workflow Consumer or Workflow Administrator submits a new Request, this activity occurs when the **Submit** button is clicked. ***Note*** *• When Workflow Consumers submit a request, they are immediately prompted to enter the required data. However, when Workflow Administrators submit a request, they are not prompted to enter the required data until they click on the first workflow step of the first workflow phase on the* ***Workflow Request*** *page.*
Data Submission Complete	Occurs when a user clicks the **Submit** button after they have entered all of the initial data that is required for a request (the Data Entry Step of the first Workflow Phase). ***Note*** *• This event occurs simultaneously with the Data Acceptance Begins event.*
SLA Start	Occurs when a user clicks the **Start Clock** button on the **Workflow Request** page to restart monitoring of SLA time for the current workflow step.
SLA Stop	Occurs when a user clicks the **Stop Clock** button on the **Workflow Request** page to stop monitoring of SLA time for the current workflow step.
Workflow Phase Begins	Occurs when the last workflow step in the previous workflow phase is completed.
Workflow Phase Cancel	Occurs when all workflow steps in a workflow phase are rolled back.
Workflow Phase Complete	Occurs when the last Workflow Step of a Workflow Phase is completed.
Workflow Step Begins	Occurs when the previous Workflow Step in a Workflow is completed.

Table 22-19 • Activities Listed in the Activity Report (cont.)

Activity	Description
Workflow Step Cancel	Occurs when a Workflow is rolled back to previous Workflow Step, which cancels the completion state of all of the Workflow Steps between the current step and the one that is rolled back to.
Workflow Step Complete	Occurs when a Workflow Step is completed, one of the following events occurs (depending upon the **Step Type**): • **Data Entry/Edit**—Occurs when a user clicks **Submit** after entering the required data. • **Normal**—Occurs when a user clicks **OK** after entering time information on the **Step Validation** dialog box. • **Update History**—Occurs when a user clicks **OK** after entering information on the Update History dialog box to document a Workflow Step/Phase. • **Workflow Assignment**—Occurs when a user clicks **Apply** on the **Assignment Details** page after assigning a user to roles associated with this Request. • **Script Execution**—Occurs when a user clicks this Workflow Step name on the **Workflow Request** page, which launches a user-specified executable file. • **Custom Web Page**—Occurs when a user clicks this Workflow Step name on the **Workflow Request** page, which opens a user-specified URL address in a new browser window.
Workflow Step Rollback	Occurs when a user enters a reason for rollback and clicks the **Rollback** button on the **Rollback Workflow Item** dialog box (which is opened by clicking the check mark next to the name of a completed Workflow Step).

Information that Can Be Included in an Activity Report

Each time an activity occurs, the following information is recorded:

Table 22-20 • Available Activity Report Fields

Field	Description
Activity Date	Date and time that an activity occurred.
Activity Name	Name of event that was recorded. See Activities Displayed in the Activity Report for a complete list.
Activity Owner	User who was “assigned” to the Workflow Step that was active when the activity occurred; the user who performed the activity.
Workflow Name	Name of the request that the activity was associated with.
Data Major	Name of the data group that contains a data element that was modified.
Data Minor	Name of the data element that was modified.

Table 22-20 • Available Activity Report Fields

Field	Description
New Value	Modified value of the edited data element.
Old Value	Previous value of the edited data element.
Project Name	Name of project that the request associated with this Activity is associated with.
Workflow Major	Name of the Workflow Phase that contains the Workflow Step that was current when the activity occurred.
Workflow Minor	Name of the Workflow Step that was current when the activity occurred.

When defining an Activity Report, you choose which of these fields to include in the report. You can also choose to include any data that was entered for a request, and you can also filter the report based upon the value of one of the available report fields.

How to Create an Activity Report

To create a Custom Activity Report, perform the following steps.

Task ***To create an Activity Report:***

1. In the navigation bar, click **Create Workflow Request Activity Report**. The **Select report objects** panel of the **Create Workflow Request Activity Report** page opens.
2. Leave **Activities** selected and click **Next**. The **Select report fields** panel opens.
3. Select the report fields that you would like to include in the Activity Report and click **Next**. The **Define report filters** panel opens, where you can filter the data that you want to appear in the report.

   Note • *For a listing of the report fields available in the Activity Report, see Information that Can Be Included in an Activity Report.*
4. Click on a field in the tree and set its filter on the right side using the drop-down boxes and the text box, selecting appropriate logical conditions which are populated according to the selected field. Each time you create a filter, click **Add** to add the filter to the current filter conditions.

   Note • *Even though you may not have included all of the available report fields in this report, you can still filter the data using all of these report fields.*
5. Click **Test** to test the created query for your report.
6. When you are satisfied with the filter conditions, click **Next**. The **Templates** panel opens, listing all available Templates.

7. Expand the templates in the tree and select the data that you want to include in the report. All of the data groups and data elements associated with the selected template are listed.

   To display only those templates that are in use in the **Available Templates** list, select the **Templates in use only** option.

8. Click **Next**. The **Specify general information** panel opens.
9. Enter a **Report name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.
10. Select the Administrator and Consumer roles that you want to have permission to view this report.
11. Click **Next**. The **Save and preview report** panel opens, which displays all the information needed to create the report.
12. Click **Save and preview**. The report is generated. This report is also saved and now appears in the list on the **All Reports** page.

Generating a Custom SQL Query Report

Edition • *This feature is available in Workflow Manager only.*

A **Custom SQL Query Report** is a report on data generated by Workflow Manager that is defined by entering an SQL query in the Report Wizard. To generate a Custom SQL Query Report, perform the following steps.

Task ***To generate a new Custom SQL Query report:***

1. In the navigation bar, click **Create Custom SQL Query Report**. The **Enter SQL Query** panel of the **Create Custom SQL Query Report** page opens.
2. Enter the SQL query which is to retrieve data for your report into the **Custom SQL Query** text field to retrieve the data for this report. Click the **Test** button to verify the query syntax.

   Note • *Refer to Enter SQL Query Panel for information about Workflow Manager database tables which you might want to return data from.*

3. Click **Next**. The **Specify general information** panel opens.
4. Enter a **Report name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.
5. Decide which roles should have the right to view your report, selecting them from the **Roles** tree
6. Click **Next**. The **Save and preview report** panel opens, summarizing the information that will be used to create your report.
7. Click **Save and preview**. Your report is generated. The report will now also appear in the list on the **All Reports** page.

Generating a Custom Stored Procedure Report

You have the option of generating a Workflow Manager report using a stored procedure. A **Custom Stored Procedure Report** is a report on data generated by Workflow Manager that is defined by specifying a stored procedure.

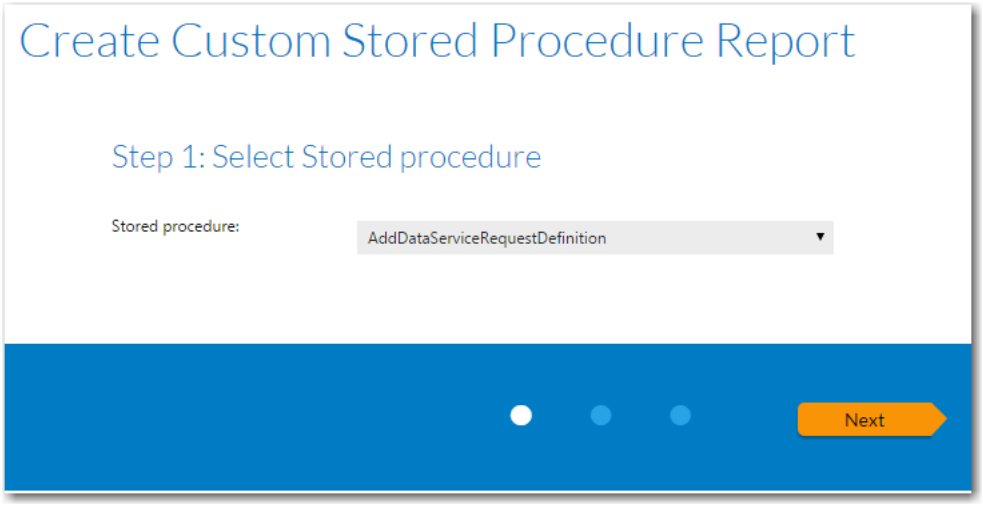
To generate a Custom Stored Procedure report, perform the following steps.

Task ***To generate a Custom Stored Procedure report:***

1. Open the `AMS_CustomReports` table and enter the names of the stored procedures you want to use to generate reports.

   Note • *For more information on stored procedures, see SQL Stored Procedures in Microsoft TechNet.*

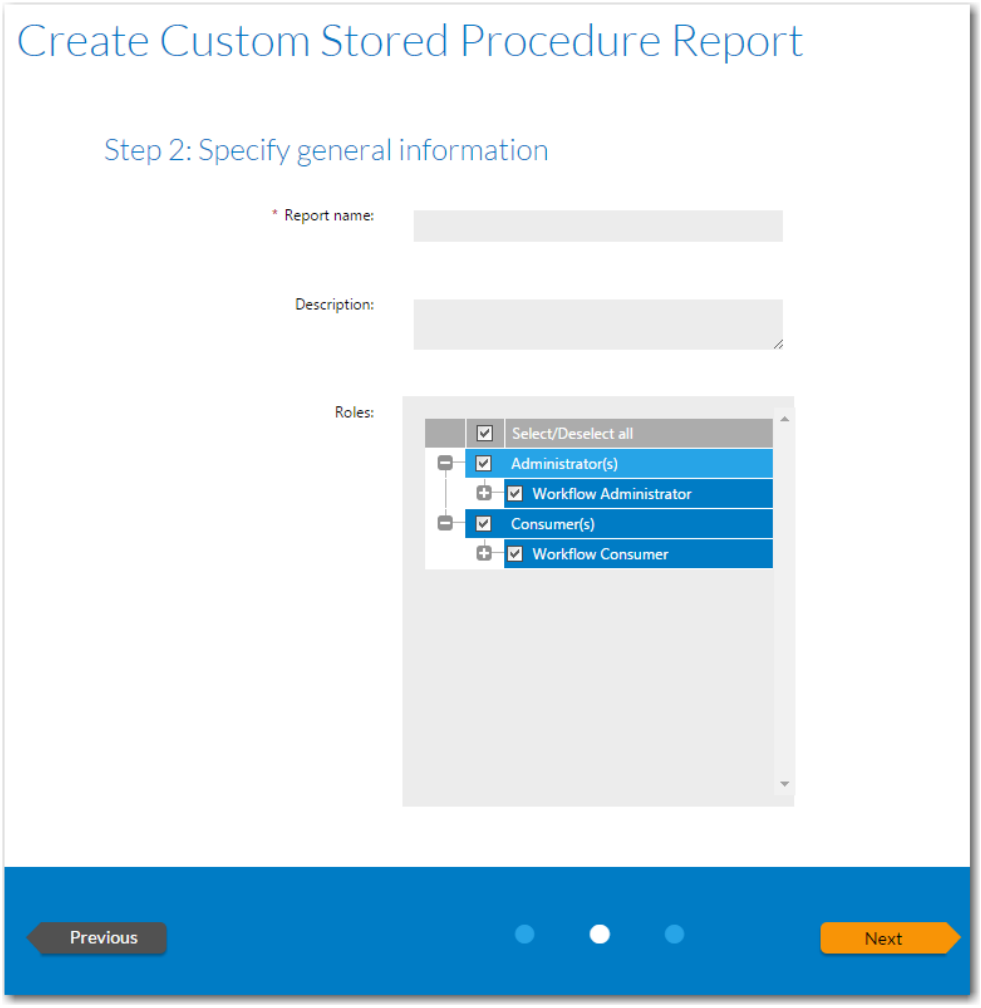
2. In the navigation bar, click **Create Custom Stored Procedure Report** on the **Reports** menu. The **Step 1: Select Stored procedure** of the **Create Custom Stored Procedure Report** page opens.



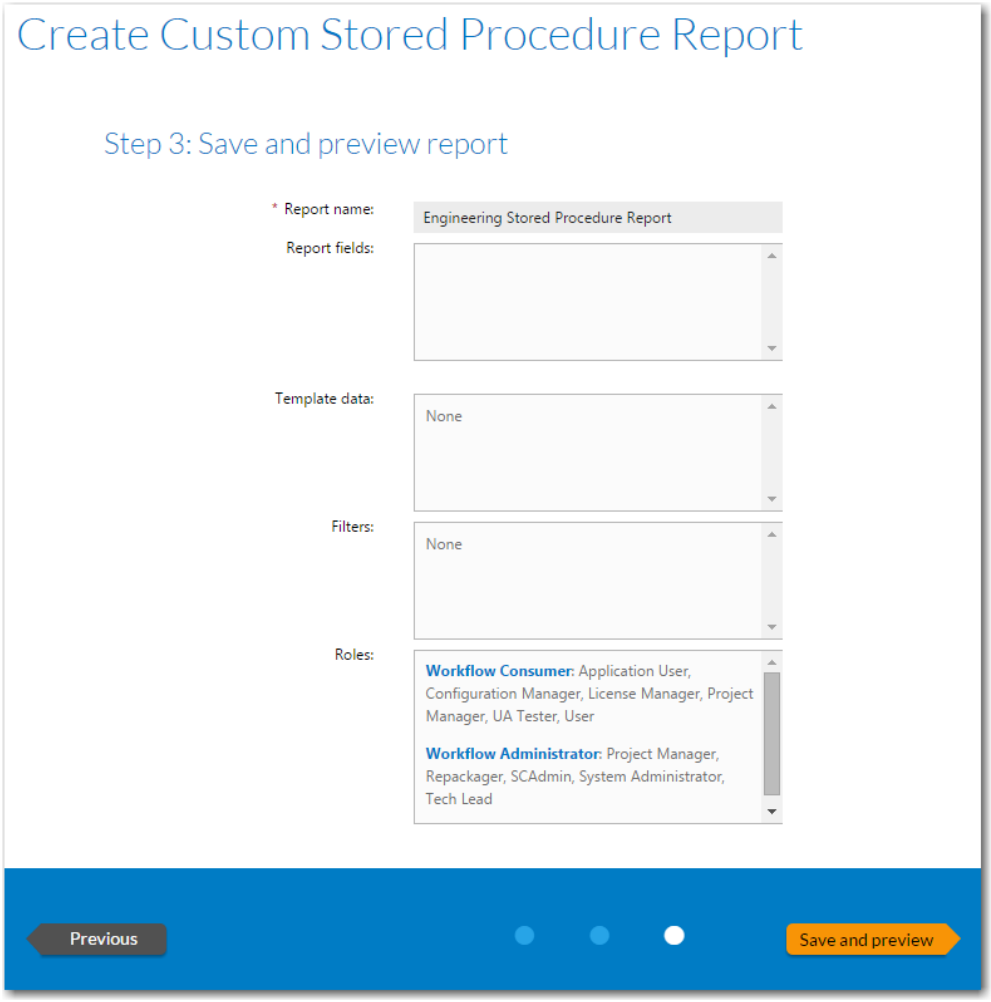
   The contents of this panel is determined by the selected stored procedure.

   Note • *This panel is customizable per customer need. The filters shown on the screen are based up on the parameters required by the stored procedure.*

3. Select a stored procedure from the list and specify any other requested information.
4. Click **Next**. The **Step 2: Specify general information** panel opens.



5. In the **Report name** field, enter a name to identify this report. This name will be listed on the **All Reports** page.
6. Enter a **Description** to identify the purpose of this report.
7. In the **Roles** section, select those roles that you want to assign permission to view this report.
8. Click **Next**. The **Step 3: Save and preview report** panel opens.



9. Click **Save and preview**. The report is displayed.

   The report is now saved and available to view by users with appropriate permission.

Wildcard Support in Reports SQL Queries

Edition • *This feature is available in Workflow Manager only.*

In Reports searches, data is always filtered using the SQL `LIKE` operator. You can combine the `LIKE` operator with wildcard characters to achieve the following results:

Table 22-21 • Wildcard Support in Reports Queries

Situation	Rule
When no wildcards are used	If you do not enter a wildcard character in the **Search** box, then Reports performs a "LIKE" search, meaning that it will identify any occurrence of your search text in the field being searched. For example, if you enter the word **`test`** in the **Search** box, Reports would interpret this as `*test*` and would return any records containing the word `test` (including `MyTestFile` and `TestFile`).
When wildcards are used	You can specify a * wildcard in the **Search** box to narrow the search results. For example, if you want to return all records which contain the word `test` but do not begin with it, enter `*test` in the **Search** box. Then records with the word `MyTest` would be returned, but not records with the word `TestFile`.

Sample SQL Queries Used to Generate Project and Workflow Request Reports

Edition • *This feature is available in Workflow Manager only.*

The following queries are used to generate the built-in Project and Workflow Requests reports. These sample queries might be helpful to refer to when you are creating your own custom reports.

Note • *Note that* `DateTimeHelper.GetUniversalDateTime()` *is used in some of the queries for demonstration purposes only, it is not valid SQL syntax.*

Projects Completed On-Time

The following is a sample query to generate data on projects that were completed on time.

```
SELECT ApplicationID FROM AMS_Application A, AMS_ApplicationStatus AST WHERE A.AppStatusID= AST.StatusID
    AND AST.IsActive = 1 AND A.StatusSummary =90 AND A.DueDate >= A.ApplicationEndDate AND A.ContractID
    = 'd135b5ae-8ac0-42b4-a5bc-e105c11b5e13'
```

Projects Completed Late

The following is a sample query to generate data on projects that were completed late.

```
SELECT ApplicationID FROM AMS_Application A, AMS_ApplicationStatus AST WHERE A.AppStatusID= AST.StatusID
    AND AST.IsActive = 1 AND A.StatusSummary =90 AND A.DueDate < A.ApplicationEndDate AND A.ContractID =
    'd135b5ae-8ac0-42b4-a5bc-e105c11b5e13'
```

On Time Workflow Requests

The following is a sample query to generate data on workflow requests that were completed on time.

```
GetOnTimeActiveApplications
```

```
SELECT ApplicationID FROM AMS_Application A, AMS_ApplicationStatus AST WHERE A.AppStatusID =
    AST.StatusID AND AST.IsActive = 1 AND A.StatusSummary <> 90 AND
    DateTimeHelper.GetUniversalDateTime() < A.RiskDate AND A.ContractID = 'd135b5ae-8ac0-42b4-a5bc-
    e105c11b5e13'
```

At Risk Workflow Requests

The following is a sample query to generate data on workflow requests that at risk of being completed late.

```
GetAtRiskApplicationCount
```

```
SELECT Count(*) FROM AMS_Application WHERE StatusSummary <> 90 AND DateTimeHelper.GetUniversalDateTime()
    > RiskDate AND DateTimeHelper.GetUniversalDateTime() < DueDate AND ContractID = 'd135b5ae-8ac0-42b4-
    a5bc-e105c11b5e13'
```

Exporting Report Data from Reports

You can choose to export data by using the export feature that is built into all lists to export the data in PDF, RTF, XLSX, or CSV format.

Reports Reference

This section includes reference information on the following pages, views, and reports:

- All Reports Page
- Search Packages Page
- Application Catalog Reports Page
- Package Report
- Reports Wizard

All Reports Page

The **All Reports** page provides access to the following reports:

- **Workflow Manager**—System Reports (Projects Report, Project SLA Report, Request Detail Report, Workflow Requests Summary Report, Workflow Phases SLA Report, and Workflow Steps SLA Report), Custom Workflow Manager Report, Workflow Request Activity Report, Custom SQL Query Report, and Custom Stored Procedure Report.
- **AdminStudio**—Archived Package Reports, Custom SQL Query Report, and Custom Stored Procedure Report.

The **All Reports** page can be viewed in either a card view or list view.

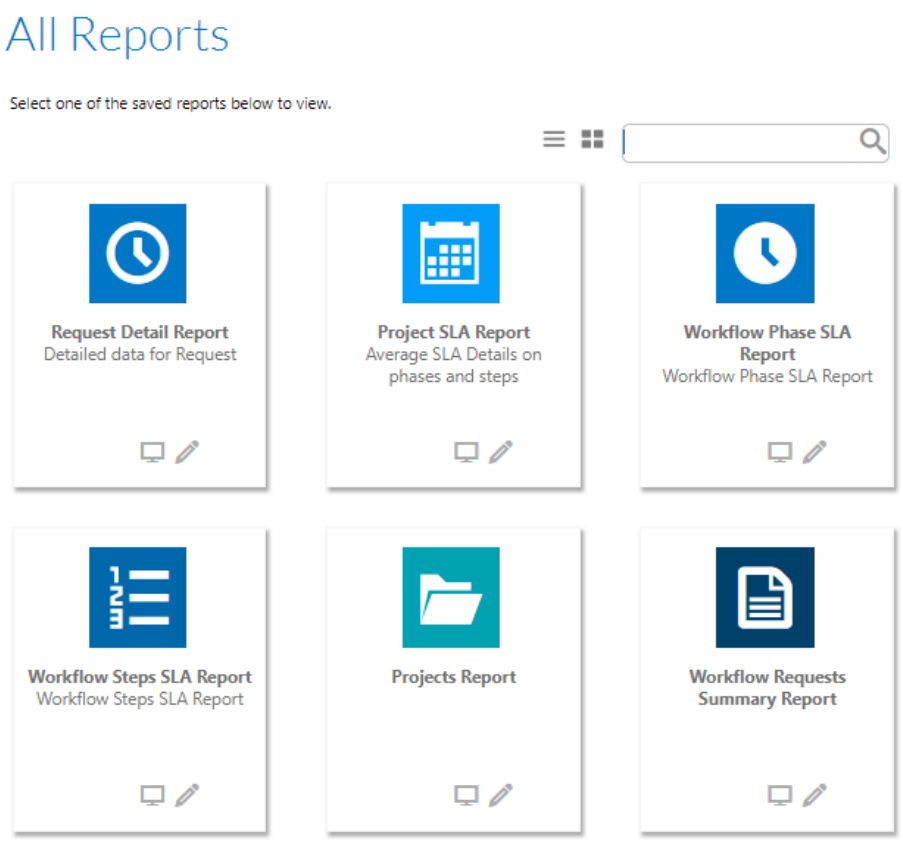


Figure 22-21: System Reports on the All Reports Page (Card View)

You can click on the icon on the top right to toggle this view to list view.

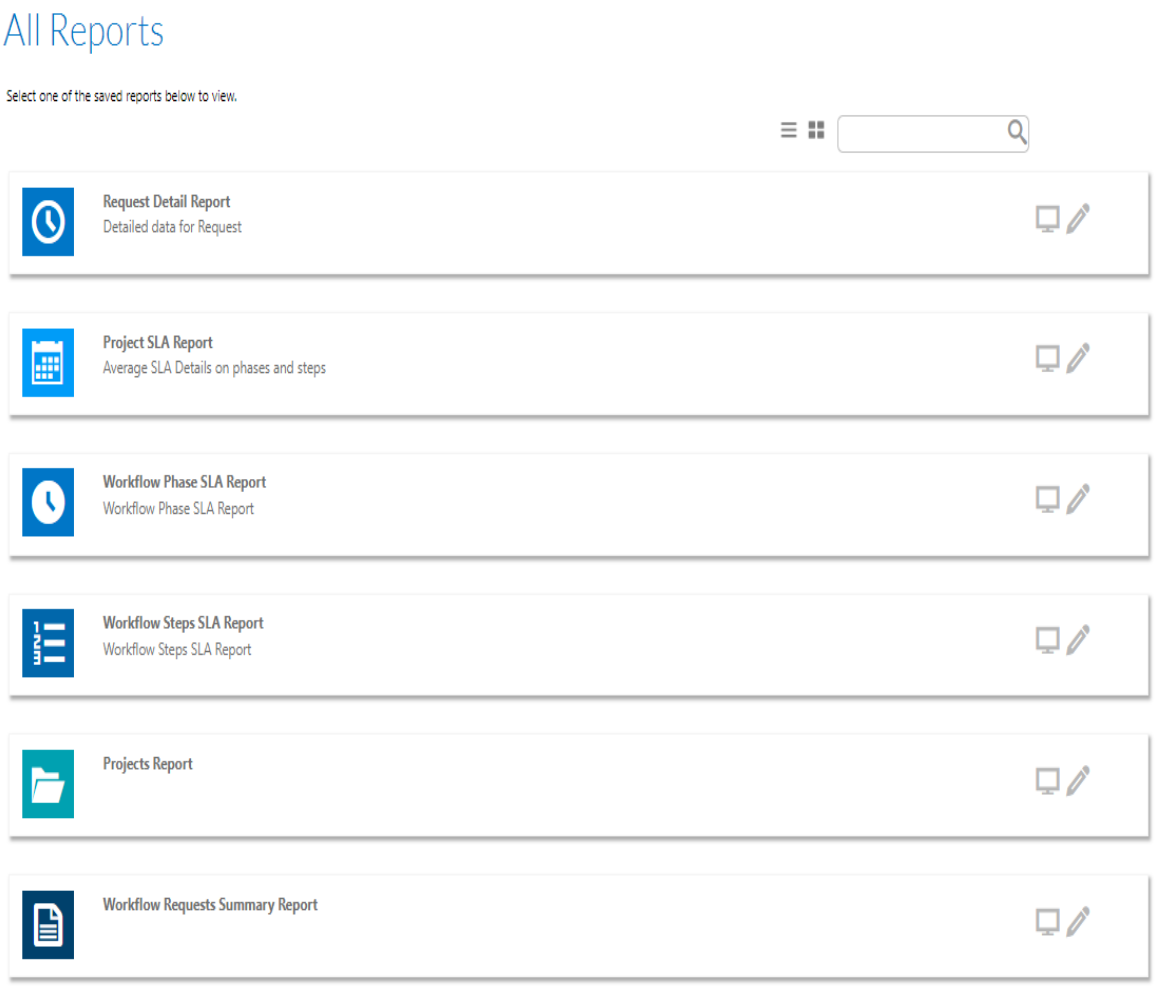


Figure 22-22: Reports on the All Reports Page (List View)

The **All Reports** page includes the following icons:

Table 22-22 • All Reports Page Icons

Icons	Description
	Click to edit the report.
	Click to view the report.
	Click to delete the report.
	Use to toggle between list and card view.

Available Reports

From the **All Reports** page, you can choose to create a new custom report, view a custom report that was already created, or view the following reports:

- Projects Report
- Workflow Requests Summary Report
- Request Detail Report
- Project SLA Report
- Workflow Phases SLA Report
- Workflow Steps SLA Report

Viewing an Existing Custom Report from the All Reports Page

To view a report, click on the report name or the View Report icon:

Figure 22-23: View Report Icon

Standard Reports

Edition • *This page is available in Workflow Manager only.*

Workflow Manager includes the following standard reports;

- Projects Report
- Workflow Requests Summary Report
- Request Detail Report
- Project SLA Report
- Workflow Phases SLA Report
- Workflow Steps SLA Report

Projects Report

Edition • *This page is available in Workflow Manager only.*

A **Projects Report** groups projects by customer and returns summarized information including the progress and Service Level Agreement (SLA) status of workflow requests. You can choose to return information about one project or all of a company's projects.

To view a Projects Report, click **Projects Report** on the **Reports** menu. The **Projects Report** page opens.

Table 22-23 • Projects Report Page Options

Option	Description
Company	Select the company that you want to report on.
Project	Choose one of the company's projects from this list, or select **** View for all **** to return information about all of a company's projects.
SLA status	To return only workflow requests with specific SLA Status values, choose one or more of: **Completed On Time**, **Completed Late**, **On Time**, **At Risk**, or **Late**. Select all of these if you want to report on all workflow requests.
View Report	Click to generate and display the Projects Report.

Projects Report Information

A Projects Report lists project summary information by Workflow Consumer, including request and Service Level Agreement (SLA) status. You can choose to include information on one project or all of a Workflow Consumer's projects.

The following information is included in the Projects Report:

Table 22-24 • Projects Report Information

Item	Description
Project Summary	General project properties and SLA settings.
Workflow Requests by SLA Status	A table listing the number of workflow requests associated with this project, grouped by SLA Status: **Completed On Time**, **Completed Late**, **On Time**, **At Risk**, and **Late**.
Workflow Request Progress Overview	This section includes the following pie charts: • **SLA Compliance Summary**—Pie chart illustrating the number of workflow requests associated with this project with a given SLA status. • **Current Phase Status**—Pie chart illustrating the number of active workflow requests associated with this project in a given phase.
Workflow Request Progress In-Progress Workflow Requests By Current Workflow Phase	Lists of in-progress workflow requests in each workflow phase, grouped by SLA status.
View Workflow Requests Summary	Click to open the Workflow Requests Summary Report for this project.

Workflow Requests Summary Report

Edition • *This page is available in Workflow Manager only.*

A **Workflow Requests Summary Report** lists request summary information by company, including progress status and Service Level Agreement (SLA) status. You can choose to include information on one request or all of a company's requests. You can also filter the report by SLA Status.

To view a Workflow Requests Summary Report, click **Workflow Requests Summary Report** on the **Reports** menu. The **Workflow Requests Summary Report** opens.

Table 22-25 • Workflow Requests Summary Report Page Options

Option	Description
Company	Select the company that you want to view a report on.
Include	Select one of the following options to specify which requests to include in this report: • **Single Workflow Request**—Select this option to return information about only one workflow request, which you select from the associated combo box. • **Multiple Workflow Requests**—Select this option to return all workflow requests with specific SLA status values. Choose one or more of **Completed On Time**, **Completed Late**, **On Time**, **At Risk**, or **Late** to indicate which workflow requests you wish to return. Select all of these options to return all workflow requests associated with a project
Additional Metadata Filter Conditions	If you chose to generate a report about a single workflow request, this check box becomes visible. Select this check box to return filter your report by the values provided by people as they complete the data elements in a workflow request. If you select this checkbox, the metadata filter fields appear, and you should do the following: • **Metadata Field**—Select the data element that you want to filter by from the list. • **Conditional Value & Values List**—Enter a value for the selected **Metadata Field** in the **Condition Value** box or select a value from the **Values List** (when available). • **Operator**—Select the appropriate **Operator** from the list (**AND** or **OR**). You can filter by up to four **Metadata Fields**.
View Report	Click to generate and display the Workflow Requests Summary Report.

Workflow Requests Summary Report Information

A **Workflow Requests Summary Report** lists request summary information by company, including progress status and Service Level Agreement (SLA) status. You can choose to include information on one request or all of a company's requests. You can also filter the report by SLA Status.

The following information is included in a Workflow Requests Summary Report:

Table 22-26 • Workflow Requests Summary Report Information

Item	Description
Administrator Company	Identifies the administrator company associated with these workflow requests.
Selected Workflow Requests	List of the SLA status values included in this report.
Workflow Requests	List of workflow requests associated with this administrator company. The list includes the following information: • **Workflow**—Hyperlinked workflow request name. Click the name to open the **Workflow Request** page for that workflow request. • **Start Date**—The date the workflow request was submitted. • **Due Date**—The date the workflow request is due to be completed, in order to meet SLA requirements. • **End Date**—The actual date the workflow request was completed. • **Progress Status**—The phase type of the workflow request's current workflow phase. • **SLA Status**—The workflow request's SLA Status. One of **On Time**, **At Risk**, **Late**, **Completed Late**, or **Completed On Time**. • **Elapsed Time**—The amount of time elapsed since the workflow request was submitted, excluding any periods when the SLA clock was stopped. • **SLA Time**—The amount of time elapsed since the workflow request was submitted, excluding any periods when the SLA clock was stopped, and excluding any time spent performing a workflow step that was not tracked. • **Company**—The name of the workflow consumer's company. • **Project**—Hyperlinked name of the project associated with this workflow request. Click the name to open the **Projects Report** for that project.

Request Detail Report

The Request Detail Report helps you analyze the SLA delivery time for any workflow request. You can use the fields at the top to filter the list of workflow requests displayed in this report, such as to display only workflow requests from a particular project, or just those using a particular workflow template, etc.

You open the **Request Detail Report** page by clicking **Request Detail Report** on the **Reports** menu.

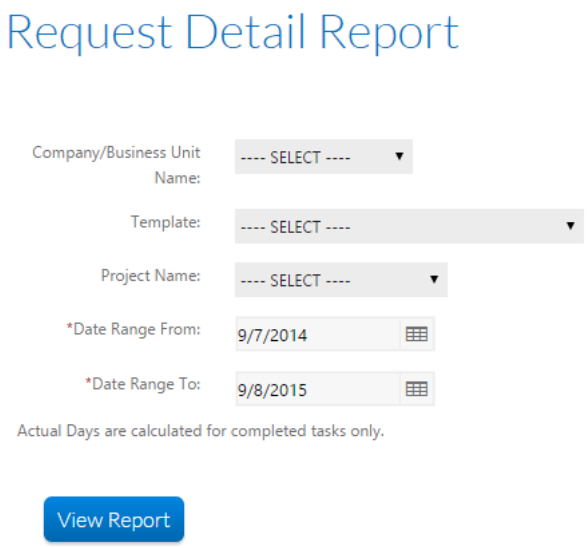


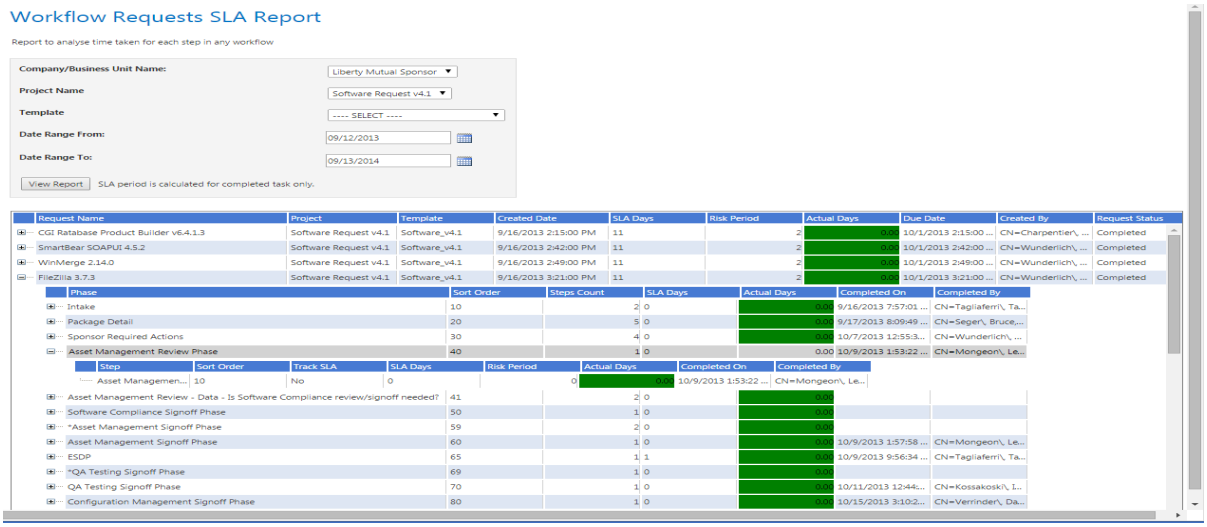
Figure 22-24: Request Detail Report / Initial View

The initial view of the Request Detail Report includes the following options:

Table 22-27 • Request Detail Report Options

Option	Description
Company/Business Unit Name	Select the name of the company or business unit that you want to view SLA information for.
Template	Select the name of the workflow template used by the workflow requests that you want to view SLA information for. ***Important*** • *To generate a report that lists SLA data for all workflow requests during a specific date range, do not make a selection from the* ***Template*** *list.*
Project Name	Select the name of the project that is associated with the workflow requests that you want to view SLA information for. ***Important*** • *To generate a report that lists SLA data for all projects during a specific date range, do not make a selection from the* ***Project Name*** *list.*
Date Range From **Date Range To**	Identify the date range for which you want to view workflow request data.
View Report	Click to generate the Request Detail Report using the specified criteria.

In the Request Detail Report, workflow requests are listed, along with summary SLA information for all phases in that workflow request. Use the plus signs to expand the listing to view the SLA data for workflow phases and steps in a particular workflow request.



Project SLA Report

You can generate a **Project SLA Report** to measure and report on the SLA status for a specific project, or for all projects, during a specific date range. This helps you helps you analyze the delivery time for any completed project, and identify bottlenecks and weak points in your process.

Using this report, you can view projects within a specific date range, and then drill down from project level to workflow requests across both phases and steps to see the SLA status at each level.

You open the **Project SLA Report** page by clicking **Project SLA Report** on the **Reports** menu.

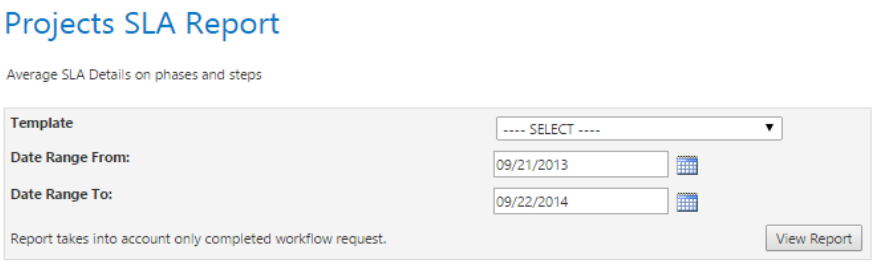


Figure 22-25: Project SLA Report / Initial View

The initial view of the Project SLA Report includes the following options:

Table 22-28 • Project SLA Report Options

Option	Description
Template	Select the name of the Workflow Template used by the project or projects that you want to view SLA information for. ***Important*** *• To generate a report that lists SLA data for all projects during a specific date range, do not make a selection from the* ***Template*** *list.*
Date Range From **Date Range To**	identify the date range for which you want to view project data.
View Report	Click to generate the Project SLA Report using the specified criteria.

In the Project SLA Report data area, click the plus signs to expand the listing to view SLA data across phases and steps for a specific project.

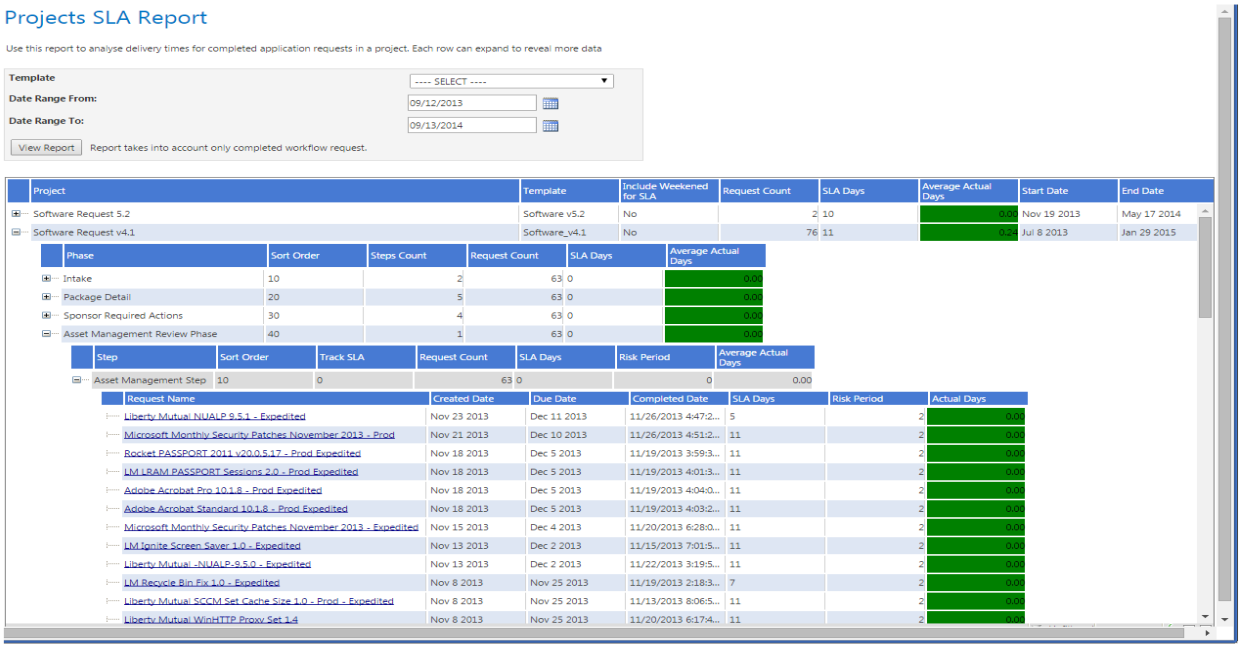


Figure 22-26: Project SLA Report

To view SLA information on a specific workflow request, click the hyperlinked **Request Name** to open the SLA Details by Phase and Workflow Step Subreport for that workflow request.

Subreport: SLA Details by Phase and Workflow Step

The **SLA Details by Phase and Workflow Step** report, which is opened by clicking on a workflow request name in the **Project SLA Report**, shows delivery time for one completed workflow request across both phases and steps.

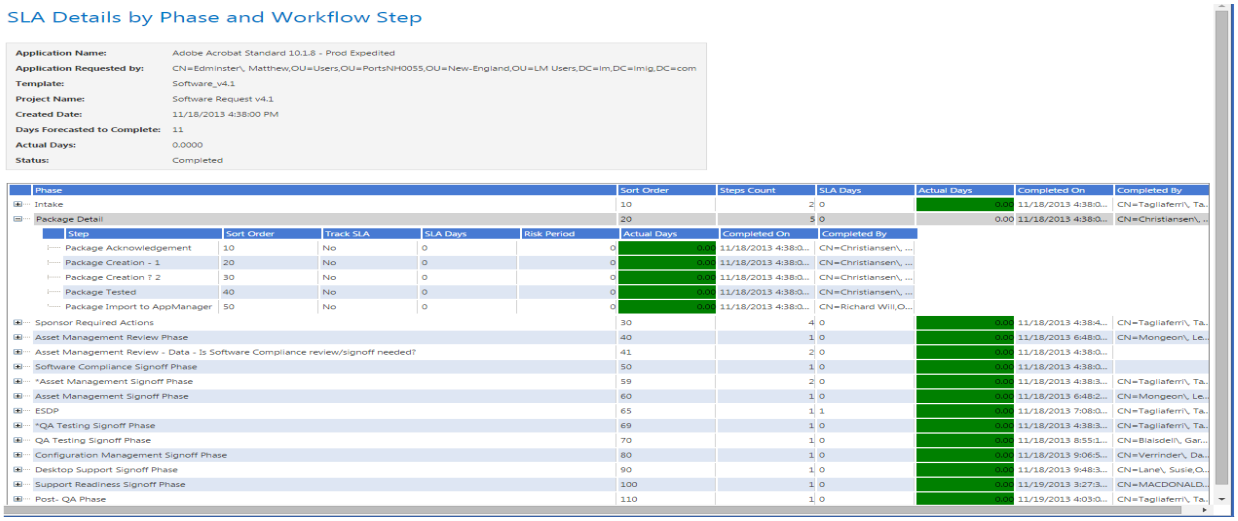


Figure 22-27: SLA Details by Phase and Workflow Step

To view the SLA data for other workflow steps in that workflow request, use the plus signs to expand the listing.

Note • *Workflow Manager stores SLA time in the `SLAPeriod` field of the `ams_applicationItem` table. This value is calculated every time the step is completed. Same `SLAPeriod` field in `ams_application` table is stored to save SLAPeriod for the whole application. This field is automatically calculated when the application is completed.*

Workflow Phases SLA Report

A **Workflow Phases SLA Report** lists all workflow phases for which step-level SLA tracking is being performed along with their **SLA Status**. SLA (Service Level Agreement) time tracking is used to determine the status of a workflow phase (or workflow request) in relationship to its SLA due date as either: In Progress, On Time, At Risk, Late, Completed on Time, or Completed Late.

Note • *For information on enabling workflow-phase level SLA tracking, see "Tracking a Workflow Request or Workflow Phase's SLA Status" in the Workflow Manager Help Library.*

To open the Workflow Phases SLA Report, click **Workflow Phases SLA Report** on the **Reports** menu.

The **Workflow Phases SLA Report** lists the following details:

Table 22-29 • Workflow Phases SLA Report

Property	Description
Only include Workflow Phases in active Workflow Requests	By default, only workflow phases from currently active workflow requests are listed. To display the SLA status of workflow phases from all workflow requests, even those that have been completed, clear the selection of this option and click **Refresh Report**.

Table 22-29 • Workflow Phases SLA Report

Property	Description
Project	Project associated with this workflow request.
Workflow	Workflow request associated with this workflow phase.
Workflow Phase	Name of workflow phase that is being tracked for SLA status.
Due Period	The elapsed time (in days) after the workflow phase becomes the current phase that it should be completed in order to be SLA compliant. If it is not completed by this date, its SLA status would be `Late` and an email alert would be sent out.
Risk Period	The elapsed time (in days) after which this workflow phase should be considered at risk of not being completed on time (corresponds to SLA status of `At Risk`).
Start Time	Time this workflow phase was initiated.
End Time	Time that this workflow phase was completed.
Due Date	Scheduled due date for this workflow phase based upon its **Due Period**.
Risk Date	Scheduled risk date for this workflow phase based upon its **Risk Period.**
SLA Status	Identifies the workflow phase's SLA Status. SLA (Service Level Agreement) time tracking is used to report the status of a workflow request and/or a single workflow phase with respect to its SLA due date, as one of: `In Progress`, `On Time`, `At Risk`, `Late`, `Completed on Time`, or `Completed Late`.

Workflow Steps SLA Report

A **Workflow Steps SLA Report** lists all workflow steps for which step-level SLA tracking is being performed along with their **SLA Status**. SLA (Service Level Agreement) time tracking is used to determine the status of a workflow step (or workflow request) in relationship to its SLA due date as either: In Progress, On Time, At Risk, Late, Completed on Time, or Completed Late.

Note *• For information on enabling workflow-step level SLA tracking, see "Tracking a Workflow Request or Workflow Step's SLA Status" in the Workflow Manager Help Library.*

To open the Workflow Steps SLA Report, click **Workflow Steps SLA Report** on the **Reports** menu.

The **Workflow Steps SLA Report** lists the following details:

Table 22-30 • Workflow Steps SLA Report

Property	Description
Only include Workflow Steps in active Workflow Requests	By default, only workflow steps from currently active workflow requests are listed. To display the SLA status of workflow steps from all workflow requests, even those that have been completed, clear the selection of this option and click **Refresh Report**.
Project	Project associated with this workflow request.
Workflow	Workflow request associated with this workflow step.
Workflow Step	Name of workflow step that is being tracked for SLA status.
Due Period	The elapsed time (in days) after the workflow step becomes the current step that it should be completed in order to be SLA compliant. If it is not completed by this date, its SLA status would be `Late` and an email alert would be sent out.
Risk Period	The elapsed time (in days) after which this workflow step should be considered at risk of not being completed on time (corresponds to SLA status of `At Risk`).
Start Time	Time this workflow step was initiated.
End Time	Time that this workflow step was completed.
Due Date	Scheduled due date for this workflow step based upon its **Due Period**.
Risk Date	Scheduled risk date for this workflow step based upon its **Risk Period.**
SLA Status	Identifies the workflow step's SLA Status. SLA (Service Level Agreement) time tracking is used to report the status of a workflow request and/or a single workflow step with respect to its SLA due date, as one of: `In Progress`, `On Time`, `At Risk`, `Late`, `Completed on Time`, or `Completed Late`.

Search Packages Page

From the **Search Packages** page, you can select or search for a specific package, and then generate a detailed Package Report. On this page, you can filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, which are grouped into three categories:

- **Package Attributes**—Search by properties assigned to the Windows Installer package. See Package Attributes.
- **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the Windows Installer package. See Package Content.
- **Application Request Attributes**—Search by information related to a package's associated request. See Application Request Attributes.

To filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, enter values in the criteria fields that you want to search on, and click **Search**. The packages that meet *any of the criteria* are then listed in the package tree in alphabetical order and are no longer grouped.

Package Attributes

You can search for packages in a catalog based on one or more of any of the following Package attribute metadata:

Table 22-31 • Package Attribute Search Fields

Metadata	Description
Package Code	Enter the GUID that identifies a particular Windows Installer `.msi` package. The Package Code associates an `.msi` file with an application or product and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F).
Product Code	Enter the GUID that uniquely identifies the particular product release of the package. The ProductCode is a Windows Installer property and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F).
Upgrade Code	Enter the GUID that identifies the family of products that are in the same upgrade path. The **UpgradeCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXsXXXXX} where each X character is a hex digit (0 through 9 or uppercase A through F). ***Note*** *• Each stand-alone product usually has its own* ***UpgradeCode*** *GUID. Every version of XYZ Product typically uses the same GUID for the* ***UpgradeCode****. In other words, Product A Version 1.0 has the same UpgradeCode as Product A Version 2.0, but has a different UpgradeCode than Product B.*
Setup File Name	Name of the Windows Installer (.msi) file that was imported into the Application Catalog.
Comments	Enter the text of any comments associated with the package.
Extended Attributes	Enter the value of any of the Extended Attributes associated with the package.

Package Content

You can search for packages in a catalog based on one or more of any of the following Package Content metadata

Table 22-32 • Package Content Search Fields

Metadata	Description
File	Enter the file name of one of the files in the Windows Installer package.
Registry Key	Enter a registry key to search on.
Registry Value	Enter a registry value to search on.
INI File	Enter any changes to an `.ini` file that are made when the product is installed.
Shortcut	Enter any shortcuts that are created when the product is installed.

Application Request Attributes

You can search for packages in a catalog based on one or more of any of the following attributes of the package's associated request:

Table 22-33 • Request Attributes Search Fields

Metadata	Description
Name	Enter the name of the package's associated request.
Upload Date	Date the request was submitted.
Due Date	Enter the date the request is scheduled to be completed, based upon its value for **Application Due Period**.
Risk Date	Enter the date at which the request's status will change to **At Risk**, which is based upon its value for **Application At Risk Period**.
Due Period	Enter, in days, the length of time this request needs to be completed in order to meet its project's Service Level Agreement (SLA) requirements.
End Date	Enter the date the request was completed.

Application Catalog Reports Page

On the **Application Catalog Reports** page, you can view a wide array of reports containing summary information on Windows Installer and virtual packages in the AdminStudio Application Catalog. These reports give you insight into the readiness of those packages for distribution and for conversion to virtual packages.

You open the **Application Catalog Reports** page by selecting **Application Catalog Reports** in the **AdminStudio Reports** subgroup of the **Reports** group in the navigation bar.

- Viewing the AdminStudio Application Catalog Reports
- Exporting a Report in PDF, Excel, or Word Format

Viewing the AdminStudio Application Catalog Reports

On the **Application Catalog Dashboard** page, you can view a wide array of reports containing summary information on Windows Installer and virtual packages in the AdminStudio Application Catalog. These reports give you insight into the readiness of those packages for distribution and for conversion to virtual packages.

You open the **Application Catalog Dashboard** page by selecting **Application Catalog Dashboard** in the **AdminStudio Reports** subgroup of the **Reports** group in the navigation bar.

For example, **Application Readiness Dashboard** report is opened and shown in below figure.

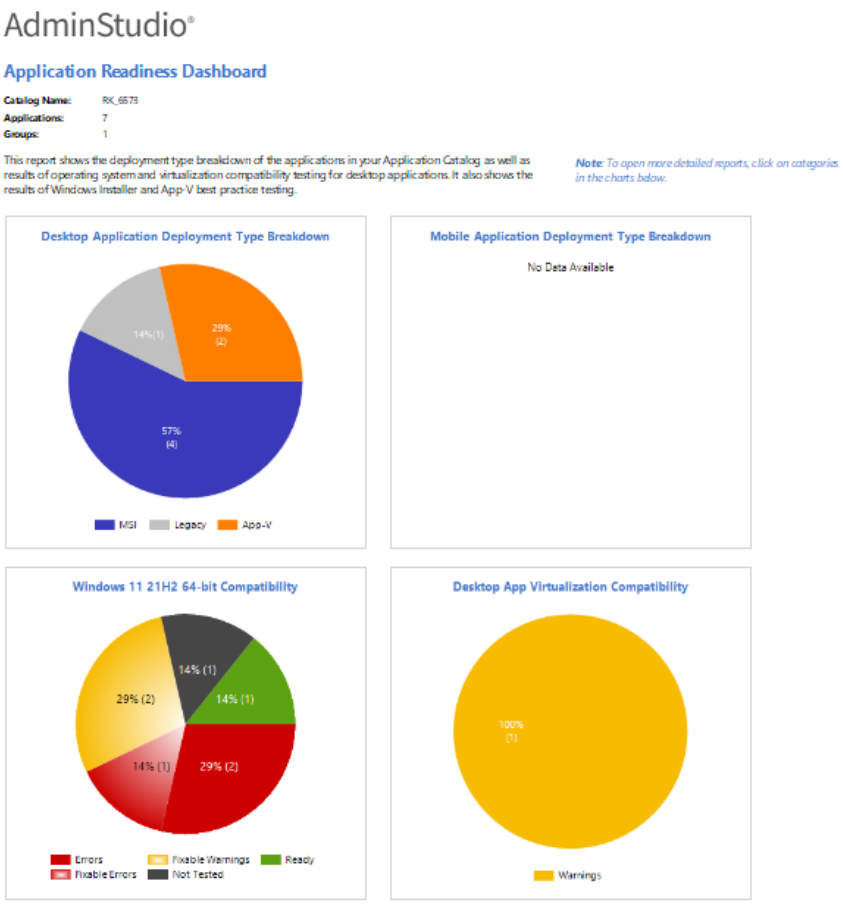


Figure 22-28: Application Readiness Dashboard Report

You switch between reports by selecting the report name from the **Select Report** list.

The available reports include test results from operating system compatibility, remote application publishing compatibility, installer best practices testing, and application conflict testing. They also include information about the App-V packages in your Application Catalog, as well as ConfigMgr (Formerly called as System Center Configuration Manager) deployment information.

For most reports, detailed sub-reports are available by clicking on one of the categories of the pie bar chart, on one of the numbers in an issue count column, or on a package name. Click on the available hyperlinks until you have explored all of the levels of the report.

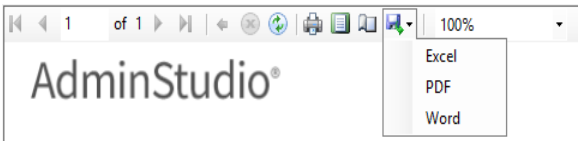
For more information, see Viewing Application Testing and Analysis Reports on the Reports Tab in the AdminStudio Help Library.

Exporting a Report in PDF, Excel, or Word Format

You can save any of the reports on the **Application Catalog Reports** page (and any of the drill-through reports) in PDF, Microsoft Excel, or Microsoft Word format.

Task ***Saving a report:***

1. View the report that you want to save.
2. In the toolbar, click the **Save** icon.



3. From the menu, select either **Excel**, **PDF**, or **Word**. The report is exported and you are prompted for a location to store the report.
4. Specify a location and click **Save**.

Note • *You can also print the currently viewed report by clicking the **Print** icon in the toolbar.*

Package Report

You can generate AdminStudio Package Reports on the **Search Packages** page, which is opened by clicking **Search Packages** under the **AdminStudio Reports** subgroup of the **Reports** group in the navigation bar. On the **Search Packages** page you can perform a search of all of the applications in the Application Catalog to locate the package you would like to generate a report for.

A **Package Report** lists detailed package information for packages of the following deployment types:

- Microsoft Windows Installer packages
- Microsoft App-V virtual packages
- Apple iOS mobile apps (local and public store)
- Google Android mobile apps (local and public store)

In a Package Report, the information is presented in a tabbed interface, as described in Navigating Through a Package Report. A Package Report includes the following major sections:

- Package Summary Information View

- Files View
- Registry View
- Shortcuts View
- ODBC Drivers View
- ODBC DS View
- Extended Attributes View
- Validation View
- Conflicts View
- History View
- Dependencies View
- Properties View

Note • *See also see Information Included in Package Reports.*

Note • *Additional information may be available for App-V packages.*

Package Summary Information View

The initial view (Page 1) of a Package Report is the **Package Summary Information** view, and it lists the following information:

Table 22-34 • Package Report / Package Summary Information

Item	Description
Product Name	Name assigned to the package.
Manufacturer	Company that authored the package.
Import Date	The date and time the package was imported into the Application Catalog.
Unresolved Conflicts	The number of detected conflicts, generated during conflict analysis of this package, which have not yet been resolved—either automatically or manually.
Product Version	Version of package that is recorded in the package's Windows Installer file.
Product Language	Decimal-based code identifying the language that this software package was authored for. For example, English is 1033, German is 1031, and Japanese is 1041.
In Software Repository	Indicates whether or not this package and its associated files are managed by the Software Repository.

Files View

The **Files** view lists the all of the files included in the selected package, and the location where these files will be installed. For each file, the following information is listed:

Table 22-35 • Package Report / Files Information

Item	Description
File	Name of file included with this package.
Target Directory	Name of directory where the file is installed.
Version	Version number of the file.
File Size	Size of the installed file.
Component	Component that the file is associated with.

Registry View

The **Registry** view lists the registry entries that will be created when this package is installed. For each registry entry, the following information is listed:

Table 22-36 • Package Report / Registry Information

Item	Description
Root	Identifies the predefined “root” key that contains the registry entry.
Key	A registry key.
Name	Name identifying the registry entry.
Value	The string of data that defines the value of the key.
Component	Package component that the registry entry is associated with.

Shortcuts View

The **Shortcuts** view lists all of the shortcuts that will be created when this package is installed. For each shortcut, the following information is listed:

Table 22-37 • Package Report / Shortcuts Information

Item	Description
Name	Name identifying the shortcut.
Target Directory	Directory and executable that the shortcut invokes.

Table 22-37 • Package Report / Shortcuts Information (cont.)

Item	Description
Component	Component associated with the shortcut.

ODBC Drivers View

The **ODBC Drivers** view lists all of the Open Database Connectivity (ODBC) drivers in the package.

ODBC Resources are ones that involve interaction with databases. ODBC drivers are libraries that implement functions involving ODBC. Each database type has its own ODBC driver. For each ODBC driver, the following information is listed:

Table 22-38 • Package Report / ODBC Drivers Information

Item	Description
Driver	Name of an Open Database Connectivity (ODBC) driver in the package. Each database type has its own ODBC driver.
Description	Description of the ODBC driver identifying its associated database type.
File	File associated with the ODBC driver.
Component	Component associated with the ODBC driver.

ODBC DS View

The **ODBC DS** view lists all of the Open Database Connectivity (ODBC) data sources in the package. An ODBC data source identifies the source database type and provides information on how to connect to that database. For each ODBC DS, the following information is listed:

Table 22-39 • Package Report / ODBC DS Information

Item	Description
Data Source	Name of the ODBC data source, which identifies the source database type and provides information on how to connect to that database.
Description	Identifies the database type.
Driver Description	Name of this ODBC data source's associated ODBC driver.
Component	Component that this ODBC data source is affiliated with.

Extended Attributes View

The **Extended Attributes** view lists all of the extended attribute metadata that has been entered for this package. For each Extended Attribute, the following information is listed:

Table 22-40 • Package Report / Extended Attributes Information

Item	Description
Name	Name identifying the attribute.
Value	Content entered for the attribute.

Validation View

The **Validation** view lists all of the ICE rule errors and warnings that were generated when the package was validated against Microsoft ICEs (Internal Consistency Evaluators)—custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards.

For each error or warning, the following information is listed:

Table 22-41 • Package Report / Validation Information

Item	Description
ICE Rule	Name of ICE Rule that generated an error or warning message.
Description	Error or warning message.
Error Level	Indicates the severity of the message as either being a Warning or an Error. • **Errors**—Package authoring that will cause incorrect behavior. • **Warnings**—Package authoring that could possibly cause incorrect behavior. Warnings can also report unexpected side-effects of package authoring.

Conflicts View

The **Conflicts** view lists all of the unresolved errors that were found when conflict analysis was performed on this package. For each error, the following information is listed:

Table 22-42 • Package Report / Conflicts Information

Item	Description
ACE Rule	Name of ACE Rule that generated the message.
Description	Message generated during conflict analysis.
Target Package	Package that conflicted with this package.

History View

The **History** view lists all of the actions that have been performed on this package since it was imported into the Application Catalog. For each action, the following information is listed:

Table 22-43 • Package Report / History Information

Item	Description
Date	Day and time the event occurred.
Action	Identifies the event that occurred.
User	Identifies the user who executed the event.
Description	Description of the event that occurred.

Dependencies View

The **Dependencies** view lists all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog. For each dependency, the following information is listed:

Table 22-44 • Package Report / Dependencies Information

Item	Description
Name	Name of a file associated with this package that has dependencies with files used by other products or operating systems in the Application Catalog.
Path	Location where this dependent file is installed.
Size	Size of the dependent file.
Version	Version of the dependent file.

Properties View

The **Properties** view of the Package Report, which is only displayed for mobile apps, lists various attributes of the selected mobile application.

Reports Wizard

Using the Reports Wizard, you can generate Custom and Activity Reports and Custom SQL Query Reports. For more information, see the following topics:

- Creating a Custom Report
- Creating an Activity Report
- Generating a Custom SQL Query Report

You can use the Reports Wizard to generate reports of Workflow Manager deployment at any Workflow Consumer site. You can choose to include or exclude data, regardless of the specific consumer implementation. You can filter the data by companies, projects, requests, workflow Items, and other data, giving you maximum flexibility.

The Reports Wizard is comprised of the following panels:

- Select Report Objects Panel
- Select Report Fields Panel
- Define Report Filters Panel
- Select Template Data Panel
- Enter SQL Query Panel
- Specify General Information Panel
- Save and Preview Report Panel

Note • *Only Workflow Administrators with appropriate role permissions can create a report. Workflow Consumers cannot create reports.*

Select Stored Procedure Panel

On the **Select Stored Procedure** panel of the Reports Wizard, select the stored procedure that you want to use to generate a report and then click **Get Report**. The contents of this panel is determined by the selected stored procedure. You will be prompted to enter the information required by the stored procedure.

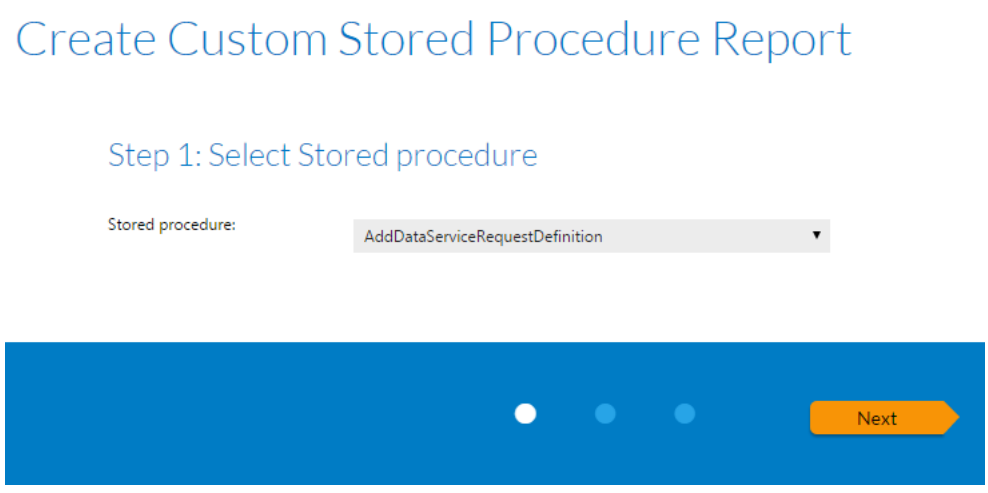


Figure 22-29: Select Stored Procedure Panel

To add a stored procedure to this list, open the `AMS_CustomReports` table and enter the name of the stored procedure you want to use to generate a report. For more information, see Generating a Custom Stored Procedure Report.

Note • *For more information on stored procedures, see SQL Stored Procedures in Microsoft TechNet.*

Select Report Objects Panel

On the **Select Report Objects** panel of the Reports Wizard, select the objects you want to include in the report:

- For a **Custom Report**, you can select **Applications**, **Companies**, **Projects**, and **Issues**.
- For an **Activity Report**, the only selection is **Activities**.

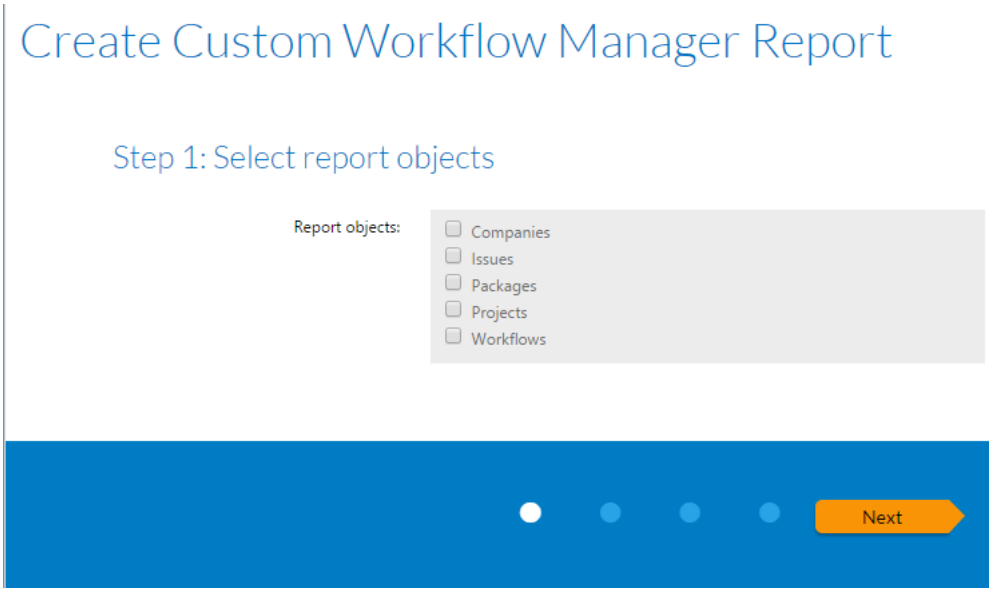


Figure 22-30: Select Report Objects Panel

Click **Next** to continue with the Reports Wizard.

Select Report Fields Panel

On the **Select Report Fields Panel** of the Reports Wizard, select the fields you want to include in the report. All of the available fields are listed by object.

- For a **Custom Report**, the **Applications**, **Companies**, **Projects**, and **Issues** objects could be listed.
- For an **Activity Report**, only the **Activities** object is listed.

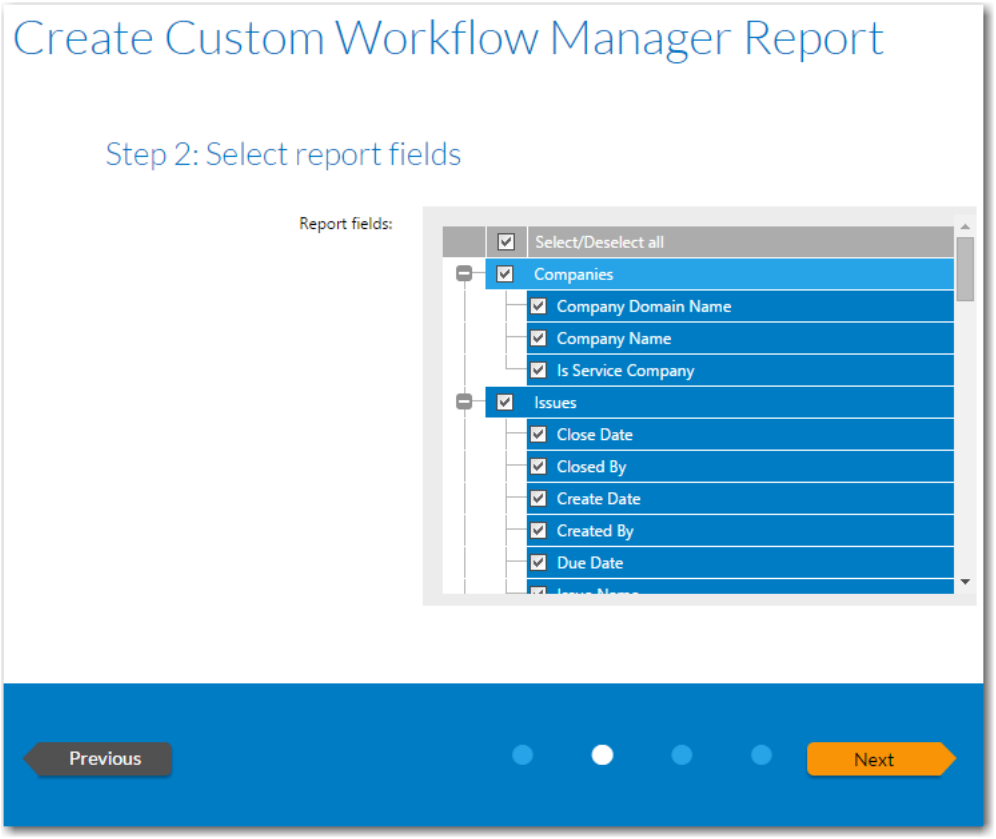


Figure 22-31: Select Report Fields Panel

Click **Next** to continue with the Reports Wizard.

Define Report Filters Panel

On the **Define Report Filters** panel of the Reports Wizard, enter a filter to define the data that you want to include in this report.

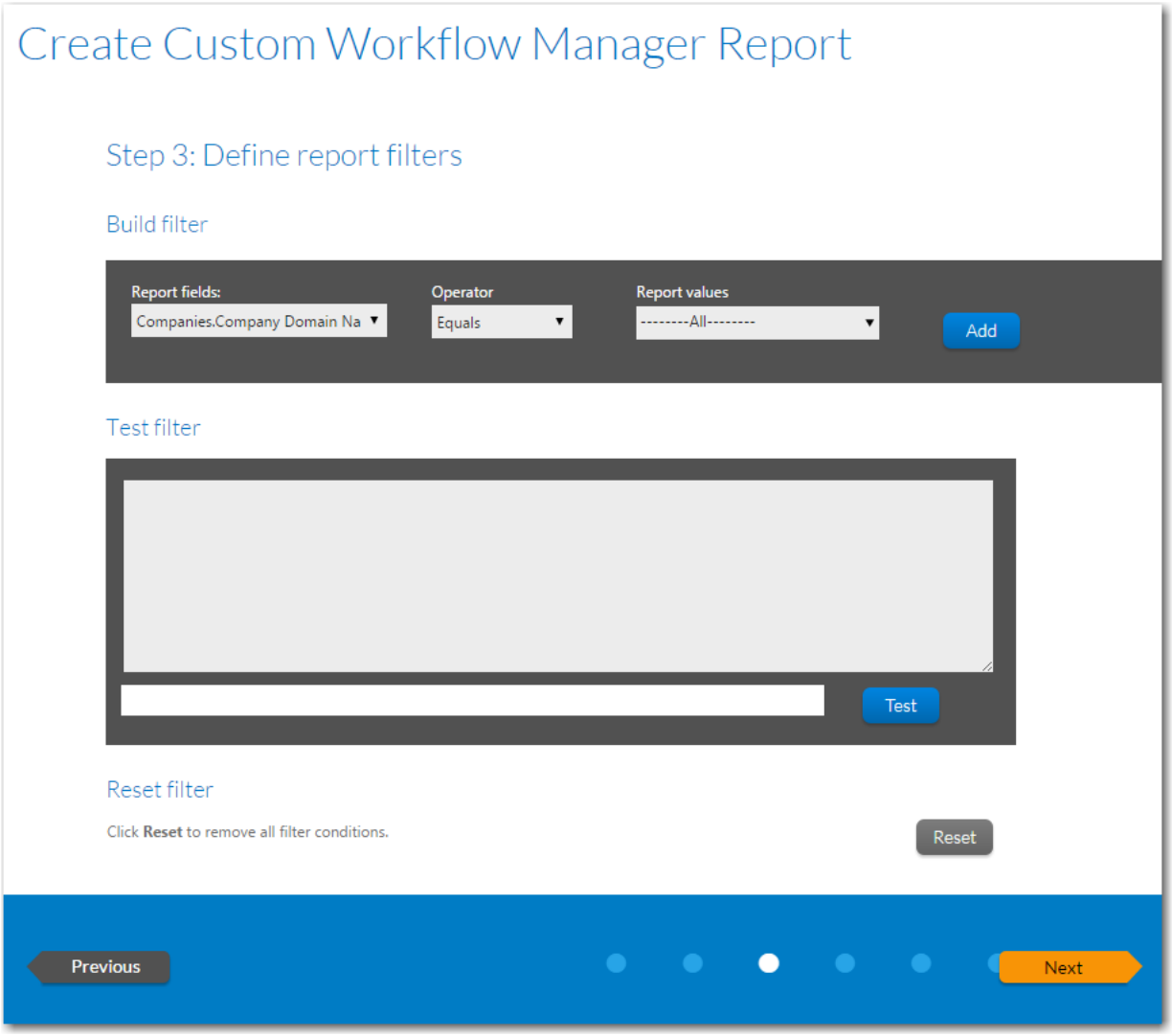


Figure 22-32: Define Report Filters Panel

The Define Report Filters panel includes the following properties:

Table 22-45 • Select Report Filters Panel of the Reports Wizard

Option	Description
Available Fields	Select a field from this tree to use to create a filter. When you click on a field to select it, all of its values populate the **Select a value for this filter** list.

Table 22-45 • Select Report Filters Panel of the Reports Wizard (cont.)

Option	Description
Set Filter Area	Use the following fields to create a filter to apply to this report: • **Select a value for this filter**—All of the values of the selected field are listed. Select the one that you want to use to create this filter. • **or alternatively type in a value for this filter**—If you want to use a value that does not appear in the list, type the value in this text box. • **Operator list box**—Select an operator from this list box to specify how you want the value in the selected field of each record to be selected, such as Equal, Greater Than, Less Than, etc. • **Add**—After you have set a filter, click Add to add the filter to the current filter conditions. It will be added to the Test Query box below, and a query is automatically run to determine if this filter generates any records. • **Conjunction Express List (AND, OR)**—After you have set one filter, and want to add another, select a conjunction from this box before you click Add to specify whether the record must meet both filter conditions (AND) or only one filter condition (OR). • **Test Query**—Click to run the specified query to determine if the filter combination generates any records. If no records are found, you are prompted to change the filters. • **Reset All**—Click to remove all filter conditions.

When you have finished defining filters, click **Next** to continue with the Reports Wizard.

Select Template Data Panel

On the **Select Template Data** panel of the Reports Wizard, you specify the Template data fields that you want to include in this report.

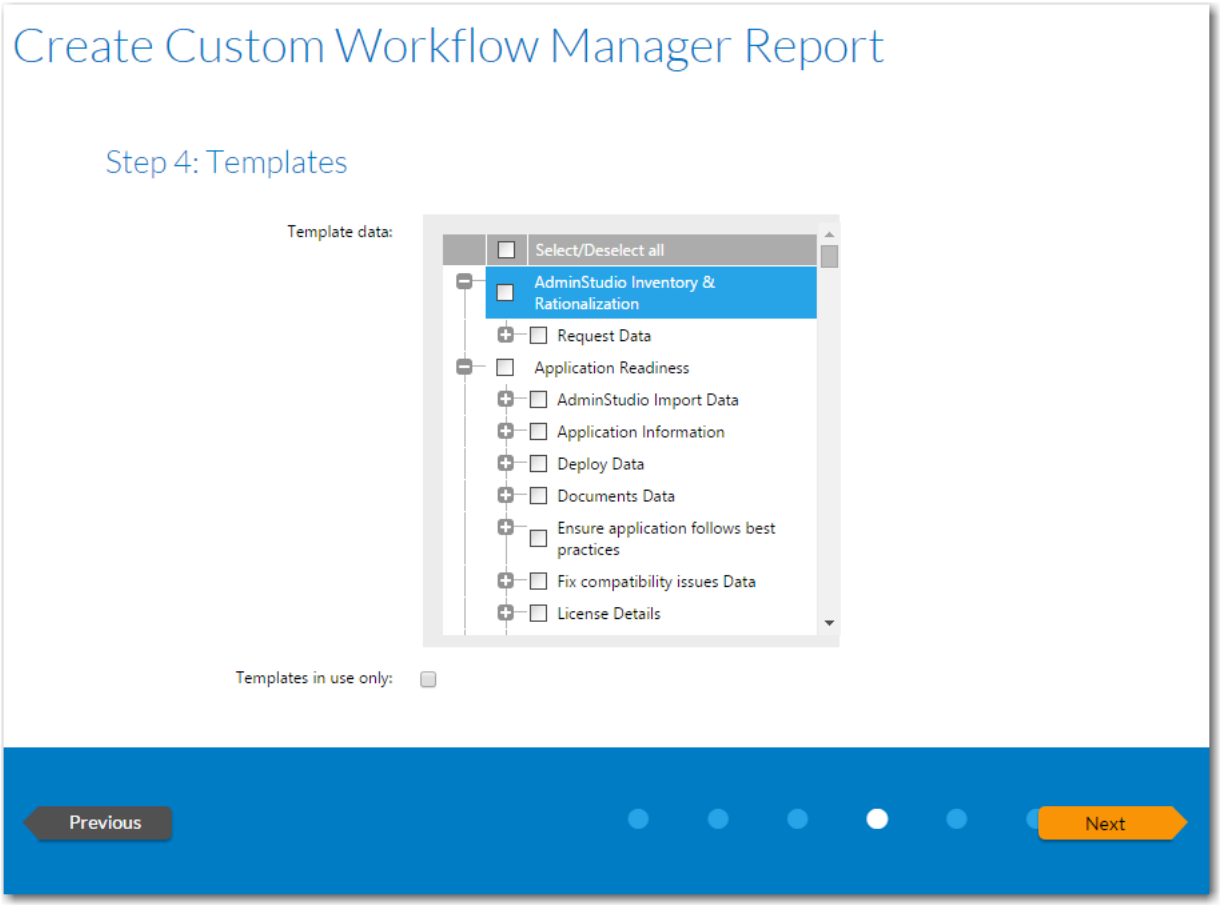


Figure 22-33: Templates Panel

In the **Available Templates** list, click the plus sign next to a Template name to expand the listing to show all data fields associated with that Template, and select the data fields that you would like to include in the report.

Select the **Templates in use only** option if you want only Templates that are associated with active projects and requests to be listed.

Click **Next** to continue with the Reports Wizard.

Enter SQL Query Panel

On the **Enter SQL Query** panel of the **Custom SQL Query Reports Wizard**, enter an SQL query to retrieve the data for this report.

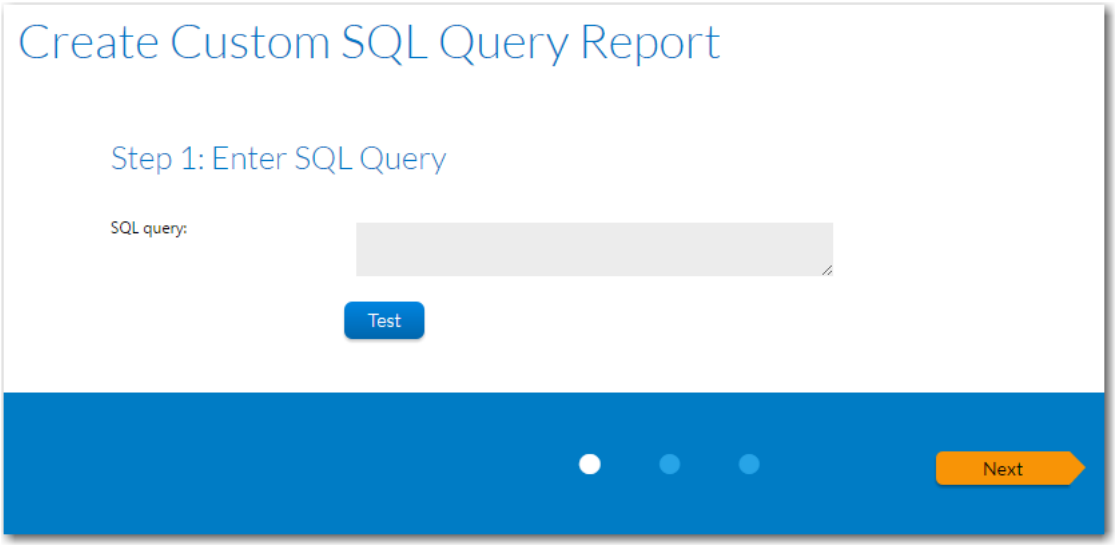


Figure 22-34: Enter SQL Query Panel

Click the **Test Query** button to verify the query syntax, and click **Next** to proceed.

Wildcard Support in Reports SQL Queries

In Reports searches, the LIKE operator is always used. You can combine the LIKE operator with a wildcard character, and the following rules apply:

Table 22-46 • Wildcard Support in Reports Queries

Situation	Rule

When no wildcards are used	If you do not enter a wildcard character in the **Search** box, then Reports performs a "LIKE" search, which searches for any occurrence of that text anywhere in the item that is being searched for.
	
	For example, if you are searching for a file name that has the word `test` anywhere in the file name, and you entered **`test`** in the **Search** box, it would be interpreted by Reports as:
	
	`*test*`
	
	And the following files would be found:
	
	`MyTestFile and TestFile`

When wildcards are used	You can specify a * wildcard in the **Search** box to narrow the search results. For example, if you are searching for a file name that includes the word `test`, but does not begin with it, and you entered `*test` in the **Search** box, `MyTest` would be returned, but not `TestFile`.

Specify General Information Panel

On the **Specify General Information** panel of the Reports Wizard, enter a **Report Name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.

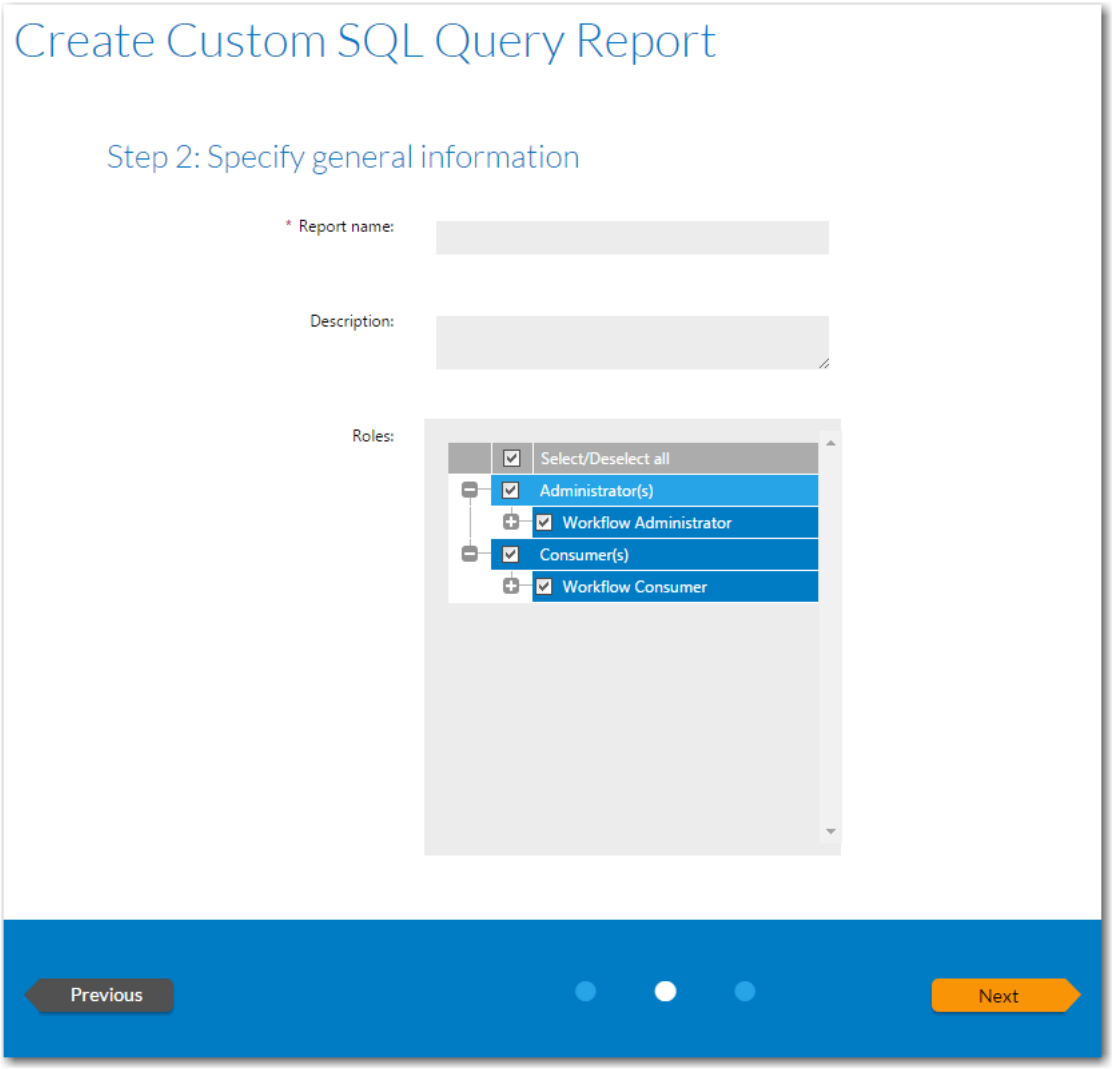


Figure 22-35: Specify General Information Panel

Next, select the **Roles** that you want to have permission to view this report.

Click **Next** to continue with the Reports Wizard.

Note • *You can change the selected roles at any time after this report is created by clicking Edit next to the Report Name on the **All Reports** page.*

Save and Preview Report Panel

The **Save and Preview Report** panel lists a summary of the selections you have made while creating the report.

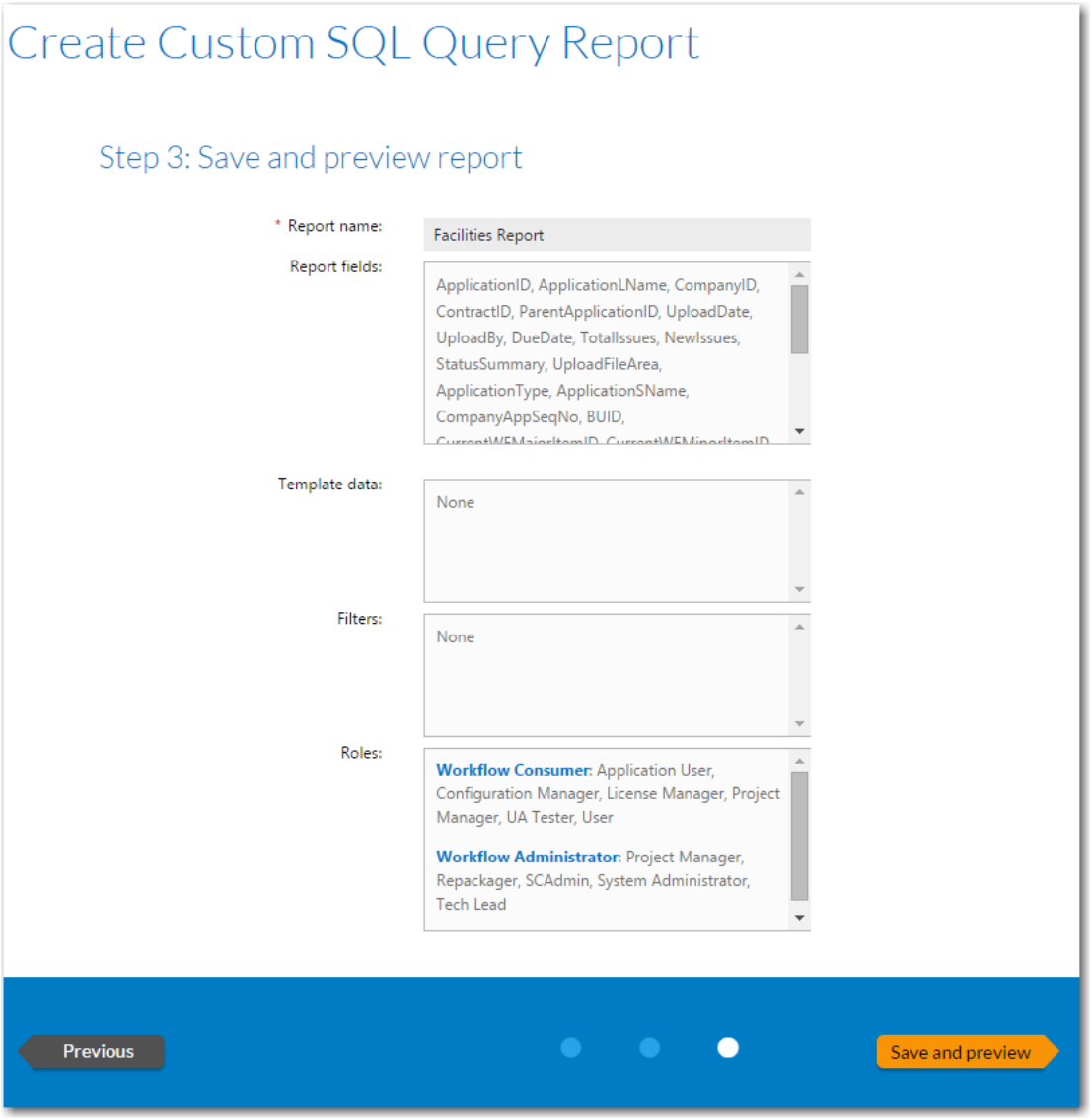


Figure 22-36: Save and Preview Report Panel

On the **Save and Preview Report** panel of the Reports Wizard, the following information is listed:

Table 22-47 • Save and Preview Report Panel

Option	Description
Report Name	Name of report.
Report Fields	List of fields that you selected to be included in this report.
Template Data	A list of the Template data you selected to be in this report.
Filters	A list of filters applied to this report.

Click **Save and preview** to generate the Report. The report is generated. This report is also saved and now appears in the list on the **All Reports** page.

23

AdminStudio PowerShell Cmdlets

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the AdminStudio PowerShell Cmdlets to integrate your existing .NET applications or scripting environments like Microsoft PowerShell with AdminStudio.

The AdminStudio Platform exposes the core tasks involved in the application readiness process lifecycle. Automating these core tasks via PowerShell scripts or .NET applications helps your enterprise achieve a higher throughput during this process.

This chapter includes the following topics:

- About the AdminStudio PowerShell Cmdlets
- Setting Up AdminStudio Module in PowerShell
- Example Script to Create Application Catalog, Import Packages, and Perform Testing
- PowerShell Cmdlets Reference

About the AdminStudio PowerShell Cmdlets

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Some of the core tasks that the AdminStudio PowerShell Cmdlets enables you to automate include:

- **Application Catalogs**—Creating a new Application Catalog or upgrading an existing Application Catalog.
- **Importing**—Importing existing packages into the AdminStudio Application Catalog.
  - Supports importing all the formats that are currently supported by Application Catalog.

  - Supports applying transforms and patches to Windows Installer packages during import.
  - Supports importing individual packages or a directory of packages.
  - Supports importing applications from System Center 2012 Configuration Manager.
- **Application virtualization compatibility**—Checking to see if your packages for suitable for conversion to virtual formats.
- **Application model properties**—Can set the application model properties of an application.
- **Virtualization**—Converting your packages to Microsoft App-V, Citrix XenApp, VMware ThinApp.
  - Allows you to do the conversion one package at a time, or a bulk folder conversion.
- **Testing**—Testing your packages by running operating system compatibility, best practice validation, and conflict analysis tests on the packages, and viewing test results.
- **Publishing**—Publishing your applications to Microsoft System Center 2012 Configuration Manager.

PowerShell support follows the standard Microsoft PowerShell conventions, including well-documented cmdlets within PowerShell.

Setting Up AdminStudio Module in PowerShell

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

AdminStudio Platform functionality is shipped in the form of a PowerShell Module that can be used in PowerShell. The AdminStudio Module can be installed on a machine or can be used temporarily per PowerShell session.

AdminStudio Microsoft .NET 4.0 Requirement

The AdminStudio DLLs are built using .NET 4.0, while PowerShell by default runs in .NET 2.0. To enable PowerShell to load the AdminStudio DLLs, you need to perform the following steps:

Task ***To enable PowerShell to load AdminStudio DLLs:***

1. Create a file named `PowerShell.exe.config` containing the following content:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <!-- http://msdn.microsoft.com/en-us/library/w4atty68.aspx -->
  <startup useLegacyV2RuntimeActivationPolicy="true">
    <supportedRuntime version="v4.0.30319"/>
    <supportedRuntime version="v2.0.50727"/>
  </startup>
</configuration>
```

2. Copy this file to the `C:\Windows\SysWOW64\WindowsPowerShell\v1.0` directory.

Enabling a Module Per PowerShell Session

To enable a ModuleModule per PowerShell session, use the following cmdlet:

```
Import-Module -Name [AdminStudioInstallDir]\Common\AdminStudio.Platform.PowerShellExtensions.dll
```

The following import is needed in any of the PowerShell sessions/scripts:

```
Import-Module -Name [AdminStudioInstallDir]\Common\AdminStudio.Utilities.dll
```

Installing the Module on a Machine

To install the AdminStudio Module on a given machine, use the following cmdlet at a PowerShell prompt:

```
Set-Alias installutil $env:windir\Microsoft.NET\Framework\v4.0.30319\installutil

installutil [AdminStudioInstallDir]\Common\AdminStudio.Platform.PowerShellExtensions.dll
```

The following import is needed in any of the PowerShell sessions/scripts:

```
Import-Module -Name [AdminStudioInstallDir]\Common\AdminStudio.Utilities.dll
```

Running the Invoke-ASPublish Cmdlet

To run the `Invoke-ASPublish` cmdlet, you need the following import, either in your PowerShell script or the current PowerShell session:

```
Import-Module -Name [AdminStudioInstallDir]\Common\AdminStudio.SCCM.Integrator.dll
```

Requirements

The prerequisite includes the following:

- For the AdminStudio Platform to work properly, PowerShell needs to run under administrator privileges and also needs to be launched with the `-STA` flag.
- AdminStudio's PowerShell Cmdlets will be available for use only on the machines where AdminStudio is installed and licensed with Enterprise Edition.

Example Script to Create Application Catalog, Import Packages, and Perform Testing

***Edition** • Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the example PowerShell script that is provided in this section to perform the following tasks:

- **Create Application Catalog**—Create a new Application Catalog database.
- **Import packages**—Import all of the `.msi` files from a specified directory into the Application Catalog.
- **Perform testing**—Run all selected tests, and report the summary of errors and warnings.

Example Script

The following script uses the `Set-ASConfigPlatform`, `New-ASCatalogPlatform`, `Invoke-ASImportPackage`, and `Test-ASPackage` AdminStudio PowerShell cmdlets to create a new Application Catalog, import packages, and then test those packages and view the test results. The AdminStudio PowerShell cmdlets and parameters are highlighted:

```
#################################################################
# Read Command Line Parameters
#################################################################
param ($CatalogName = $(Read-Host "Enter New Catalog Name")

#################################################################
# Input required from users
#################################################################
$folder             = "C:\code\Demo"
$ConnectionString   = 'PROVIDER=MSOLEDBSQL19;Data Source=localhost;Initial Catalog=MyNewCat;Integrated
Security=SSPI;'

#################################################################
# Non-User Settings
#################################################################
$shive                = "HKLM:\SOFTWARE\Wow6432Node\InstallShield\AdminStudio\15.0\"
$slocation            = "Product Location"
$sAsLoc               = (Get-ItemProperty $shive $slocation).$slocation
$sCurrentLoc          = [Environment]::CurrentDirectory=(Get-Location-PSProvider
FileSystem).ProviderPath
$sAsLoc               = $sAsLoc + "Common\"
$global:oPkgArray     = @()
$global:oPkgArrayFail = @()

#################################################################
# Functions
#################################################################
function Import ($s)
{
    $f = [System.IO.File]::GetAttributes($s)
    $d = ($f-band [System.IO.FileAttributes]::Directory)
    if (!$d)
    {
        Write-Host 'Importing:' $s
        $obj = Invoke-ASImportPackage-PackagePath $s;
    if ($obj.GetType().FullName-eq 'AdminStudio.Platform.Helpers.PackageHelper')
    {
        #Write-Host 'Success' $s
        $global:oPkgArray = $global:oPkgArray + $obj
    }
    else
    {
        Write-Host 'Failured to import:' $s -foregroundcolor red
        $global:oPkgArrayFail = $global:oPkgArrayFail + $obj
    }
    }
}

function LoadDLL ($s)
{
```

```
    $FileName = $sAsLoc + $s
    import-module-name $FileName
}

function PrepAS ()
{
    cd $sAsLoc
    LoadDLL 'AdminStudio.Platform.PowerShellExtensions.dll'
    LoadDLL 'AdminStudio.Utilities.dll'
    LoadDLL 'AdminStudio.SCCM.Model.dll'
    Set-ASConfigPlatform-ConnectionString    $ConnectionString
}

function Write-Host-Indent ()
{
    Write-Host '                '-nonewline
}

function Write-Host-Drawline ()
{
    Write-Host '************************************'-foregroundcolor yellow
}

function Test ($o)
{
    Write-Host 'Testing Package:' $o.DisplayedProductName-nonewline
    Write-Host ' RowId:' $o.RowID-foregroundcolor gray
    $oTestResults = Test-ASPackage-PackageId $o.RowID
    $errors = 0;
    $warn = 0;
    foreach ($oTestResult in $oTestResults.Stats)
    {
        $errors = $errors + $oTestResult.Errors
    $warn =$warn + $oTestResult.Warnings
    }
    Write-Host-Indent
    Write-Host 'Errors:' $errors
    Write-Host-Indent
    Write-Host 'Warnings:' $warn
}

################################################################
# Main Loop
################################################################
$tBegin = Get-Date
Write-Host 'Begin:' $tBegin-foregroundcolor gray
Write-Host-Drawline
Write-Host '      Import from Folder and Test'
Write-Host-Drawline
Write-Host 'Directory =' $folder-foregroundcolor gray
Write-Host 'ConnectionString =' $ConnectionString-foregroundcolor gray
Write-Host 'AdminStudio Directory =' $sAsLoc-foregroundcolor gray
Write-Host-Drawline

################################################################
# Load Required DLLs
```

```
#################################################################
PrepAS
Write-Host

#################################################################
# Create Catalog
#################################################################
Write-Host 'Creating New Catalog' $CatalogName-foregroundcolor yellow
New-ASCatalog-CatalogName $CatalogName
Write-Host
Write-Host 'Importing Applications from' $folder-foregroundcolor yellow

# Iterate Toplevel Folder Only for Importing
foreach ($file in Get-Childitem-include '*.msi' -Recurse $folder)
{
    Import ($file)
}

Write-Host 'Packages that Import Succeeded:' $global:oPkgArray.Count
Write-Host 'Packages that Import Failed:' $global:oPkgArrayFail.Count

$tEnd = Get-Date
$tDiff = $tEnd - $tBegin
Write-Host 'End:' $tEnd-foregroundcolor gray
Write-Host 'Total Time:' $tDiff.Hours' hours' $tDiff.Minutes' minutes ' $tDiff.Seconds 'seconds'

#################################################################
#Run tests
#################################################################
foreach ($oPkg in $global:oPkgArray)
{
    Test ($oPkg);
}

#################################################################
# Write out end time
cd $sCurrentLoc
$tEnd = Get-Date
$tDiff = $tEnd - $tBegin
Write-Host 'End:' $tEnd-foregroundcolor gray
Write-Host 'Total Time:' $tDiff.Hours' hours' $tDiff.Minutes' minutes ' $tDiff.Seconds 'seconds'
```

Output

When you run the script in Example Script, you will see output similar to the following:

```
PS C:\code\Script> .\MyScript.ps1
Enter New Catalog Name: MyScript
Begin: 2/18/2016 11:27:57 AM
************************************
    Import from Folder and Test
************************************
Directory = C:\code\Demo
ConnectionString = PROVIDER=MSOLEDBSQL19;Data Source=localhost;Initial Catalog=MyNewCat;Integrated
Security=SSPI;
AdminStudio Directory = C:\Program Files (x86)\AdminStudio\2016\Common\
```

```
************************************

Creating New Catalog MyScript

Importing Applications from C:\code\Demo
Importing: C:\code\Demo\Firefox_MSI\Firefox.msi
Packages that Import Succeeded: 1
Packages that Import Failed: 0
End: 2/18/2016 11:28:00 AM
Total Time: 0  hours 0  minutes 3 seconds
Testing Package: Mozilla_Firefox RowId: 2
                 Errors: 0
                 Warnings: 382
End: 3/18/2016 11:28:35 AM
Total Time: 0  hours 0  minutes 37 seconds
```

PowerShell Cmdlets Reference

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Following AdminStudio PowerShell Cmdlets are available.

Table 23-1 • AdminStudio PowerShell Cmdlets

Cmdlet	Description
Add-ASKeywords	Adds App Portal keywords to the Application Catalog.
Add-ASPackageForConversion	Adds a package to the Automated Application Converter project file for conversion.
Get-ASAppVShortcuts	Returns the available shortcuts and its properties.
Get-ASApplicationDetails	Returns details of applications in the Application Catalog.
Get-ASApplicationID	Returns the `ApplicationID` for a given `PackageID`.
Get-ASAppPortalCategories	Returns an XML stream of existing categories in App Portal.
Get-ASAppPortalTemplates	Returns an XML stream of existing templates in App Portal.
Get-ASCatalogItem	Returns a list of the root items of the specified type: `Group`, `Application`, or `Package`.
Get-ASConfigPlatform	Retrieves configuration information, such as retrieving the database connection string to which the current PowerShell session is configured.

Table 23-1 • AdminStudio PowerShell Cmdlets

Cmdlet	Description
Get-ASApplicationDeploymentSummary	Returns the deployment history of a given distribution system.
Get-ASDeploymentSystemPackageTree	Use to query System Center 2012 Configuration Manager for a list of application IDs for all of its applications, which can then used to import applications into the Application Catalog using the Invoke-ASImportAppFromDeploymentSystem cmdlet.
Get-ASDistributionProperty	Use to retrieve the property value of the distribution system for a package.
Get-ASKeywords	Returns a list of App Portal keywords in the Application Catalog, in a comma-delimited list.
Get-ASOptionProperty	Returns a property value for a given property name.
Get-ASPackage	Returns a package object, given the `PackageID`.
Get-ASPackageTestSummary	Returns a summary of various tests performed for the package that is specified using the `-PackageID` parameter.
Get-ASProperty	Returns the value for a property specified using the `-PropertyName` parameter associated to a specified package specified using `-PackageId` parameter.
Get-ASPackageFeedDetails	Returns the product details like FileName, SilentCommandLineSwitches, etc. for the specified PackageFeedId
Get-ASTestDetails	Displays the details of an application compatibility or Microsoft ICE test that is run using the `Test-ASPackage` cmdlet.
Get-ASTestState	Use to return the test state (selected or not selected) of a given test.
Get-ASVirtualReadiness	Gets the virtual readiness of a given package.
Invoke-ASAppVBulkUpgrade	Used for bulk conversion of App-V 4.x packages (`.sft`) to App-V 5.x packages (`.appv`).
Invoke-ASConvertFolder	Converts a folder of packages to specified virtual formats using Automatic Application Converter.
Invoke-ASConvertPackage	Use to convert a package from one package type to another using an XML input file.

Table 23-1 • AdminStudio PowerShell Cmdlets

Cmdlet	Description
Invoke-ASConvertPackageEx	Use to invoke the Application Catalog Conversion Wizard process to convert a package from one package type to another.
Invoke-ASImportAppFromDeploymentSystem	Use to import an application from System Center 2012 Configuration Manager into the Application Catalog, using the ID returned from the Get-ASDeploymentSystemPackageTree cmdlet.
Invoke-ASImportPackage	Invokes an import process on a single package.
Invoke-ASWrapPackage	Use to convert a Windows Installer (`.msi`) or installation package (`.exe`) to a ASWrapPackage file (`.ps1`).
Invoke-ASPackageCustomize	Used to generate the customize transform file (.mst) for the existing msi package which is download/imported from the Package Feed Module.
Invoke-ASPackageFeedSearch	Invokes an application details like PackageFeedId, Types, etc. from the Package Feed Module
Invoke-ASPackageFeedDownload	Use to download an application from the Package Feed Module.
Invoke-ASPackageFeedSync	Invokes a package feed sync.
Invoke-ASPublish	Publishes a package to a specified distribution system.
Invoke-ASUpdatePackage	Invokes a scan process on a single package.
Invoke-ASUpdatePackage	Run on an existing `.exe` package to extract the bundled `.msi` details.
Invoke-ASSignPackage	Invokes a signing process for the MSIX package.
New-ASCatalog	Use to create a new Application Catalog.
New-ASDistributionConnection	Use to define named connections to ConfigMgr (Formerly called as System Center Configuration Manager) and Citrix XenApp distribution systems.
New-ASDistributionConnectionEx	Use to create distribution connections using an XML file to provde input parameters.
New-ASCreateSoftwareTag	Use to create a new Software Tag.
Remove-ASApplication	Use to delete a package using its OID.

Table 23-1 • AdminStudio PowerShell Cmdlets

Cmdlet	Description
Remove-ASGroup	Use to delete a group using its Row ID.
Remove-ASPackage	Use to delete a package using its Row ID.
Resolve-ASPackage	Use to run application compatibility fixes on a package.
Set-ASAppVShortcutProperty	Use to select or unselect the shorcut properties to be installed.
Set-ASCatalog	Use to set the default Application Catalog.
Set-ASConfigPlatform	Sets defaults for most of the parameters.
Set-ASDistributionProperty	Use to set the property value of the distribution system for a package.
Set-ASOptionProperty	Use to set the property value of the download and import using package feed module.
Set-ASProperty	Use to set the application model properties of a package.
Set-ASSoftwareRepositoryState	Use to perform **CheckOut** and **UndoCheckOut** operations on a Software Repository-enabled Application Catalog.
Set-ASTestState	Use to set a given test to either run or not run.
Set-ASSoftwareTagProperties	Use to set the software tag properties.
Start-ASConversion	Starts automated conversion using a given .AACX file.
Test-ASConflicts	Runs conflict analysis on the specified package.
Test-ASPackage	Validates the package for best practices.
New-ASPackageRequest	Use to add package request in the Backlog.

Add-ASKeywords

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Add-ASKeywords` cmdlet to add App Portal catalog item keywords to the Application Catalog (as individual records in the `ASKeywords` table). After creating a keyword, you can use the `Set-ASProperty` cmdlet to assign the keyword to an application, as described in the Set-ASProperty topic under Keywords.

***Note** • Keywords created using the Add-ASKeywords cmdlet are also available for selection on the **Keywords** dialog box. For more information, see Specifying Catalog Item Keywords.*

***Note** • Until you create a keyword using either the Add-ASKeywords cmdlet or the **Edit Keywords** dialog box, you cannot use the Set-ASProperty cmdlet to assign it to an application. If you attempt to do so, an error will be returned.*

Example

The following is an example of the Add-ASKeywords cmdlet:

```
Add-ASKeywords –NewKeywords 'accounting, spreadsheet, project management, graphs'
```

Parameters

The Add-ASKeywords cmdlet has the following parameter:

Table 23-2 • Add-ASKeywords Parameters

Parameter	Description
NewKeywords	Use to add App Portal catalog item keywords to the Application Catalog. If you are adding more than one keyword, you must enclose the comma-delimited list in single quote marks, such as: `Add-ASKeywords–NewKeywords 'accounting, spreadsheet, graphs'`

Return Values

One of the following values is returned:

- **True**—Keyword was added to the ASKeywords table.
- **False**—Keyword was not added to the ASKeywords table.

Add-ASPackageForConversion

***Edition** • Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The Add-ASPackageForConversion cmdlet adds a package to the Automated Application Converter project file for conversion. It could be used to add a series of packages to an Automated Application Converter project file for conversion. This cmdlet allows you to make decisions in your script for a list of packages and choose which one you need to add for conversion.

Note • *To make sure that the Automated Application Converter project file is cleaned of any packages in it, use the* `-CleanProjectFile` *parameter. Usually this is used the first time you add a package.*

Examples

The following are examples of the `Add-ASPackageForConversion` cmdlet:

```
Add-ASPackageForConversion-PackagePath C:\Packages\Reader\Reader.msi-CleanProjectFile
```

```
Add-ASPackageForConversion-PackagePath C:\Packages\Orca\Orca.msi-AACSettings C:\Packages\test.aacx
```

In this example, two packages are added to the default Automated Application Converter project file (which is specified in the Automated Application Converter settings file using the `Set-ASConfigPlatform -AACSettings` cmdlet).

This example includes the `-CleanProjectFile` parameter in the first cmdlet to clean up the list of packages in the project file when it is created. However, it is not necessary to specify the parameter in subsequent `Add-ASPackageForConversion` cmdlets.

Parameters

The `Add-ASPackageForConversion` cmdlet has the following parameters:

Table 23-3 • Add-ASPackageForConversion Parameters

Parameter	Description
PackagePath	Mandatory parameter which specifies the path to the package that needs to be added to the Automated Application Converter settings file for conversion.
[AACSettings]	Use to specify the Automated Application Converter project file to use during conversion. If it is not supplied, a copy of the project file specified in the platform settings file will be used.
[CleanProjectFile]	Specify this parameter the first time you add a package so that references to any packages in the Automated Application Converter project file will be removed.
[CommandLine]	Use to specify command line parameters that can be used to silently install this package during repackaging.
[HardTimeOut]	Use to specify the hard time-out (in minutes) for the package installation.
[SoftTimeout]	Use to specify the soft time-out (in minutes) for the package installation.
[Transforms]	List of transforms to use during repackaging. When specifying multiple transform files, use commas to separate them.
[UseSingleStepSnapshot]	Use to specify that you want to use the Snapshot installation technology to repackage the package. If this parameter is not used, the Installation Monitoring installation technology will be used.

Table 23-3 • Add-ASPackageForConversion Parameters (cont.)

Parameter	Description
[IsCompressed]	Specifies that this package (mostly `.exe` files) is compressed, so the entire folder tree including this `.exe` will be copied for repackaging. For `.exe` files, the default value is `Compressed=False`.

Return Values

The name and path to the Automated Application Converter settings file is returned in the following format:

`[Path]/ProjectFileName``.Copy.aacx`

Get-ASAppVShortcuts

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

The `Get-ASAppVShortcuts` cmdlet returns the available Shortcuts and its properties for a given PackageID.

Example

The following is an example of the `Get-ASAppVShortcuts` cmdlet:

```
Get-ASAppVShortcuts -PackageID 1
```

Parameters

The `Get-ASAppVShortcuts` cmdlet has the following parameters:

Table 23-4 • Get-ASAppVShortcuts Parameters

Parameter	Description
PackageID	Specify the `PackageID` of the package that you need the shortcuts and it properties.

Return Values

Available Shortcuts and their properties for the given package is returned.

Set-ASAppVShortcutProperty

Get-ASApplicationDetails

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Get-ASApplicationDetails` cmdlet returns details of applications in the Application Catalog.

Example

The following is an example of the `Get-ASApplicationDetails` cmdlet:

```
Get-ASApplicationDetails -ProductName -Manufacturer -Version
Get-ASApplicationDetails -Manufacturer “Google Inc.”
```

Parameters

If switches are not provided, it should list all packages in the catalog in the following format:

Table 23-5 • Parameters

Parameter	Mandatory	Description
`ProductName`	False	ProductName of the Package in the Application Catalog.
`Manufacturer`	False	Manufacturer of the Package in the Application Catalog.
`Version`	False	Version of the Package in the Application Catalog.

Return Values

- If parameters are provided, it returns a list of products present in the Application Catalog matching the parameter values provided a the API.
- If no parameters are provided, it returns a list of all products present in the Application Catalog in the following format:

```
PackageId:
ProductName:
Manufacturer:
Version:
PackageType:
ApplicationId:
PackageCatalogPath:
```

Get-ASApplicationID

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The Get-ASApplicationID cmdlet returns the ApplicationID for a given PackageID. This is useful when you have a PackageID from Invoke-ASImportPackage and need to publish the application to ConfigMgr (Formerly called as System Center Configuration Manager) using the Invoke-ASPublish cmdlet. The Invoke-ASPublish cmdlet requires a mandatory ApplicationID parameter.

Example

The following is an example of the Get-ASApplicationID cmdlet:

```
Get-ASApplicationID -PackageID 10
```

Parameters

The Get-ASApplicationID cmdlet has the following parameters:

Table 23-6 • Get-ASApplicationID Parameters

Parameter	Description
PackageID	Specify the PackageID of the package that you need the ApplicationID for.

Return Values

The ApplicationID for the package is returned.

Get-ASAppPortalCategories

***Edition** • Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The Get-ASAppPortalCategories cmdlet returns an XML stream of existing categories in App Portal, such as:

```
<Categories Status="Synced">
  <Category Id="1" XPath="Software">
    <Name>Software</Name>
  <Category Id="3" XPath="Software/Microsoft">
      <Name>Microsoft</Name>
    </Category>
  <Category Id="2" XPath="Hardware">
    <Name>Hardware</Name>
  </Category>
  <Category Id="13" XPath="Data">
    <Name>Data</Name>
  </Category>
  <Category Id="14" XPath="LM Desktop QA">
    <Name>LM Desktop QA</Name>
  </Category>
</Categories>
```

You can then use the Set-ASProperty cmdlet to set the App Portal category for an application, such as:

```
Set-ASProperty -PackageID 1-PropertyName "Categories"-PropertyValue "Software/Microsoft"
```

When the application is published to an App Portal-linked distribution system, an App Portal catalog item will be created and will appear in the specified category.

Example

The following is an example of the `Get-ASAppPortalCategories` cmdlet:

```
Get-ASAppPortalCategories
```

Parameters

The `Get-ASAppPortalCategories` cmdlet has no parameters.

Note • *For more information on setting App Portal properties using PowerShell cmdlets, see App Portal Information Tab under Set-ASProperty. App Portal properties that can be set include: Categories, Template, Keywords, Long Description, and Brief Description.*

Get-ASAppPortalTemplates

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

In App Portal, you can use templates to automatically assign a defined set of properties to a catalog item. The `Get-ASAppPortalTemplates` cmdlet returns an XML stream of existing templates in App Portal, such as:

```
<Templates Status="Synced">
  <Template Id="172">
    <Name>MyTemplateTwo</Name>
  </Template>
</Templates>
```

You can then use the `Set-ASProperty` cmdlet to set the App Portal template for an application, such as:

```
Set-ASProperty -PackageID 1-PropertyName "Templates"-PropertyValue "Standard SCCM 2012"
```

When the application is published to an App Portal-linked distribution system, an App Portal catalog item will be created using the specified template.

Example

The following is an example of the `Get-ASAppPortalTemplates` cmdlet:

```
Get-ASAppPortalTemplates
```

Parameters

The `Get-ASAppPortalTemplates` cmdlet has no parameters.

***Note** • For more information on setting App Portal properties using PowerShell cmdlets, see App Portal Information Tab under Set-ASProperty. App Portal properties that can be set include: Categories, Template, Keywords, Long Description, and Brief Description.*

Get-ASCatalogItem

***Edition** • Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Get-ASCatalogItem` cmdlet returns a list of the root items of the specified type: `Group`, `Application`, or `Package`. For example, if you use this cmdlet with the `ItemType` of `Group`, the applications in the specified group will be listed, along with each application's `RowID`. If you use an `ItemType` of `Application`, that application's packages will be listed, along with each package's `RowID`.

You can use this cmdlet to display the groups, applications, and packages in your Application Catalog from a source outside of Application Catalog. Also, once the `RowID` of an item in the Application Catalog is known, you can use other AdminStudio PowerShell Cmdlets cmdlets to perform actions on that item.

Example

The following is an example of the `Get-ASCatalogItem` cmdlet:

```
Get-ASCatalogItem -ItemType 'Group'-ItemId 1
```

Parameters

The `Get-ASCatalogItem` cmdlet has the following parameters:

Table 23-7 • Get-ASCatalogItem Parameters

Parameter	Description
-ItemType	Specify one of the following types to identify which type of catalog item you want to list the contents of, in single quote marks: • `Group` • `Application` • `Package` For example: `Get-ASCatalogItem-ItemType 'Group'-ItemId 10`
-ItemId	Use to specify the ID number of the group, application or package that you want to list the contents of. For example: `Get-ASCatalogItem-ItemType 'Group'-ItemId 10`

Sample Script

Below is a sample script that uses the `Get-ASCatalogItem` cmdlet. If you use option 11 in this script, the entire package tree will be returned.

```
################################################################
# Input required from users
################################################################
$DefaultExt         =  @('*.msi','*.sft')
$TestsToEnable      =  @('0001','0002','0003','0004','0005','0007','0008','0009','0012','0014',
'0021','0023','0029','0030','0035','0038','0039','0044')
$TestsToDisable     =  @('0501','0502','0503','0504','0505','0506','0507','0508','0509', '0510',
'0511','0512','0513','0514','0515','0516','0517','0519','0520','0521','0522','0523','0524','0525','0526
','0527','0528','0529','0530','0531','0533','0534','0535','0537','0538','0539','0540','0541','0542','05
43','0544','0545','0546','0547','0548','0549','0550','0551','0552','0553','0401','0402','0403','0404','
0405','0406','0407','0408','0409','0410','0411','0412','0413','0414','0415','0416','0417','0419','0420'
,'0421','0422','0423','0424','0425','0426','0427','0428','0429','0430','0435','0437','0438','0440','044
1','0443','0444','0445','0446','0447','0448','0449','0450','0439','0442','0451','0452','0453','0454','0
301','0302','0303','0304','0305','0306','0307','0308','0309','0310','0311','0312','0313','0314','0315',
'0316','0318','0319','0320','0321','0322','0323','0324','0325','0326','0327','0328','0329','0330','0335
','0338','0340','0341','0343','0344','0345','0346','0347','0348','0349','0350','0339','0342','0352','03
53','0354','0201','0202','0203','0204','0205','0206','0207','0208','0209','0210','0211','0212','0213','
0214','0215','0216','0217','0219','0220','0221','0222','0223','0224','0225','0226','0227','0228','0229'
,'0230','0235','0237','0238','0244','0245','0246','0247','0248','0249','0250','0239','0251','0252','025
3','0101','0102','0103','0104','0105','0106','0107','0108','0109','0110','0111','0112','0113','0114','0
115','0116','0117','0119','0120','0121','0122','0123','0124','0125','0126','0127','0128','0129','0130',
'0131','0133','0134','0135','0137','0138','0139','0144','0145','0146','0147','0148','0149','0150','0151
','0152','0153','0001','0002','0003','0004','0005','0006','0007','0008','0009','0010','0011','0012','00
13','0014','0015','0016','0018','0019','0020','0021','0022','0023','0024','0025','0026','0027','0028','
0029','0030','0035','0038','0039','0044','0045','0046','0047','0048','0049','0050','0052','0053','ICE01
','ICE02','ICE03','ICE04','ICE05','ICE06','ICE07','ICE08','ICE09','ICE10','ICE12','ICE13','ICE14','ICE1
5','ICE16','ICE17','ICE18','ICE19','ICE20','ICE21','ICE22','ICE23','ICE24','ICE25','ICE26','ICE27','ICE
28','ICE29','ICE30','ICE31','ICE32','ICE33','ICE34','ICE35','ICE36','ICE38','ICE39','ICE40','ICE41','IC
E42','ICE43','ICE44','ICE45','ICE46','ICE47','ICE48','ICE49','ICE50','ICE51','ICE52','ICE53','ICE54','I
CE55','ICE56','ICE57','ICE58','ICE59','ICE60','ICE61','ICE62','ICE63','ICE64','ICE65','ICE66','ICE67','
ICE68','ICE69','ICE70','ICE71','ICE72','ICE73','ICE74','ICE75','ICE76','ICE77','ICE78','ICE79','ICE80',
'ICE81','ICE82','ICE83','ICE84','ICE85','ICE86','ICE87','ICE88','ICE89','ICE90','ICE91','ICE92','ICE93'
,'ICE94','ICE95','ICE96','ICE97','ICE98','ICE99','ICE100','ICE101','ICE102','ICE103','ICE104','ICE105')
$folder             = "C:\code\demo3\MSIPackage"
$global:CatalogName = 'MyNewCatalog'
$ConnectionString   = 'PROVIDER=MSOLEDBSQL19;Data Source=localhost;Initial Catalog=' +
$global:CatalogName + ';Integrated Security=SSPI;'
$SCCMTargetGroup    = "Applications"
$sAACProjectFile    = "c:\code\script\AACText.aacx"

################################################################
# Non-User Settings
################################################################
$shive              = "HKLM:\SOFTWARE\Wow6432Node\InstallShield\AdminStudio\15.0\"
$slocation          = "Product Location"
$sAsLoc             = (Get-ItemProperty $shive $slocation).$slocation
$sCurrentLoc        = [Environment]::CurrentDirectory=(Get-Location-PSProvider FileSystem).ProviderPath
$sAsLoc             = $sAsLoc + "Common\"
$global:oPkgArray   = @()
$global:oPkgArrayError = @()
$global:oPkgArrayPass = @()
$global:oPkgArrayFail = @()
```

```
#################################################################
# Functions
#################################################################
function Import ($s)
{
    $f = [System.IO.File]::GetAttributes($s)
    $d = ($f -band [System.IO.FileAttributes]::Directory)
    if (!$d)
    {
        Write-Host 'Importing:' $s-foregroundcolor white
        $obj = Invoke-ASImportPackage-PackagePath $s
        if ($obj.GetType().FullName-eq 'AdminStudio.Platform.Helpers.PackageHelper')
        {
            $global:oPkgArray = $global:oPkgArray + $obj
        }
        else
        {
            Write-Host 'Failured to import:' $s-foregroundcolor red
            $global:oPkgArrayError = $global:oPkgArrayError + $obj
        }
    }
}

function ImportFolder ()
{
    [String] $InputFolder = Read-Host "Enter folder to import from (Blank for default)"

    if ($InputFolder)
    {
        $folder = $InputFolder
    }

    if ($folder)
    {
        Write-Host 'Importing Applications from' $folder-foregroundcolor yellow
        foreach ($file in Get-Childitem-include $DefaultExt-Recurse $folder)
        {
            Import ($file)
        }
        Write-Host 'Packages that Import Succeeded:' $global:oPkgArray.Count
        Write-Host 'Packages that Import Failed:' $global:oPkgArrayError.Count
    }
}

function LoadDLL ($s)
{
    $FileName = $sAsLoc + $s
    import-module-name $FileName
}

function PrepAS ()
{
    LoadDLL 'AdminStudio.Platform.PowerShellExtensions.dll'
    LoadDLL 'AdminStudio.Utilities.dll'
    LoadDLL 'AdminStudio.SCCM.Model.dll'
```

```
    Set-ASConfigPlatform-ConnectionString   $ConnectionString
}

function Write-Host-Indent ()
{
    Write-Host '                '-nonewline
}

function Write-Host-Drawline ()
{
    Write-Host '************************************'-foregroundcolor yellow
}

function Write-Heading ($s)
{
    Write-Host-Drawline
    Write-Host $s
    Write-Host-Drawline
}

function Write-Host-Timestamp ()
{
    $tEnd = Get-Date
    $tDiff = $tEnd - $tBegin
    #Write-Host 'End:' $tEnd-foregroundcolor gray
    Write-Host 'Total Time:' $tDiff.Hours' hours' $tDiff.Minutes' minutes ' $tDiff.Seconds 'seconds'
}

function Write-ShorterName ($s)
{
    $s=$s.Replace("The Windows Installer database is scanned for ", "")
    Write-Host ' '-nonewline
    if ($s.Length -gt 55)
    {
          Write-Host $s.Substring(0,55)-foregroundcolor white
    }
    else
    {
          Write-Host $s-foregroundcolor white
    }
}

function WriteVirtReadiness ($Text)
{
    Write-Host-Indent
    Write-Host-Indent
    Write-Host 'Blocker:'$Text-foregroundcolor white
}

function DisplayVirtReadiness ($Package)
{
    $VirtResult = Get-ASVirtualReadiness-PackagePath $Package.FileName-PackageId $Package.RowID
    if ($VirtResult-band 32)
    {
        WriteVirtReadiness("ClickOnce")
    }
```

```
if ($VirtResult-band 64)
{
    WriteVirtReadiness("Shell Extension")
}
if ($VirtResult-band 128)
{
    WriteVirtReadiness("OS Integrated")
}
if ($VirtResult-band 256)
{
    WriteVirtReadiness("Boot Service")
}
if ($VirtResult-band 512)
{
    WriteVirtReadiness("Too Large")
}
if ($VirtResult-band 1024)
{
    WriteVirtReadiness("Surrogate DLL")
}
if ($VirtResult-band 2048)
{
    WriteVirtReadiness("COM Plus")
}
if ($VirtResult-band 4096)
{
    WriteVirtReadiness("Device Driver")
}
if ($VirtResult-band 8192)
{
    WriteVirtReadiness("Questionable")
}
if ($VirtResult-band 16384)
{
    WriteVirtReadiness("Unsuitable")
}
if ($VirtResult-band 32768)
{
    WriteVirtReadiness("64-Bit Package")
}
if ($VirtResult-band 65536)
{
    WriteVirtReadiness("ASP.NET/IIS Application")
}
if ($VirtResult-band 131072)
{
    WriteVirtReadiness("WMI Provider")
}
if ($VirtResult-band 262144)
{
    WriteVirtReadiness("J2EE Application Server")
}
if ($VirtResult-band 524288)
{
    WriteVirtReadiness("Unsupported Applicationr")
}
```

```
    if ($VirtResult-band 1048576)
    {
        WriteVirtReadiness("Unsupported Application")
    }
    if ($VirtResult-band 2097152)
    {
        WriteVirtReadiness("URL Protocol")
    }
    if ($VirtResult-band 4194304)
    {
        WriteVirtReadiness("Default Program")
    }
}

function Test ($o)
{
    Write-Host 'Testing Package:' $o.DisplayedProductName-nonewline-foregroundcolor white
    Write-Host ' RowId:' $o.RowID-foregroundcolor gray
    $oTestResults = Test-ASPackage-PackageId $o.RowID
    $errors = 0;
    $warn = 0;
    foreach ($oTestResult in $oTestResults.Stats)
    {
          $errors =  $errors + $oTestResult.Errors
          $warn   =  $warn +   $oTestResult.Warnings
    }
    Write-Host-Indent
    Write-Host 'Errors:' $errors-foregroundcolor red
    Write-Host-Indent
    Write-Host 'Warnings:' $warn-foregroundcolor yellow
    if ($errors -eq 0)
    {
        Write-Host-Indent
        Write-Host 'Virtualization Readiness:'
        $global:oPkgArrayPass = $global:oPkgArrayPass + $o
        DisplayVirtReadiness ($o)
    }
    else
    {
        $global:oPkgArrayFail = $global:oPkgArrayFail + $obj
    }
}

function TestImportedPackages ($Array)
{
    $global:oPkgArrayPass = @()
    $global:oPkgArrayFail = @()
    foreach ($oPkg in $Array)
    {
       Test ($oPkg);
    }
}

function ConvertToAppV5 ($o)
{
    $ext = $o.FileName.Substring($o.FileName.Length - 3,3)
```

```
    $ext = $ext.ToLower()
    if ($ext -eq 'sft')
    {
        Write-Host 'Converting Package:' $o.DisplayedProductName-nonewline-foregroundcolor white
        Write-Host ' RowId:' $o.RowID-foregroundcolor gray
        $oPackage = Invoke-ASConvertPackageEx-PackagePath $o.FileName-BuildAppV
        $oPackage
    }
    else
    {
        Write-Host 'Skipping non SFT Package:' $o.DisplayedProductName-foregroundcolor gray
    }
}

function ConvertApp5ImportedPackages ($Array)
{
    foreach ($oPkg in $Array)
    {
       ConvertToAppV5 ($oPkg);
    }
}

function ConvertMSIToAppV5 ($o)
{
    $ext = $o.FileName.Substring($o.FileName.Length - 3,3)
    $ext = $ext.ToLower()
    if ($ext -eq 'msi')
    {
        Write-Host 'Converting Package:' $o.DisplayedProductName-nonewline-foregroundcolor white
        Write-Host ' RowId:' $o.RowID-foregroundcolor gray
        Invoke-ASConvertPackageEx-PackagePath $o.FileName-BuildAppV-BuildSymantec
        $oPackage
    }
    else
    {
        Write-Host 'Skipping non MSI Package:' $o.DisplayedProductName-foregroundcolor gray
    }
}

function ConvertApp5FomMSIImportedPackages($Array)
{
    Set-ASConfigPlatform-AACSettingsFile $sAACProjectFile
    foreach ($oPkg in $Array)
    {
       ConvertMSIToAppV5 ($oPkg);
    }
}

function CreateNewCatalog ()
{
    $global:oPkgArray   = @()
    [String] $global:CatalogName = Read-Host "Enter New Catalog Name (Blank to default)"

    ###############################################################
    # Create Catalog
    ###############################################################
```

```
    if ($global:CatalogName)
    {
        Write-Host 'Creating New Catalog' $global:CatalogName-foregroundcolor yellow
        New-ASCatalog-CatalogName $global:CatalogName
        Write-Host
    }
}

function DisableAllTests ()
{
    Write-Host 'Disabling All Tests...'
    foreach ($Test in $TestsToDisable)
    {
       $TestDetails = Get-ASTestDetails-TestId $Test
       if ($TestDetails)
       {
            # Write-Host 'Disabling Test:' $Test -nonewline -foregroundcolor yellow
            # Write-ShorterName ($TestDetails.TestBriefDescription)
            $State = Set-ASTestState-TestId $Test-TestState 0
       }
    }
}

function EnableSelectedTests ()
{
    foreach ($Test in $TestsToEnable)
    {
       $TestDetails = Get-ASTestDetails-TestId $Test
       if ($TestDetails)
       {
            Write-Host 'Enabling Test:' $Test-nonewline-foregroundcolor yellow
            Write-ShorterName ($TestDetails.TestBriefDescription)
            $State = Set-ASTestState-TestId $Test-TestState 1
       }
    }
}

function OutputPackages ($Array)
{
    if ($Array.Count -gt 0)
    {
        foreach ($oPkg in $Array)
        {
            $oPkg.FileName
        }
    }
    else
    {
         Write-Host 'Empty List'
    }
}

function DisplayPackages([int] $LevelPack, [array] $Packages)
{
   for ($i=0; $i -lt $LevelPack; $i++)
   {
```

```
       Write-Host  '      '  -nonewline
   }

    foreach ($Package in $Packages)
    {
          Write-Host  '      -  ' -nonewline-foregroundcolor Yellow
          Write-Host  $Package.ProductName-nonewline-foregroundcolor white
          Write-Host  ' '-nonewline
          Write-Host  $Package.ProductVersion-nonewline-foregroundcolor yellow
          Write-Host  ' '-nonewline
          Write-Host  '[Type=' $Package.Flags ']'  -foregroundcolor blue-nonewline
          Write-Host  ' '-nonewline
          Write-Host  '[RowId=' $Package.RowId ']'  -foregroundcolor gray
    }
}

function DisplayApplications([int]$LevelApp, [array] $Applications)
{
    foreach ($App in $Applications)
    {
          DisplayPackages $LevelApp (Get-ASCatalogItem-ItemId $App.RowId -ItemType 'Package')
    }
}

function DisplayApplicationGroups ([int]$Level, [array] $Group)
{
    foreach ($item in $Group)
    {
       for ($i=0; $i -lt $Level; $i++)
       {
           Write-Host  '      '  -nonewline
       }

       if ($item.Description-eq 'Application Group')
       {
          Write-Host  '   +  '-nonewline-foregroundcolor Yellow
          Write-Host  $item.GroupName-nonewline-foregroundcolor white
          Write-Host  ' '-nonewline
          Write-Host  '[RowId=' $item.RowId ']' -foregroundcolor gray
          DisplayApplications $Level (Get-ASCatalogItem-ItemId $item.RowId-ItemType 'Application')
        }
        else
        {
          Write-Host  '   +  '-nonewline-foregroundcolor Yellow
          Write-Host  $item.GroupName-foregroundcolor gray-nonewline
          Write-Host  ' ' -nonewline
          Write-Host  '[RowId=' $item.RowId ']'  -foregroundcolor gray
          DisplayApplicationGroups ($Level + 1) (Get-ASCatalogItem-ItemId $item.RowId-ItemType 'Group')
        }
    }
}

function DisplayPackageTree ()
{
    Write-Host  '+  Applications'
    DisplayApplicationGroups 0  (Get-ASCatalogItem-ItemId 1-ItemType 'Group')
```

```
}

function Menu ()
{
    do
    {
      Write-Host
      Write-Host-Drawline
      Write-Host 'Catalog Name:' $global:CatalogName-foregroundcolor white
      Write-Host-Drawline
      Write-Host '1. Create a Catalog'-foregroundcolor white
      Write-Host '2. Import from a folder'-foregroundcolor white
      Write-Host '3. Enable Only Industry Standard Windows 7 32 bit Tests'-foregroundcolor white
      Write-Host '4. Test imported packages'-foregroundcolor white
      Write-Host '5. List packages with 0 Errors '-foregroundcolor white
      Write-Host '6. List packages with more than 0 Errors '-foregroundcolor white
      Write-Host '7. List all imported packages'-foregroundcolor white
      Write-Host '8. Convert ALL Imported AppV4 Packages to AppV5'-foregroundcolor white
      Write-Host '9. Convert ALL Imported MSI Packages to AppV5'-foregroundcolor white
      Write-Host '10. Convert packages with 0 Errors Imported MSI Packages to AppV5'-foregroundcolor
white
      Write-Host '11. Display Package Tree'-foregroundcolor white
      Write-Host
      Write-Host '0. Exit'-foregroundcolor white
      Write-Host
      [String] $menu = Read-Host "Enter Option"
      $tBegin = Get-Date
      if ($menu-eq '1')
      {
          CreateNewCatalog
      }
      elseif ($menu-eq '2')
      {
          ImportFolder
      }
      elseif ($menu-eq '3')
      {
          DisableAllTests
          EnableSelectedTests
      }
      elseif ($menu-eq '4')
      {
          TestImportedPackages ($global:oPkgArray)
      }
      elseif ($menu-eq '5')
      {
          Write-Heading 'Applications ready for Windows 7:'
          OutputPackages ($global:oPkgArrayPass)
      }
      elseif ($menu-eq '6')
      {
          Write-Heading 'Applications NOT ready for Windows 7:'
          OutputPackages ($global:oPkgArrayFail)
      }
      elseif ($menu-eq '7')
      {
```

```
            Write-Heading 'All Packages:'
            OutputPackages ($global:oPkgArray)
        }
        elseif ($menu-eq '8')
        {
            ConvertApp5ImportedPackages($global:oPkgArray)
        }
        elseif ($menu-eq '9')
        {
            ConvertApp5FomMSIImportedPackages ($global:oPkgArray)
        }
        elseif ($menu-eq '10')
        {
            ConvertApp5FomMSIImportedPackages ($global:oPkgArrayPass)
        }
        elseif ($menu-eq '11')
        {
            DisplayPackageTree
        }
        Write-Host-Timestamp
    }
    while ($menu-ne '0')
}

#################################################################
# Main Loop
#################################################################
cd $sAsLoc
Write-Host-Drawline
Write-Host 'Default Directory =' $folder-foregroundcolor gray
Write-Host 'Default ConnectionString =' $ConnectionString-foregroundcolor gray
Write-Host 'AdminStudio Directory =' $sAsLoc-foregroundcolor gray
Write-Host-Drawline
PrepAS

#################################################################
# Run Interactively
Menu

#################################################################
# You do not need to use the Menu, you could just call:
#
#ImportFolder
#DisableAllTests
#EnableSelectedTests
#TestImportedPackages
#ConvertApp5FomMSIImportedPackages ($global:oPkgArrayPass)
#
#This would:
#  1. Import Folder of packages
#  2. Enable tests I care about
#  3. Test the packages
#  4. Convert any packages with 0 errors convert to App V5 and Symantec SWV
#################################################################
```

```
cd $sCurrentLoc
```

Notes

In this script, make sure that you define the following parameters correctly:

```
$global:CatalogName = 'MyNewCatalog'
```

```
$ConnectionString = 'PROVIDER=MSOLEDBSQL19;Data Source=localhost;
Initial Catalog=' + $global:CatalogName + ';Integrated Security=SSPI;'
```

Return Values

A list of the root items of the specified type is returned: `Group`, `Application`, or `Package`.

Get-ASConfigPlatform

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Get-ASConfigPlatform` cmdlet to retrieve configuration information, such as to retrieve the database connection string to which the current PowerShell session is configured.

Example

```
Get-ASConfigPlatform -ConnectionString
```

For example, if the `Get-ASConfigPlatform` cmdlet is used with the `-ConnectionString` parameter, the database connection string to which the current PowerShell session was configured using `Set-ASConfigPlatform` cmdlet will be returned, such as:

```
PROVIDER=MSOLEDBSQL19;Data Source=ADMIN-PC;Initial catalog=TesCatalog;Integrated Security=SSPI
```

Parameters

The `Get-ASConfigPlatform` cmdlet has the following parameters:

Table 23-8 • Get-ASConfigPlatform Parameters

Parameter	Description
[ConnectionString]	Use to specify the connection string for Application Catalog.
[Group]	Use to specify the group to import packages into. The group should already exist in the Application Catalog.
[AACSettingsFile]	Specify the Automated Application Converter project file (.aacx) to use for all conversion tasks. This setting can be overridden by individual conversion cmdlets.
[OutputPath]	Specify the default output folder under which all virtualized packages will be stored. This setting can be overridden by individual conversion cmdlets.

Table 23-8 • Get-ASConfigPlatform Parameters

Parameter	Description
[HardTimeout]	Hard time-out (in minutes) for the package installation.
[SoftTimeout]	Soft time-out (in minutes) for the package installation.
[BuildAppV]	Build Microsoft App-V packages (`*.sft`). Specify **`0`** (do not build) or **`1`** (build).
[BuildXenApp]	Build Citrix XenApp profiles (`*.profile`). Specify **`0`** (do not build) or **`1`** (build).
[BuildThinApp]	Build VMWare ThinApp packages (`*.exe`). Specify **`0`** (do not build) or **`1`** (build).
[BuildMSI]	Build Windows Installer packages (`*.msi`). Specify **`0`** (do not build) or **`1`** (build).
[AppVServerHost]	Host name portion of the server location for App-V packages.
[AppVServerPort]	Port number portion of the server location for App-V packages.
[AppVServerProtocol]	Protocol portion of the server location for App-V packages.

Return Values

The connection string to which the current PowerShell session is configured is returned.

Get-ASApplicationDeploymentSummary

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Get-ASApplicationDeploymentSummary` cmdlet to obtain the deployment history of a given distribution system.

Example

```
Get-ASApplicationDeploymentSummary -ConnectionName MySCCM2012Connection
```

Parameters

The `Get-ASApplicationDeploymentSummary` cmdlet has the following parameters:

Table 23-9 • Get-ASApplicationDeploymentSummary Parameters

Parameter	Description
ConnectionName	Name of the distribution system connection for which to obtain deployment summary data. This is the same data that is displayed on the **Server Options > Distribution System** tab of the Application Catalog **Options** dialog box. ***Note*** *• For information on creating a named connection to a deployment system, see Creating Multiple Named Connections to Distribution Systems.*

Return Values

This cmdlet returns an `ApplicationDeploymentInfo` object, such as the following:

```
public class ApplicationDeploymentInfo
{
    public int groupId { get; private set; }
    public string appId { get; private set; }
    public string scopeId { get; private set; }
    public string customId { get; private set; }
    public int revision { get; private set; }
    public string publishDate { get; private set; }
}
```

Get-ASDeploymentSystemPackageTree

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Get-ASDeploymentSystemPackageTree` cmdlet to query a deployment system, such as System Center 2012 Configuration Manager, for a list of all of its applications and groups. This list contains a unique application ID for each of the applications in the deployment system. Using these IDs, you can import applications into the Application Catalog using the Invoke-ASImportAppFromDeploymentSystem cmdlet.

Example

```
Get-ASDeploymentSystemPackageTree -SystemConnectionName SCCM2012
```

Output

The output of the `Get-ASDeploymentSystemPackageTree` cmdlet is a list of the applications with their application IDs in the specified deployment system, such as this list which was generated for a System Center 2012 Configuration Manager server:

```
<root name="Applications">
    <Group name="QA" id="16777226">
        <Application id="16785243" name="Evernote" ObjectType="Application"
            appId="Application_5d4c9ec2-9279-4fd1-ad0b-b2d2a36dd268" />
        <Application id="16785247" name="Basic-1" ObjectType="Application"
            appId="Application_16435057-dd35-41c0-822f-88055ee0bb01" />
        <Application id="16785271" name="Blender" ObjectType="Application"
            appId="Application_356c4eb2-b6a2-4f00-bcaa-dfb71991dde0" />
        <Application id="16785279" name="Create!tools_5_5" ObjectType="Application"
            appId="Application_a31b8a48-37e9-42ae-9959-012179e6bbce" />
        <Application id="16785308" name="AdobeFlash" ObjectType="Application"
            appId="Application_bf185866-33ec-4ddb-968a-3771e5ee7f5a" />
    </Group>
</root>
```

Parameters

The `Get-ASDeploymentSystemPackageTree` cmdlet has the following parameters:

Table 23-10 • Get-ASDeploymentSystemPackageTree Parameters

Parameter	Description
-SystemConnectionName	Use to specify a named connection to a deployment system. See Creating Multiple Named Connections to Distribution Systems for more information.

Note *• To import an application using the Application ID identified using the* `Get-ASDeploymentSystemPackageTree` *cmdlet, see Invoke-ASImportAppFromDeploymentSystem.*

Return Values

XML output containing a list of all of a deployment system's applications and groups is returned.

Get-ASDistributionProperty

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Important *• Starting in AdminStudio 2021 R2, Get-ASDistributionProperty is obsolete and Get-ASProperty should used instead.*

You can use the `Get-ASDistributionProperty` cmdlet to retrieve the property value of the distribution system for a package. The property name will be fetched from the `Property_PluginId` table for the corresponding distribution system. For example, for Casper server distribution system, the property will be fetched from `[Property_D66C6D178DD549419D129CED61BA4C9E]`.

Example

The following is an example of the `Get-ASDistributionProperty` cmdlet:

```
Get-ASProperty -PackageId 9 -SystemConnectionName 'Altiris' -PropertyName 'PackageName'
```

Important • *All parameters are mandatory.*

Parameters

The `Get-ASDistributionProperty` cmdlet has the following parameters:

Table 23-11 • Get-ASDistributionProperty Parameters

Parameter	Description
PackageId	(Required) Use to specify the ID number of the package that you want to get distribution properties for. The ID number is obtained from the `cstblPackage` table.
-SystemConnectionName	(Required) Use to specify a named connection to a deployment system. See Creating Multiple Named Connections to Distribution Systems for more information.
PropertyName	(Required) Use to specify the name of the property that you want to get. The name of the property is fetched from the `Property_PluginId` table for the corresponding distribution system.

Returns

The corresponding property value's display text is returned.

Note • *The value of a property that is stored may not be the value which is displayed in the AdminStudio user interface. For example, Workspace ONE has a property called `pushmodeinternal` (enum) and one of its values is `ondemand`. However, in the AdminStudio user interface, this value is displayed as `On Demand`. The `Get-ASDistributionProperty` cmdlet returns the display value (`On Demand`), not the stored value (`ondemand`).*

Get-ASKeywords

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Get-ASKeywords` cmdlet to return a comma-delimited list of App Portal catalog item keywords in the Application Catalog's ASKeywords table.

Example

The following is an example of the `Add-ASKeywords` cmdlet:

```
Get-ASKeywords
```

After creating the keywords, you can use the `Set-ASProperty` cmdlet to assign keywords to an application, as described in the Set-ASProperty topic under Keywords.

Note *• Keywords are created using either the Add-ASKeywords cmdlet or on the* ***Edit Keywords*** *dialog box, as described in Specifying Catalog Item Keywords. Keywords are assigned to an application using the Set-ASProperty cmdlet or using the* ***Keywords*** *dialog box.*

Return Values

Returns a comma-delimited list of App Portal catalog item keywords in the Application Catalog's `ASKeywords` table.

Get-ASPackage

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Get-ASPackage` cmdlet to return a package object, given the `PackageId`. The default connection string set in the platform settings file is used to query the Application Catalog. If a package with the specified ID is found, it is returned.

Examples

```
$oPackage = Get-ASPackage -PackageId 10
'Package Path is: " + $oPackage.FileName
```

Parameters

The Get-ASPackage cmdlet has the following parameters:

Table 23-12 • Get-ASPackage Parameters

Parameter	Description
PackageId	ID of the package that will be returned.

Return Values

If a package with the specified ID is found, its package object is returned.

Get-ASPackageTestSummary

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the Get-ASPackageTestSummary cmdlet to return a summary of various tests performed for the package that is specified using the -PackageId parameter.

Examples

```
Get-ASPackageTestSummary -PackageId 35
```

Parameters

The Get-ASPackageTestSummary cmdlet has the following parameters:

Table 23-13 • Get-ASPackageTestSummary Parameters

Parameter	Description
PackageId	ID of the package that will be returned.

Sample

Below is sample code using the Get-ASPackageTestSummary cmdlet.

```
function ASPackageTestSummary ()
{
    [String] $Item = Read-Host "Enter Package to see summary of"
    if ($Item)
    {
        Write-Host 'Return Value: '
        $oTestResults = Get-ASPackageTestSummary $Item
        foreach ($oTestResult in $oTestResults)
        {
            Write-Host ($oTestResult.CategroyName)-foregroundcolor yellow
```

```
            Write-Host '     Errors:' ($oTestResult.TotalErrors)-foregroundcolor white
            Write-Host '     Warnings:' ($oTestResult.TotalWarnings)-foregroundcolor white
            Write-Host '    Overall Assessment:' ($oTestResult.OverallAssessment)-foregroundcolor white
        }
    }
}
```

Return Values

A summary of the tests that were performed for the specified package is returned.

Get-ASPackageFeedDetails

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition*

The `Get-ASPackageFeedDetails` cmdlet helps you to fetch the FileName, SilentCommandLineSwitches, etc. for the given PackageFeedId

Tip • *The below returned values can be used for:*

- ***FileName*** *returned will be used in Invoke-ASPackageFeedDownload to download the application from the* ***Package Feed Module***
- ***SilentCommandLineSwitches*** *returned will be used in Invoke-ASImportPackage as [InstallCommandLine] parameter to import the application.*

Example

```
Get-ASPackageFeedDetails -PackageFeedId 2
```

Parameter

The `Get-ASPackageFeedDetails` has the following parameter:

Table 23-14 • Get-ASPackageFeedDetails Parameters

Parameter	Description
PackageFeedId	ID of the package that will be returned ***Note*** • ***PackageFeedId*** *is one of the returned value of Invoke-ASPackageFeedSearch*

Return Values

The product details for the specified package feed ID or error message **Specify a valid package feed id** is returned.

Get-ASProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Get-ASProperty` cmdlet to return the value for a property specified using the `-PropertyName` parameter associated to a specified package specified using `-PackageId` parameter. You can use new -**SystemConnectionName** switch to set the property value of the distribution system for a package. The property name will be fetched from the Property_PluginId table for the corresponding distribution system. For example, for Casper server distribution system, the property will be fetched from [Property_D66C6D178DD549419D129CED61BA4C9E]. The property value should be the display text that is displayed in the AdminStudio user interface.

Note • *Starting in AdminStudio 2021 R2, Get-ASDistributionProperty is obsolete and Get-ASProperty should used instead.*

Note • *The Get-ASProperty cmdlet also supports* ***ASCMPackageData*** *table properties. This is a dynamic table that stores some of the properties of the new package types such as .apk, .ipa, .dmg, etc. These property values can be fetched using this cmdlet.*

Example

```
Get-ASProperty -PackageId 35 -PropertyName "AutoInstall"
Get-ASProperty -PackageId 9 -SystemConnectionName 'Altiris' -PropertyName 'PackageName'
```

Parameters

The `Get-ASProperty` cmdlet has the following parameters:

Table 23-15 • Get-ASProperty Parameters

Parameter	Description
PackageId	ID of the package that will be returned.
PropertyName	Name of the property. ***Note*** • *The name of the property is fetched from the Property_PluginId table for the corresponding distribution system to set properties related to other Distribution System except ConfigMgr.*
SystemConnectionName	Use to specify a named connection to a deployment system. For more information, see Creating Multiple Named Connections to Distribution Systems.

Note • *For more information on parameters, see Set-ASProperty.*

Return Values

The property value for the specified property for the specified package is returned.

Get-ASTestDetails

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Get-ASTestDetails` of the AdminStudio PowerShell Cmdlets to display the details of an application compatibility or Microsoft ICE test that is run using the Test-ASPackage cmdlet. For example:

Example

```
Get-ASTestDetails -TestId nnnn
```

Parameters

The `Get-ASTestDetails` cmdlet has the following parameters:

Table 23-16 • Get-ASTestDetails Parameters

Parameter	Description
TestId *nnnn*	Use to specify the ID of the application compatibility or Microsoft ICE test that you want to see a description of, where *nnnn* is the test ID. The test ID number can be found in the results that are generated by the `Test-ASPackage` cmdlet when the `-DetailedResults` parameter is used.

Return Values

The details of an application compatibility or Microsoft ICE test that was run using the `Test-ASPackage` cmdlet are returned.

Get-ASTestState

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Get-ASTestState` cmdlet to return the test state (selected or not selected) of a given test.

Example

The following is the syntax used to return the test state of a given test:

```
Get-ASTestState -TestId nnnnn
```

For example:

```
Get-ASTestState -TestId ICE33
```

Parameters

The ASTestState cmdlet has the following parameters:

Table 23-17 • ASTestState Parameters

Parameter	Description
TestId	Use to specify the ID number of the test that you are checking the test state of. This is the same ID number that identifies the test on the Application Catalog **Select Tests to Execute** dialog box.

Return Values

One of the following values is returned:

- **True**—Test is selected to run.
- **False**—Test is not selected to run.

Get-ASOptionProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

Use the `Get-ASOptionProperty` to obtain the property value for the given property name.

Example

```
Get-ASOptionProperty -OptionType PackageFeedOptions -PropertyName PackageFeedDownloadPath
```

Parameters

The cmdlet `Get-ASOptionProperty` has the following parameter:

Table 23-18 • Get-ASOptionProperty Parameters

Parameter	Description
PropertyName	Use to specify the below property names that you want to set: • PSTemplateLocation • PSOutputLocation • PSWrapOnImport • EXETemplateLocation • EXEOutputLocation • EXEWrapOnImport • PackageFeedDownloadPath

Return Value

The property value for the given property name is returned

See Also
Set-ASOptionProperty

Get-ASVirtualReadiness

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Get-ASVirtualReadiness` cmdlet to obtain the virtualization readiness status of a given package.

Example

```
Get-ASVirtualReadiness -PackagePath "\\111.22.33.44\Packages\win32-setup.msi" -PackageId 425
```

Parameters

The `Get-ASVirtualReadiness` cmdlet has the following parameters:

Table 23-19 • Get-ASVirtualReadiness Parameters

Parameter	Description
PackagePath	Mandatory parameter which specifies the path to the package that you want to obtain the virtualization readiness status of.

Table 23-19 • Get-ASVirtualReadiness Parameters

Parameter	Description
[PackageId]	Specify the package ID of package that you are testing so that the virtualization readiness status returned by the `Get-ASVirtualReadiness` cmdlet will be stored in the Application Catalog.

Return Values

If an error or warning is generated, one of the following values is returned

Table 23-20 • Error and Warning Return Values

Value	Type	Name	Description
32		**Click Once**	Package contains a ClickOnce application. ClickOnce is a per-user installation format that is often incompatible with the per-machine nature of virtual package deployment. A ClickOnce application also may try to automatically update itself, which results in invalid versioning in the application virtualization client.
64		**Shell Extension**	Package contains a shell extension. Shell extensions extend Windows Explorer and cannot be loaded from a virtual package. This extension may be critical to the use of this application, and, if so, this application will not function when virtualized. However if this extension is non-critical, the application may function when virtualized.
128		**OS Integrated**	Package contains files that are closely integrated with the operating system. The files that make up applications like Internet Explorer or Windows Media Player, or frameworks like the .NET Framework, do not make good candidates for virtualization. These files should instead be installed locally on the machine.
256		**Boot Service**	Package contains a service that starts at boot-time. Virtualized services are limited to the lifetime of the virtual application, so services that must start at boot-time do not make good candidates for virtualization to App-V or XenApp formats. It may be possible to extract this service such that it can be installed locally on the machine and allow the rest of the package to be virtualized.
512		**Too Large**	Package contains more than 4 GB of files. Since App-V 4.x and XenApp do not support packages that contain more than 4 GB of files, this application cannot be successfully virtualized to App-V 4.x or XenApp as an uncompressed package. However, if the compressed size of the package is less than 4 GB, then this application can be virtualized to these formats as a compressed package.

Table 23-20 • Error and Warning Return Values

Value	Type	Name	Description
1024		**COM Surrogate DLLs**	Package contains a COM DLL that uses surrogate virtualization. App-V, XenApp, and ThinApp do not support COM DLL surrogate virtualization, so this package may not work correctly if virtualized..
2048		**COM Plus**	Package contains a COM Plus component. App-V, XenApp, and ThinApp do not support COM+ components, so this package may not work correctly if virtualized.
4096		**Device Drive**	Package contains a device driver. System-level drivers such as print drivers or USB device drivers do not work from a virtualized environment. It may be possible to extract this driver such that it can be installed locally on the machine and allow the rest of the package to be virtualized.
8192		**Questionable**	Package is questionable for conversion.
16384		**Unsuitable**	Package is unsuitable for conversion.
32768		**64-Bit Package**	Package is a 64-bit package. XenApp and ThinApp do not support virtualization of 64-bit packages.
65536		**ASP.NET/IIS Application**	Package contains an ASP.NET or IIS application component, which is not supported by App-V 4.x, App-V 5.x, XenApp, and ThinApp. If the ASP.NET or IIS application component is not an important part of the application, or if it can be separately installed from the package, this error can be suppressed and ignored.
131072		**WMI Provider**	Package contains a WMI provider component, which is not supported by App-V 4.x, App-V 5.x, XenApp, and ThinApp. If the WMI Provider component is not an important part of the application, or if it can be separately installed from the App-V package, this error can be suppressed and ignored.
262144		**J2EE Application Server**	Package contains a J2EE application server, which is not supported by App-V, XenApp, or ThinApp. If the J2EE application is not an important part of the application, or if it can be separately installed from the package, this error can be suppressed and ignored.
524288		**Unsupported Application**	Package contains an application known to not be a good candidate for virtualization.

Table 23-20 • Error and Warning Return Values

Value	Type	Name	Description
1048576	⚠	**Unsupported Application**	Package contains some files that indicate the presence of unsupported applications such as antivirus software or various server software such as Exchange Server or SQL Server. If these unsupported application components are not an important part of the application, or if they can be separately installed from the package, this error can be suppressed and ignored.
2097152	⚠	**URL Protocol**	Package registers an URL protocol.
4194304	⚠	**Default Program**	Package registers its capabilities in the Default Programs list.

Invoke-ASAppVBulkUpgrade

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Invoke-ASAppVBulkUpgrade` cmdlet is used for bulk conversion of App-V 4.x packages (`.sft`) to App-V 5.x packages (`.appv`).

Example

The following is an example of the `Get-ASAppVBulkUpgrade` cmdlet:

```
Invoke-ASAppVBulkUpgrade -GroupID 11 -UpgradeComments "Bulk Upgrade"
```

Parameters

The `Invoke-ASAppVBulkUpgrade` cmdlet has the following parameters:

Table 23-21 • Invoke-ASAppVBulkUpgrade Parameters

Parameter	Description
GroupID	Specifies the group ID of the group containing legacy App-V packages in the catalog.
UpgradeComments	Specifies the comments to document the upgraded package.

Return Values

A success or failure message is returned along with the details of the failure.

Invoke-ASConvertFolder

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Invoke-ASConvertFolder` cmdlet to convert a folder of packages to specified virtual formats using Automatic Application Converter. The conversion settings specified in the platform settings file are applied across all packages found in the specified folder.

Tip • *If you need to apply settings on per package basis, it is recommended that you use the* `Add-ASPackageForConversion` *cmdlet.*

Examples

```
Invoke-ASConvertFolder -FolderPath C:\Packages -BuildAppV -OutputPath C:\VirtualizedPackages
```

Parameters

The `Invoke-ASConvertFolder` cmdlet has the following parameters:

Table 23-22 • Invoke-ASConvertFolder Parameters

Parameter	Description
FolderPath	Use to specify the path to the folder where the packages to be converted are stored.
[AACSettings]	Use to specify the Automated Application Converter project file to use during conversion. If it is not supplied, a copy of the project file specified in the platform settings file will be used.
[OutputPath]	Use to specify the output folder under which all output will be collected.
[BuildAppV]	Specify this parameter to build App-V packages.
[BuildXenApp]	Specify this parameter to build Citrix XenApp profiles.
[BuildThinApp]	Specify this parameter to build VMware ThinApp packages.
[BuildMSI]	Specify this parameter to build Windows Installer packages.
[HardTimeOut]	Hard time-out (in minutes) for the package installation.
[SoftTimeout]	Soft time-out (in minutes) for the package installation.
[UseSingleStepSnapshot]	Use to specify that you want to use the Snapshot installation technology to repackage the package.

Table 23-22 • Invoke-ASConvertFolder Parameters (cont.)

Parameter	Description
[ApplyTransforms]	If this parameter is used, transforms found in same folder as the package will be used during the conversion process.
[VMPlatform]	Specify platform to use for automated repackaging, such as **600Sx64**, **501**, or **502S**.

Return Values

A success or failure message is returned.

Invoke-ASConvertPackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Note • *In future releases, the Invoke-ASConvertPackage cmdlet will be obsolete.*

AdminStudio has a generic platform API, ASConvertPackage, to support conversions. It accepts XML as input parameter.

Example

```
Invoke-ASConvertPackage -XmlPath "C:\Users\Administrator\Desktop\Test\Conversion.xml"
```

Sample XML Templates

The following sample XML templates are provided:

- Intune Conversion
- Automated Application Converter Conversions

Intune Conversion

```
<?xml version="1.0" encoding="utf-8" ?>
<ConversionInfoData>
<PluginId>3C2F8362-A53C-44A2-B524-748F91FB6864</PluginId>
<TargetType>IntuneWin</TargetType>>
<PackageId>1</PackageId>
<PackageSourceFolder>D:\Packages\App-Vs\App-VPackage\Firefox_Upgrade</PackageSourceFolder>
<PackageOutputFolder>D:\Packages\App-Vs\App-VPackage\Firefox_Upgrade\Output</PackageOutputFolder>
</ConversionInfoData>
```

Automated Application Converter Conversions

```
<?xml version="1.0" encoding="utf-8" ?>
<ConversionInfoData>
<PluginId>63807502-5E5B-4B10-8CB4-AB318FA324A0</PluginId>
<TargetType>Msix</TargetType>>
```

```
<PackageId>1</PackageId>
<AACSettingsFilePath>D:\AAC\MSIXOption.aacx</AACSettingsFilePath>
<CommandLine></CommandLine>
</ConversionInfoData>
```

In this XML file, you need to change the `TargetType` node to one of the following conversion formats:

```
Msi
Msix
hinApp
Profile
Appv4
Appv5
```

Invoke-ASConvertPackageEx

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Note • *In future releases, the Invoke-ASConvertPackageEx cmdlet will be renamed to Invoke-ASConvertPackage.*

The `Invoke-ASConvertPackageEx` cmdlet invokes the Application Catalog Conversion Wizard process to convert a package from one package type to another. You can use the `Invoke-ASConvertPackageEx` cmdlet to:

- Convert an App-V 4.x package to App-V 5.0 format.
- Convert one or multiple Windows Installer packages or legacy installers to virtual packages using default Automated Application Converter settings.

You are required to specify a target type and an Automated Application Converter settings file.

Example

`Invoke-ASConvertPackageEx -PackageID` *`n`* `-TargetType` *`type`* `-AACSettings` *`PathToSettingsFile`*

For example:

```
Invoke-ASConvertPackageEx -PackageID 5 -TargetType AppV5 -AACSettings C:\MyProjectFile\MySettings.aacx
```

Parameters

The `Invoke-ASConvertPackageEx` cmdlet has the following parameters:

Table 23-23 • Invoke-ASConvertPackageEx Parameters

Parameter	Description
PackageId	The ID of the source package to be converted.

Table 23-23 • Invoke-ASConvertPackageEx Parameters (cont.)

Parameter	Description
TargetType	Follow the `-TargetType` parameter with one of the following to identify the deployment type of the converted package: • AppV4 • AppV5 • Profile • ThinApp • Msi • MSIX
AACSettings	Enter the fully qualified path to the `.accx` project file that will be used for the conversion, which contains virtual machine login information and conversion defaults. ***Note*** • *For more information, see Creating an Automated Application Converter Settings File.*
CommandLine	This parameter is used to pass command line switches to the installer during an unattended installation, such as when Automated Application Converter automatically launches a virtual machine to perform repackaging. If this parameter is set, it takes precedence. If this parameter is not set, then any specified command line in the Automated Application Converter plugin options takes precedence. If neither are specified, then Automatic Application Converter automatically uses a basic UI mode for MSI packages.

Return Values

Success or failure messages are returned.

Invoke-ASImportAppFromDeploymentSystem

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Invoke-ASImportAppFromDeploymentSystem` cmdlet to import an application from a deployment system, such as System Center 2012 Configuration Manager, into the Application Catalog using the application ID returned from the Get-ASDeploymentSystemPackageTree cmdlet.

Example

The following is an example of the `Invoke-ASImportAppFromDeploymentSystem` cmdlet and its parameters:

```
Invoke-ASImportAppFromDeploymentSystem -ConnectionName <Name> -SystemDeploymentID <ID>
    -TargetASGroupPath "<Path excluding the root folder ('Applications')>"
```

The following is an example of the `Invoke-ASImportAppFromDeploymentSystem` cmdlet using sample values:

```
Invoke-ASImportAppFromDeploymentSystem -ConnectionName SCCM2012 -SystemDeploymentID 16778779 -
    TargetASGroupPath "Test\SubTest"
```

Parameters

The `Invoke-ASImportAppFromDeploymentSystem` cmdlet has the following parameters:

Table 23-24 • Invoke-ASImportAppFromDeploymentSystem Parameters

Parameter	Description
ConnectionName	Use to specify named connection to a deployment system. See Creating Multiple Named Connections to Distribution Systems for more information.
SystemDeploymentID	The ID of the application you are importing from your deployment system such as: `-SystemDeploymentID 16778779` You obtain this ID by first running the Get-ASDeploymentSystemPackageTree cmdlet.
TargetASGroupPath	Use to specify the destination group in the Application Catalog tree for the imported application. ***Note*** *• When specifying this path. exclude the root folder (*`Applications`*).*

Return Values

A success or failure message is returned.

Invoke-ASImportPackage

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Invoke-ASImportPackage` cmdlet invokes the import process on a single package.

Examples

In this example, the `Orca.msi` package from the given path is imported, as well as the specified transform file.

```
Invoke-ASImportPackage -PackagePath C:\Packages\Orca\Orca.msi -Transforms c:\Packages\Orca\Orca.mst
```

You can also import .exe files into the Application Catalog using the Invoke-ASImportPackage cmdlet:

```
Invoke-ASImportPackage -PackagePath C:\Packages\ABCapp\ABC.exe
```

Parameters

The Invoke_ASImportPackage cmdlet has the following parameters:

Table 23-25 • Invoke_ASImportPackage Parameters

Parameter	Description
PackagePath	Specify the path to the package to be imported.
[Group]	Use to specify the group into which the package will be imported. When specifying the Group parameter, you need to include the Applications root group in the path to the group. For example: `Invoke-ASImportPackage -PackagePath C:\Packages\Orca\Orca.msi -Group "Applications\SubGroup\SubGroup1"` If you do not include the Applications root group in the path, the packages will be imported under the root node instead of the specified folder path.
[Transforms]	List of transforms to apply during the import process. Specify the full paths. When specifying multiple transform files, use commas to separate them.
[Patches]	List of patches to apply during the import process. Specify the full paths. When specifying multiple patches, use commas to separate them.
[InstallCommandLine]	Use to set the package's **Install command line** property. This property will be transferred to ConfigMgr (Formerly called as System Center Configuration Manager) when the package's application is published. `Invoke-ASImportPackage -PackagePath "C:\Packages\Calc2020\Calc2020.msi" -InstallCommandLine "msiexec /i 'Calc2020.msi'"`
[ExistingPackageId]	In the case of reimporting an existing package, use this parameter to specify the existing package's ID.
PackageFeedId	Identify the package you want to download. Invoke-ASImportPackage -PackagePath "C:\Program Files (x86)\AdminStudio Shared\Package Feed Downloads\Notepad++ Team\Notepad++ (x86)\7.9.5" -PackageFeedId 143

Table 23-25 • Invoke_ASImportPackage Parameters (cont.)

Parameter	Description
IsCustomization	Choose one of the following option: 0 - If you choose this option, custom.mst file will not be generated. 1 - If you select this option, custom.mst file will generate Invoke-ASImportPackage -PackagePath 'D:\AdminStudio Shared\Package Feed Downloads\Box\Box Notes\1.4.0\BoxNotesSetup.exe' -PackageFeedId '3075' -IsCustomization 0

Return Values

A success or failure message is returned.

Invoke-ASWrapPackage

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Invoke-ASWrapPackage` to convert a Windows Installer (`.msi`) or installation package (`.exe`) to a wrapped package file (`.ps1/.exe`).

Example

```
Invoke-ASWrapPackage -PackageId 25 -OutputLocation "C:\Users\ssahoo\Desktop" -WrapType Exe
```

Parameters

The `Invoke-ASWrapPackage` has the following parameters:

Table 23-26 • Invoke-ASWrapPackage Parameters

Parameter	Description
PackageID	(Required) Enter the package ID to identify the package you want to wrap.
OutputLocation	(Optional) If you want to change the default packaging location, specified on the **Wrap Options** tab of the Application Catalog **Options** dialog box, use this parameter.
WrapType	Either add .ps1 or .exe

Return Values

A success or failure message is returned.

Invoke-ASPackageCustomize

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

The new Invoke-ASPackageCustomize cmdlet is used to generate the customize transform file (.mst) for the existing msi package which is download/imported from the Package Feed Module.

Examples

```
Invoke-ASPackageCustomize -PackageId 8
```

Parameters

The `Invoke-ASPackageCustomize` cmdlet has the following parameters:

Table 23-27 • Invoke-ASPackageCustomize Parameters

Parameter	Description
PackageId 8	Specify the package id. `Invoke-ASPackageCustomize -PackageId 8`

Return Values

It will successfully generate the customize transform file (.mst).

Invoke-ASPackageFeedSearch

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

The new `Invoke-ASPackageFeedSearch` cmdlet is used to search application in Package Feed Module.

Examples

```
Invoke-ASPackageFeedSearch -ProductName Firefox -Vendor Mozilla
```

```
Invoke-ASPackageFeedSearch -ProductName Firefox
```

```
Invoke-ASPackageFeedSearch  -Vendor Mozilla
```

Tip • *You can search by using the Product Name and/or the Vendor Name*

Parameters

The Invoke-ASPackageFeedSearch cmdlet has the following parameters:

Table 23-28 • Invoke-ASPackageFeedSearch Parameters

Parameter	Description
ProductName	Specify the name of the application need to be downloaded ***Note*** • *Specify at least three characters matching the application name*
Vendor	Specify the vendor name of the application need to be downloaded ***Note*** • *Specify at least three characters matching the vendor name*

Return Values

The details of the application matching the product name and the vendor name or failure message is returned.

Invoke-ASPackageFeedDownload

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition*

The Invoke-ASPackageFeedDownload cmdlet helps you to download the setup file using the parameter **FileName** from the **Package Feed Module**.

Examples

```
Invoke-ASPackageFeedDownload -PackageFeedId '38' -FileName 'SetupDWGTrueView2013_64bit.exe' -
    MD5Checksum 'd35274affa92456b0f0d1f8914856fba'
```

Parameters

The Invoke-ASPackageFeedDownload has the following parameters:

Table 23-29 • Invoice-ASPackageFeedDownload Parameters

Parameter	Description
PackageFeedId	Enter the PackageFeedId to identify the package you want to download. ***Note*** • *The PackageFeedId is the returned value of Invoke-ASPackageFeedSearch*

Table 23-29 • Invoice-ASPackageFeedDownload Parameters

Parameter	Description
FileName	Enter the File name returned for the package feed ID ***Note*** • *The FileName is the returned value of Get-ASPackageFeedDetails*
MD5Checksum	It is used to match the MD5 value of the package which is downloaded from Package Feed Module. This is a mandatory parameter.
DownloadPackageFeedIcon	**Yes** - If you choose this option, then icon file will be downloaded along with the package. **No** - If you choose this option, then icon file will not be downloaded. Invoke-ASPackageFeedDownload -PackageFeedId '3075' -FileName 'BoxNotesSetup.exe' -MD5Checksum '6ac468f060098782b4559d17772e58b3' -DownloadPackageFeedIcon 'Yes'

Return Value

A success or failure message is returned.

Invoke-ASPackageFeedSync

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition*

Use the `Invoke-ASPackageFeedSync` cmdlet to sync package feed data.

Examples

```
Invoke-ASPackageFeedSync
```

Parameter

The cmdlet `Invoke-ASPackageFeedSync` requires no parameter

Return Value

A success or failure message is returned

Invoke-ASPublish

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Invoke-ASPublish` cmdlet to publish a package to a deployment server, such as System Center 2012 Configuration Manager, Citrix XenApp Server, Symantec Altiris Server, or Workspace ONE Server. For publishing, an `ApplicationID` is needed instead of a `PackageID`.

Tip • *If you have the* `PackageID`*, you can determine the* `ApplicationID` *by using the* `Get-ASApplicationID` *cmdlet.*

Specifying an Import-Module Cmdlet

Because publishing requires `AdminStudio.SCCM.Integrator.dll`, you need to specify an `Import-Module` cmdlet either in your PowerShell session or in the PowerShell script.

You can import this module for your PowerShell session as shown below:

```
Import-Module-Name [AdminStudioInstallDirectory]\Common\AdminStudio.SCCM.Integrator.dll
```

You can either specify this import in your PowerShell script or in a PowerShell session at the command prompt.

Examples

First, use a package's `PackageID` to obtain its `ApplicationID`:

```
$oAppID = Get-ASApplicationID -PackageID 10
```

Then, use the `ApplicationID` to publish the package:

```
Invoke-ASPublish -ConnectionName "SCCM12"-ApplicationID 35 -PackageIDs "75,80" -
    TargetGroup "Applications\Marketing" -Password "ABC1234"
```

Parameters

The `Invoke-ASPublish` cmdlet has the following parameters:

Table 23-30 • Invoke-ASPublish Parameters

Parameter	Description
ConnectionName	Use to specify named connection to a deployment system. See Creating Multiple Named Connections to Distribution Systems for more information.
ApplicationID	Specify the `ApplicationID` of the application you are publishing.

Table 23-30 • Invoke-ASPublish Parameters (cont.)

Parameter	Description
PackageIDs	Specify the PackageIDs of the application that you are publishing. ***Note*** • *It is an optional parameter. If you do not specify PackageIDs parameter then entire application will be published.*
[TargetGroup]	Specify the target group on the deployment server that you want to publish this application to. ***Note*** • *If you publish an application with an empty Target Group to Workspace ONE server, the application will be published to the default organization group to which the particular Workspace ONE user belongs to.*
[Password]	Specify the password of the deployment server you are publishing to.

Note • *Workspace ONE permits publishing a single application only once to an Organization Group. Therefore, if you attempt to publish an application to an Workspace ONE Organization Group that already contains that application, the publication will fail.*

Return Values

A success or failure message is returned.

Invoke-ASUpdatePackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Starting with AdminStudio 2016, when you import suite executables (`.exe`) that contain child packages, the child packages are automatically extracted during import and are available for testing.

However, if you are upgrading an existing Application Catalog (from releases prior to AdminStudio 2016) that already contains installers with bundled child packages, you will need to run the `Invoke-ASUpdatePackage` cmdlet on each of those existing installers to extract details on the child packages bundled with those installers.

The `Invoke-ASUpdatePackage` cmdlet is run on an existing `.exe` package to extract the bundled `.msi` details. Along with child package extraction, the `Invoke-ASUpdatePackage` also extracts Java dependency information and identifies the type of suite `.exe`.

Example

The following is an example of the Invoke-ASUpdatePackage cmdlet:

```
Invoke-ASUpdatePackage -PackageId 2
```

Parameters

The Invoke-ASUpdatePackage cmdlet has the following parameters:

Table 23-31 • Invoke-ASUpdatePackage Parameters

Parameter	Description
PackageID	ID of the package from cstblPackage table.

Procedure

To use the Invoke-ASUpdatePackage cmdlet to extract bundled .msi details from an .exe package, perform the following steps.

Task ***To extract bundled .msi details from an .exe package:***

1. Select an .exe package from cstblpackage table.
2. Open a PowerShell cmdlet and set a connection to an existing Application Catalog database.
3. Import the necessary PowerShell assemblies.
4. Enter the following:

   ```
   Invoke-ASUpdatePackage -PackageId 2
   ```

   Information about the bundled .msi packages is extracted. The extracted .msi packages will be listed in the AdminStudio user interface.

Invoke-ASSignPackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

Use the Invoke-ASSignPackage cmdlets to sign the MSIX package.

Example

The following is an example of the Invoke-ASSignPackage cmdlet:

```
Invoke-ASSignPackage -PackageId 1 -CertificateInfoType PfxFile -FilePath
    "E:\Packages\MSIX\Orca\FlexeraDigCert.pfx" -Password "121"
```

Parameters

The `Invoke-ASSignPackage` has the following parameters:

Table 23-32 • Invoke-ASSignPackage Parameters

Parameter	Description
PackageID	(Required) Enter the package ID to identify the package you want to Sign.
CertificateInfoType	Follow the -CertificateInfoType parameter with one of the following to identify the Signing type of the MSIX package: • PfxFile • CertificateName
PfxFile	If the selected certificate info type is PfxFile, Enter the following: • File Path - pfx file path (Required) • Password - Valid Password (Required)

CertificateName	If the selected certificate info type is Certificate Name, Enter the following:
	• Subject - Subject of the certificate (Required)
	• Select one of the following StoreType:
	  ◦ User
	  ◦ System
	  ◦ SpecificStore

SpecificStore	If the selected store type is SpecificStore, Enter the following: • StoreName - Enter the store name

Return Values

A success or failure message is returned.

New-ASCatalog

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `New-ASCatalog` cmdlet to create a new Application Catalog.

Examples

The following is the syntax used to create a new Application Catalog:

```
New-ASCatalog -CatalogName NameOfNewCatalog
```

For example:

```
New-ASCatalog -CatalogName CAT2016FEB
```

Important • *When AdminStudio executes the New-ASCatalog cmdlet, it uses the upgrade.xml file, which contains a list of the SQL scripts that need to be run to create a new Application Catalog. By default, the upgrade.xml file is installed in the Support subdirectory of the AdminStudio installation directory. If you want to create a new Application Catalog using an upgrade.xml file in a different location, you need to provide the path to that file in the New-ASCatalog cmdlet, such as:*

```
New-ASCatalog C:\MyScripts -CatalogName MyNewCatalog
```

Note • *Before using the New-ASCatalog cmdlet to create a new Application Catalog, you need to have already used the Set-ASConfigPlatform cmdlet with the ConnectionString parameter to enter the connection information to the SQL database.*

Parameters

The New-ASCatalog cmdlet has the following parameters:

Table 23-33 • New-ASCatalog Parameters

Parameter	Description
CatalogName	Use to enter a name for the new Application Catalog. Upon successful creation, you will be automatically connected to the new Application Catalog.
[UseSoftwareRepository]	Use to enable the software repository in the new Application Catalog. This parameter requires you to also supply the user name, password, and path to the repository. for example: `New-ASCatalog -CatalogName mycatalog -UseSoftwareRepository SoftwareRepositoryUser=JoeSmith; SoftwareRepositoryPassword=mypassword123; SoftwareRepositoryPath=C:\MyRepository;`
[ScriptPath]	When AdminStudio executes the New-ASCatalog cmdlet, it uses the upgrade.xml file, which contains a list of the SQL scripts that need to be run to create a new Application Catalog. By default, the upgrade.xml file is installed in the Support subdirectory of the AdminStudio installation directory. If you want to create an Application Catalog using an upgrade.xml file in a different location, you need to use this parameter to provide the path to that file: `New-ASCatalog -ScriptPath C:\MyScripts`

Return Values

A success or failure message is returned.

New-ASDistributionConnection

Important • *Starting with AdminStudio 2020, the* `New-ASDistributionConnection` *cmdlet is obsolete. You should instead use* `New-ASDistributionConnectionEx`.

You can use the `New-ASDistributionConnection` cmdlet to use PowerShell to define named connections to ConfigMgr (Formerly called as System Center Configuration Manager), Citrix XenApp Server, Symantec Altiris Management Server, and Workspace ONE Server distribution systems. This enables you to refer to those connection settings by name in AdminStudio PowerShell Cmdlets cmdlets.

Example

```
New-ASDistributionConnection
    -Name SCCM2012
    -PluginID 0
    -ServerAddress 172.01.02.03
    -SiteCode ABC
    -DistributionWindowsAuthentication 0
    -DistributionUser MyDomain\UserName
    -DistributionPassword Password123
    -ShareWindowsAuthentication 0
    -SharePath \\172.01.02.03\SharedLocation
    -ShareUserName MyDomain\UserName
    -SharePassword Passw0rd123
```

Parameters

The `New-ASDistributionConnection` cmdlet has the following parameters:

Table 23-34 • New-ASDistributionConnection Parameters

Parameter	Description
Name	Use to specify the name of this new named connection to a distribution system.
PluginID	Use to specify the plug-in with which the distribution system is associated. This parameter is mapped to the object identifier (OID) of the `ASCMSupportedPackageTypes` database table; you can provide any value available in this table to identify the plug-in.
ServerAddress	Use to specify the distribution server address.
SiteCode	When connecting to a ConfigMgr (Formerly called as System Center Configuration Manager) distribution system, use to specify the site code.

Table 23-34 • New-ASDistributionConnection Parameters (cont.)

Parameter	Description
DistributionWindowsAuthentication	Use to specify whether the distribution connection should use Windows Authentication or a custom user name and password. Available options are: • **0** = Do not use Windows Authentication • **1** = Use Windows Authentication
[DistributionUser]	If not using Windows Authentication, use this parameter to specify the user name to use when connecting to the distribution system.
[DistributionPassword]	If not using Windows Authentication, use this parameter to specify the password to use when connecting to the distribution system.
SharePath	Use to specify the path to which packages are published.
ShareWindowsAuthentication	Use to specify whether the share path should use Windows Authentication or a custom user name and password. Available options are: • **0** = Do not use Windows Authentication • **1** = Use Windows Authentication
[ShareUserName]	If not using Windows Authentication, use this parameter to specify the user name to use when connecting to the share path.
[SharePassword]	If not using Windows Authentication, use this parameter to specify the password to use when connecting to the share path.

Return Values

No values are returned.

New-ASDistributionConnectionEx

Note • *Starting in AdminStudio 2020, New-ASDistributionConnection is obsolete and* `New-ASDistributionConnectionEx` *should used instead.*

AdminStudio has a generic platform API, `New-ASDistributionConnectionEx`, to support creating distribution connections.It accepts XML as an input parameter.

Example

```
New-ASDistributionConnectionEx -XmlPath "C:\test data\ConfigMgr.xml"
```

Example XML Input Files

The following are examples of XML input files for different distribution platforms:

- Configuration Manager
- Intune
- WorkSpace ONE
- Altiris
- Microsoft AppV Server
- Citrix XenApp

Configuration Manager

```
<?xml version="1.0" encoding="utf-8" ?>
<ConfigMgr>
<DistributionInformation>
<PluginId>EDFBE09C-E305-4ABF-BEEB-77032335ACB7</PluginId>
<Name></Name>
<Server></Server>
<SiteCode></SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
</DistributionSystemAuthentication>
<ShareInformation>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
<ShareFolderPath></ShareFolderPath>
</ShareInformation>
</ConfigMgr>
```

Intune

Note • *The `UseClientSecret` switch is false by default.*

```
<?xml version="1.0" encoding="utf-8" ?>
<Intune>
<PluginId>E36D3170-FBEA-4288-9A30-34313599D9DE</PluginId>
<Name></Name>
<UseClientSecret>False</UseClientSecret>
<ClientId></ClientId>
```

```
<TenantIdOrTenantName></TenantIdOrTenantName>
<ClientSecret></ClientSecret>
</Intune>
```

WorkSpace ONE

```
<?xml version="1.0" encoding="utf-8" ?>
<WorkspaceONE>
<DistributionInformation>
<PluginId>F9A0BED5-1B8F-4D1E-BF6C-C724D083BFC7</PluginId>
<Name></Name>
<Server></Server>
<SiteCode></SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
false
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
</DistributionSystemAuthentication>
</WorkspaceONE>
```

Altiris

```
<?xml version="1.0" encoding="utf-8" ?>
<Altiris>
<DistributionInformation>
<PluginId>DD86877B-D338-490A-B579-93DFB09B71AE</PluginId>
<Name></Name>
<Server></Server>
<SiteCode></SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
</DistributionSystemAuthentication>
<ShareInformation>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
<ShareFolderPath></ShareFolderPath>
</ShareInformation>
</Altiris>
```

Microsoft AppV Server

```
<?xml version="1.0" encoding="utf-8" ?>
<AppVServer>
<DistributionInformation>
<PluginId>FE347079-E08D-4782-9557-9B21BCD0FE8A</PluginId>
<Name></Name>
```

```
<Server></Server>
<SiteCode></SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
</DistributionSystemAuthentication>
<ShareInformation>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
<ShareFolderPath></ShareFolderPath>
</ShareInformation>
</AppVServer>
```

Citrix XenApp

```
<?xml version="1.0" encoding="utf-8" ?>
<XenApp>
<DistributionInformation>
<PluginId>D8F3EF54-D321-4A5E-A08F-18F5689D829A</PluginId>
<Name></Name>
<Server></Server>
<SiteCode></SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
</DistributionSystemAuthentication>
<ShareInformation>
<UseWindowsAuthentication>
true
</UseWindowsAuthentication>
<UserName></UserName>
<Password></Password>
<ShareFolderPath>/ShareFolderPath>
</ShareInformation>
</XenApp>
```

Example XML Template for Configuration Manager

The following is an example of how a populated template for Configuration Manager would look like.

```
<?xml version="1.0" encoding="utf-8" ?>
<ConfigMgr>
<DistributionInformation>
<PluginId>EDFBE09C-E305-4ABF-BEEB-77032335ACB7</PluginId>
<Name>CMServer</Name>
<Server>10.80.150.234</Server>
```

```
<SiteCode>CM5</SiteCode>
</DistributionInformation>
<DistributionSystemAuthentication>
<UseWindowsAuthentication>
false
</UseWindowsAuthentication>
<UserName>ar\administrator</UserName>
<Password>password</Password>
</DistributionSystemAuthentication>
<ShareInformation>
<UseWindowsAuthentication>
false
</UseWindowsAuthentication>
<UserName>ar\administrator</UserName>
<Password>password</Password>
<ShareFolderPath>\\10.80.150.234\Publish</ShareFolderPath>
</ShareInformation>
</ConfigMgr>
```

Note • *Though AdminStudio supports Casper distribution server and a connection to the same can be created from the Application Manager user interface, creating a connection to Casper using AdminStudio PowerShell APIs aren't supported.*

New-ASCreateSoftwareTag

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

You can use the `New-ASCreateSoftwareTag` cmdlet to create a new Software Tag.

Examples

The following is the syntax used to create a new Software Tag:

```
New-ASCreateSoftwareTag -PackageID 1
```

Parameters

The `New-ASCreateSoftwareTag` cmdlet has the following parameters:

Table 23-35 • `New-ASCreateSoftwareTag` Parameter

Parameter	Description
PackageID	ID of the package that will be returned from the catalog

Note • *For more information to set the software tag properties see Set-ASSoftwareTagProperties.*

Return Values

A success or failure message is returned.

Remove-ASApplication

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Remove-ASApplication` cmdlet to delete a package using its `ApplicationId`.

Note • *The Remove-ASApplication cmdlet removes the linked packages and applications as well as the targeted packages and applications for each respective operation.*

Example

The following is the syntax used to delete an application:

```
Remove-ASApplication -ID nn
```

For example:

```
Remove-ASApplication -ID 67
```

Parameters

The `Remove-ASApplication` cmdlet has the following parameters:

Table 23-36 • Remove-ASApplication Parameters

Parameter	Description
ID	Specifies the ID for the application which is being removed. (Required)

Returns

One of the following values is returned:

- **0**—Success. The delete operation completed successfully.
- **1**—Insufficient access rights. Permission to **Delete** is not available.
- **2**—Object is locked. The group contains locked virtual packages. You will need to unlock the virtual packages in order to delete the group.
- **3**—General failure. Operation did not complete successfully.
- **4**—Protected group. The deletion of protected groups is prohibited.
- **5**—Item not found. The requested item could not be deleted because it could not be found.

Remove-ASGroup

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Remove-ASGroup` cmdlet to delete a group using its `GroupID`.

Note • *The Remove-ASGroup cmdlet removes the linked packages and applications as well as the targeted packages and applications for each respective operation.*

Example

The following is the syntax used to delete a group:

```
Remove-ASGroup -ID nn
```

For example:

```
Remove-ASGroup -ID 32
```

Parameters

The `Remove-ASGroup` cmdlet has the following parameters:

Table 23-37 • Remove-ASGroup Parameters

Parameter	Description
ID	Specifies the ID for the group which is being removed. (Required)

Returns

One of the following values is returned:

- **0**—Success. The delete operation completed successfully.
- **1**—Insufficient access rights. Permission to **Delete** is not available.
- **2**—Object is locked. The group contains locked virtual packages. You will need to unlock the virtual packages in order to delete the group.
- **3**—General failure. Operation did not complete successfully.
- **4**—Protected group. The deletion of protected groups is prohibited.
- **5**—Item not found. The requested item could not be deleted because it could not be found.

Remove-ASPackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Remove-ASPackage` cmdlet to delete a package using its `PackageId`.

Note • *The Remove-ASPackage cmdlet removes the linked packages and applications as well as the targeted packages and applications for each respective operation.*

Example

The following is the syntax used to delete a package:

```
Remove-ASPackage -ID nn
```

For example:

```
Remove-ASPackage -ID 45
```

Parameters

The `Remove-ASPackage` cmdlet has the following parameters:

Table 23-38 • Remove-ASPackage Parameters

Parameter	Description
ID	Specifies the package ID for the package which is being removed. (Required)

Returns

One of the following values is returned:

- **0**—Success. The delete operation completed successfully.
- **1**—Insufficient access rights. Permission to **Delete** is not available.
- **2**—Object is locked. The group contains locked virtual packages. You will need to unlock the virtual packages in order to delete the group.
- **3**—General failure. Operation did not complete successfully.
- **4**—Protected group. The deletion of protected groups is prohibited.
- **5**—Item not found. The requested item could not be deleted because it could not be found.

Resolve-ASPackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Resolve-ASPackage` cmdlet to run application compatibility fixes on a package. This only picks up issues that are fixable. This will also return the path to fix transform that was produced, so the user can start a re-import.

Example

The following is the syntax used to run application compatibility fixes on a package:

```
Resolve-ASPackage -PackageId nn -DetailedResults
```

For example:

```
Resolve-ASPackage -PackageId 45 -DetailedResults
```

Parameters

The `Resolve-ASPackage` cmdlet has the following parameters:

Table 23-39 • Resolve-ASPackage Parameters

Parameter	Description
PackageId	Specifies the package ID for the package on which fixes need to be run. (Required)
[DetailedResults]	Returns detailed results of the operation. If this parameter is not used, the `Resolve-ASPackage` cmdlet returns a summary of the operation.

Returns

Either a summary of results or detailed results are returned, depending upon whether the `-DetailedResults` parameter was used.

Set-ASAppVShortcutProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

You can use the `Set-ASAppVShortcutProperty` cmdlet to define a shortcut component to be created on the target machine after successful installation.

Examples

The following is an example of the `Set-ASAppVShortcutProperty` cmdlet:

```
Set-ASAppVShortcutProperty -PackageId 1 Name 'Skype' -Target 'Skype™5..561\Phone\Skype.exe' -Publish 1
```

Table 23-40 • Set-ASAppVShortcutProperty Parameters

Parameter	Description
PackageId	Specifies the package ID for the package. ***Note*** • *It supports the package types .appv and .sft*
Name	Use to specify the name of the shortcut. (package value)
Target	Use to specify the associated executable file of the shortcut. (package value)
Publish	Use to select or unselect the shortcut components which are to be deployed from the ConfigMgr.

Return Values

A success or failure message is returned.

Get-ASAppVShortcuts

Set-ASCatalog

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Set-ASCatalog` cmdlet to upgrade an existing Application Catalog from one version to another.

Before using the `Set-ASCatalog` cmdlet to upgrade an existing Application Catalog, you first need to identify and connect to that Application Catalog using the Set-ASConfigPlatform cmdlet with the `ConnectionString` parameter.

Examples

The following is the syntax used to upgrade an existing Application Catalog:

```
Set-ASCatalog
```

When the `Set-ASCatalog` cmdlet is executed, AdminStudio detects the version of the connected Application Catalog and upgrades it to the latest version listed in the `upgrade.xml` file.

Parameters

The Set-ASCatalog cmdlet includes the following parameters:

Table 23-41 • Set-ASCatalog Parameters

Parameter	Description
[ScriptPath]	When AdminStudio executes the Set-ASCatalog cmdlet, it uses the upgrade.xml file, which contains a list of the SQL scripts that need to be run to upgrade an Application Catalog. By default, the upgrade.xml file is installed in the Support subdirectory of the AdminStudio installation directory. If you want to upgrade an Application Catalog using an upgrade.xml file in a different location, you need to use this parameter to provide the path to that file: `Set-ASCatalog -ScriptPath C:\MyScripts`

Return Values

A success or failure message is returned.

Set-ASConfigPlatform

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use this cmdlet to set defaults for most of the parameters. These defaults will be used when a specific argument/setting is not overridden in various other PowerShell cmdlets under AdminStudio Platform.

For example, if you want to set the default virtual technology to use for all your conversions, then use:

```
Set-ASConfigPlatform -BuildAppV 1
```

Then during conversion, if you do not specify any virtual technology parameter, the above defaults will be used.

Note • *Microsoft SQL Server 2012 Native Client has reached End of Life with vendor (Microsoft) in July 2022 and is no longer supported. Therefore, the SQL Native Client 2012 has been replaced with the MS OLEDB driver and its associated components.*

Example

The following is an example of the Set-ASConfigPlatform cmdlet and its ConnectionString parameter with **SQL Server Authentication**:

```
Set-ASConfigPlatform -ConnectionString "PROVIDER=MSOLEDBSQL19;Data Source=SCHLTENG01\MSSQL_5500;
    User ID=jsmith;Password=admin8032;InitialCatalog=MKTGCAT2016;"
```

The following is an example of the Set-ASConfigPlatform cmdlet and its ConnectionString parameter with **Windows NT Authentication**:

```
Set-ASConfigPlatform -ConnectionString "PROVIDER=MSOLEDBSQL19;Data Source=SCHLTENG01\MSSQL_5500;
    Integrated Security=SSPI;InitialCatalog=MKTGCAT2016;"
```

The following is an example of the `Set-ASConfigPlatform` cmdlet and its `ConnectionString` parameter with **Azure Server Authentication**:

```
Set-ASConfigPlatform -ConnectionString "PROVIDER=MSOLEDBSQL19;Data Source=<XXXXX>;Initial
    catalog=MKTGCAT2016;User ID=sa;Password=<XXXXX>;"
```

The following is an example of the `Set-ASConfigPlatform` cmdlet and its `ConnectionString` parameter with **Azure Active Directory Authentication**:

```
Set-ASConfigPlatform -ConnectionString "PROVIDER=MSOLEDBSQL19;Data Source=<XXXXX>;Initial
    catalog=MKTGCAT2016;User ID=xxx@xx.com;Password=<XXXXX>;Authentication=ActiveDirectoryPassword;"
```

Parameters

The `Set-ASConfigPlatform` cmdlet has the following parameters:

Table 23-42 • Set-ASConfigPlatform Parameters

Parameter	Description

[ConnectionString]	Sets the default connection string for the AdminStudio Platform. A connection string consists of a set of elements, separated by semi-colons.
	• `PROVIDER`—Because AdminStudio only supports SQL Server databases, this element must always be set to MSOLEDBSQL19
	• `Data Source`—Identifies the database server.
	• `Initial Catalog`—Identifies the Application Catalog name.
	The elements used in the connection string vary depending upon the authentication method you are using.
	SQL Server Authentication
	When using this authentication method, you need to include the `User ID` and `Password` elements to provide the SQL Server login credentials:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`User ID=UserName;`
	`Password=passwd;`
	`Initial Catalog=CatalogName;`
	Windows NT Authentication
	When using this authentication method, you need to include the `Integrated Security=SSPI` element to specify that you are using Windows NT authentication:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`Initial Catalog=CatalogName;`
	`Integrated Security=SSPI;`
	Note • *The connection string is encrypted before setting in the `PlatformSettings.xml` file.*
	Azure SQL Server Authentication
	When using this authentication method, you need to include the `User ID` and `Password` elements to provide the Azure Server login credentials:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`User ID=UserName;`
	`Password=passwd;`
	`Initial Catalog=CatalogName;`

Table 23-42 • Set-ASConfigPlatform Parameters (cont.)

Parameter	Description

	Azure Active Directory Authentication
	When using this authentication method, you need to include the `User ID` and `Password` elements to provide the Azure Active Directory login credentials:
	`PROVIDER=MSOLEDBSQL19;`
	`Data Source=SCHLTENG01\MSSQL_5500;`
	`User ID=UserName;`
	`Password=passwd;`
	`Authentication=ActiveDirectoryPassword;`
	`Initial Catalog=CatalogName;`

[Group]	Use to specify the default group name where all imported packages should be placed. ***Note*** • *This setting can be overridden by the Invoke-ASImportPackage cmdlet.* ***Note*** • *This default group does not need to already exist in Application Catalog in order for the platform cmdlets to work.* Also, when specifying the `Group` parameter, you need to include the `Applications` root group in the path to the group. For example: `Set-ASConfigPlatform -BuildAppV 1 -Group "Applications\SubGroup\SubGroup1"` If you do not include the `Applications` root group in the path, the packages will be imported under the root node instead of the specified folder path.
[AACSettingsFile]	Specify the Automated Application Converter project file (.aacx) to use for all conversion tasks. This setting can be overridden by individual conversion cmdlets.
[OutputPath]	Specify the default output folder under which all virtualized packages will be stored. This setting can be overridden by individual conversion cmdlets.
[HardTimeout]	Hard time-out (in minutes) for the package installation.
[SoftTimeout]	Soft time-out (in minutes) for the package installation.
[BuildAppV]	Build Microsoft App-V packages (`*.sft`). Specify **`0`** (do not build) or **`1`** (build).
[BuildXenApp]	Build Citrix XenApp profiles (`*.profile`). Specify **`0`** (do not build) or **`1`** (build).
[BuildThinApp]	Build VMWare ThinApp packages (`*.exe`). Specify **`0`** (do not build) or **`1`** (build).
[BuildMSI]	Build Windows Installer packages (`*.msi`). Specify **`0`** (do not build) or **`1`** (build).
[AppVServerHost]	Host name portion of the server location for App-V packages.

Table 23-42 • Set-ASConfigPlatform Parameters (cont.)

Parameter	Description
[AppVServerPort]	Port number portion of the server location for App-V packages.
[AppVServerProtocol]	Protocol portion of the server location for App-V packages.

Return Values

No values are returned.

Set-ASDistributionProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Important • *Starting in AdminStudio 2021 R2, Set-ASDistributionProperty is obsolete and Set-ASProperty should used instead.*

You can use the `Set-ASDistributionProperty` cmdlet to set the property value of the distribution system for a package. The property name is obtained from the `Property_PluginId` table for the corresponding distribution system. The property value should be the display text that is displayed in the AdminStudio user interface.

Example

The following is an example of the `Set-ASDistributionProperty` cmdlet:

```
Set-ASDistributionProperty -PackageId 10 -SystemConnectionName 'Altiris' -PropertyName 'PackageName'
    -PropertyValue 'PackageABC'
```

Important • *All parameters are mandatory.*

Parameters

The `Set-ASDistributionProperty` cmdlet has the following parameters:

Table 23-43 • Set-ASDistributionProperty Parameters

Parameter	Description
PackageId	(Required) Use to specify the ID number of the package that you want to set distribution properties for. The ID number is obtained from the `cstblPackage` table.

Table 23-43 • Set-ASDistributionProperty Parameters

Parameter	Description
-SystemConnectionName	(Required) Use to specify a named connection to a deployment system. See Creating Multiple Named Connections to Distribution Systems for more information.
PropertyName	(Required) Use to specify the name of the property that you want to set. The name of the property is fetched from the `Property_PluginId` table for the corresponding distribution system.
PropertyValue	(Required) Use to specify the value of the distribution property that you want to set. For enum values, specify the display text of the drop down selection that is displayed in the AdminStudio user interface.

Return Value

One of the following values is returned:

- **True**—Property value is set.
- **False**—Property value is not set.

Set-ASProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Set-ASProperty` cmdlet to set the application model properties of a package. You can use new **-SystemConnectionName** switch to set the property value of the distribution system for a package. The property name is obtained from the Property_PluginId table for the corresponding distribution system. The property value should be the display text that is displayed in the AdminStudio user interface.

- Example
- Parameters
- Available Application Properties
- Available Deployment Type Properties

Note • *Starting in AdminStudio 2021 R2, Set-ASDistributionProperty is obsolete and Set-ASProperty should used instead.*

Example

To set application model properties using the `Set-ASProperty` cmdlet, use the following syntax:

```
Set-ASProperty -PackageID n -SystemConnectionName "Connection Name" -PropertyName "Name"-
PropertyValue "Value"
```

where:

- **Name**—Name of application model property.
- **Value**—Value of application model property.
- **n**—Package ID number.
- **Connection Name**—Name of the Distribution Connection.

For example, you would use the following code to set the `PrestagedDPSetting` property to `ManualCopy`:

Set-ASDistributionProperty -PackageId 10 -SystemConnectionName 'Altiris' -PropertyName 'PackageName' -PropertyValue 'PackageABC'

```
Set-ASProperty -PropertyName "PrestagedDPSetting" -PropertyValue "ManualCopy" -PackageID 1
```

To set multiple properties simultaneously, you should create a PowerShell script file containing multiple `Set-ASProperty` cmdlets, such as:

```
Set-ASProperty -PropertyName "PrestagedDPSetting" -PropertyValue "ManualCopy" -PackageID 1
Set-ASProperty -PropertyName "AutoInstall"        -PropertyValue "True"       -PackageID 1
Set-ASProperty -PropertyName "RunAs32"            -PropertyValue "False"      -PackageID 1
```

Parameters

The `Set-ASProperty` cmdlet includes the following parameters:

Table 23-44 • Set-ASProperty Parameters

Parameter	Description
PropertyName	Use to specify the name of the application model property that you want to set. ***Note*** *• The name of the property is fetched from the Property_PluginId table for the corresponding distribution system to set properties related to other Distribution System except ConfigMgr.*
PropertyValue	Use to specify the value of the application model property that you want to set.
SystemConnectionName	Use to specify a named connection to a deployment system. For more information, see Creating Multiple Named Connections to Distribution Systems.

Available Application Properties

The following application properties that appear on the **Application View** can also be set using the `Set-ASProperty` cmdlet:

- General Information Tab
- App Portal Information Tab

General Information Tab

The following application properties can be set using the `Set-ASProperty` cmdlet. These properties are also displayed on the **General Information** tab of the Application Catalog **Application View**.

Table 23-45 • Application View / General Information Tab Properties

Name Displayed in Application Catalog	Property Name	Possible Values
Administrator comments	`Description`	Any string value
Manufacturer	`Publisher`	Any string value
Install from Install Application task sequence	`AutoInstall`	`True` `False`

Distribution priority	`DistributionPriority`	`High`
		`Medium`
		`Low`

Distribute to preferred DP	`PreferredDistribute`	`True` `False`

Prestaged DP settings	`PrestagedDPSetting`	`Auto`
		`OnlyContentChange`
		`ManualCopy`

Display supersedes information to user	`DisplaySupersedes`	`True` `False`
Distribution point groups	`DistributionPointGroups`	`Any string value`
Localized description	`LocalizedDescription`	Any string value
User documentation	`UserDocumentation`	Any string value
Icon file	`Icon`	Name of `.ico` file

Classification	`Classification`	`Unknown`
		`Desktop`
		`Server`

Flexera Identifier	`FID`	`Flexera ID not found`
		`Not connected to Flexera Service Gateway`
		`Multiple applications detected`
		`Error while fetching Flexera ID`
		`Not synchronized with FlexNet Manager Platform`

App Portal Information Tab

The following application properties can be set using the Set-ASProperty cmdlet. These properties are also displayed on the **App Portal Information** tab of the Application Catalog **Application View**.

Table 23-46 • Application View / General Information Tab Properties

Name Displayed in Application Catalog	Property Name	Possible Values

Categories	Categories	String value of path to existing category. For example:
		`Set-ASProperty -PackageID 1 -PropertyName "Categories" -PropertyValue "HR/Office/Excel 2007"`
		`Set-ASProperty -PackageID 1 -PropertyName "Categories" -PropertyValue "HR/Office/Excel 2007,IT/Software/Excel"`
		Multiple categories can be set by separating them by a comma.
		
		Important • *Categories must already exist in App Portal. If you attempt to use* Set-ASProperty *to assign an application to a non-existent category, the cmdlet will fail.*

Keywords	Keywords	Single or comma-delimited list of keywords. For example:
		`Set-ASProperty -PackageID 1 -PropertyName "Keywords" -PropertyValue "Admin"`
		`Set-ASProperty -PackageID 1 -PropertyName "Keywords" -PropertyValue "Admin,IT,HR"`
		
		Note • *When using the* Set-ASProperty *cmdlet to assign keywords to an application, follow these rules:*
		• *Until you create a keyword using either the Add-ASKeywords cmdlet or the* ***Edit Keywords*** *dialog box, you cannot use the* Set-ASProperty *cmdlet to assign it to an application. If you attempt to do so, an error will be returned.*
		• *If you attempt to assign multiple keywords (in a comma-delimited list) to an application, if one of them has not yet been created, the cmdlet will fail and no keywords will be assigned.*
		• *If you attempt to assign a duplicate keyword to an application using the Set-ASProperty cmdlet, the cmdlet will fail.*
		• *Do not use the single-quote character (') in a keyword.*

Available Deployment Type Properties

The following deployment type properties that appear on the **Home Deployment Type View** can also be set using the Set-ASProperty cmdlet.

- Package Information Tab
- Deployment Data Tab

Package Information Tab

The following deployment type properties can be set using the `Set-ASProperty` cmdlet. These properties are also displayed on the **Package Information** tab of the Application Catalog **Home Deployment Type View**.

Table 23-47 • Home Deployment Type View / Package Information Tab Properties

Name Displayed in Application Catalog	Property Name	Possible Values
Manufacturer	`Manufacturer`	Any string value
Administrator Comments	`AdministratorComments`	Any string value
Original File	`OriginalMsiFileName`	Any string value
Original name of package	`SoftwareProductName`	Any string value
If package name was modified, the modified name is stored in this property	`DisplayedProductName`	Any string value

Deployment Data Tab

In Application Catalog, deployment type properties are displayed on the **Deployment Data** tab of the Application Catalog **Home Deployment Type View** on the following subtabs:

- Content Subtab
- Programs Subtab
- User Experience Subtab

Content Subtab

The following properties, which are displayed on the **Content** subtab of the **Deployment Type** tab, can be set using the `Set-ASProperty` cmdlet of the AdminStudio PowerShell Cmdlets.

Note • *Class can be of* `ASCMMSIContent`, `ASCMScriptContent`, *or* `ASCMAppvContent`.

Table 23-48 • Home Deployment Type View / Content Subtab

Name Displayed in Application Catalog	Property Name	Possible Values
Use fallback source location for content	`FallbackToUnprotectedDP`	`True` `False`
Content location	`Location`	`Any string value`
Deployment option when client is on fast (LAN) network ***Note*** • *App-V packages only.*	`OnFastNetwork`	`Download` `DownloadContentForStreaming`

Table 23-48 • Home Deployment Type View / Content Subtab

Name Displayed in Application Catalog	Property Name	Possible Values
Deployment option when client is on slow network	OnSlowNetwork	DoNothing Download DownloadContentForStreaming ***Note*** • *The* DownloadContentForStreaming *option only applies to App-V packages.*
Enable peer-to-peer content distribution ***Note*** • *App-V packages only.*	PeerCache	True False
Allow client to share content on same subnet ***Note*** • *MSI and EXE packages only.*	PeerCache	True False
Persist content in the client cache	PinOnClient	True False
Load content to App-V cache ***Note*** • *App-V packages only.*	RequireLoad	True False
Uninstall content settings	UnInstallContentSettings	SameAsInstall NoneRequired Different
Uninstall content location	UnInstallContentLocation	Any String Value (based on Uninstall Content settings)

Programs Subtab

The following properties, which are displayed on the **Programs** subtab of the **Deployment Type** tab, can be set using the Set-ASProperty cmdlet of the AdminStudio PowerShell Cmdlets.

Note • *Class can be of* ASCMMsiInstaller *or* ASCMScriptInstaller.

Note • *This subtab is only visible for MSI and EXE (script installer only) packages.*

Table 23-49 • Home Deployment Type View / Programs Subtab

Name Displayed in Application Catalog	Property Name	Possible Values
Install command line	`InstallCommandLine`	Any string value
Install folder	`InstallFolder`	Any string value
Uninstall command line	`UninstallCommandLine`	Any string value
Uninstall folder	`UninstallFolder`	Any string value
Run installation as 32-bit process on 64-bit client	`RunAs32`	`True` `False`
Installation source management product code	`SourceUpdateProductCode`	Any valid GUID
Repair command line	`RepairCommandLine`	`Any String Value`
Repair folder	`RepairFolder`	`Any String Value`

User Experience Subtab

The following properties, which are displayed on the **User Experience** subtab of the **Deployment Type** tab, can be set using the `Set-ASProperty` cmdlet of the AdminStudio PowerShell Cmdlets.

Note • *Class can be of* `ASCMMsiUserExperience` *or* `ASCMScriptUserExperience`.

Note • *This subtab is only visible for MSI and EXE (script installer only) packages.*

Table 23-50 • Home Deployment Type View / User Experience Subtab

Name Displayed in Application Catalog	Property Name	Possible Values

Installation behavior	`InstallBehaviour`	`User`
		`System`
		`Any`

Logon requirement	`LogonRequirement`	`True`
		`Null`
		`False`

Installation program visibility	`ProgramVisibility`	`Maximized`
		`Normal`
		`Minimized`
		`Hidden`

Table 23-50 • Home Deployment Type View / User Experience Subtab

Name Displayed in Application Catalog	Property Name	Possible Values
Enforce specific behavior	EnforceBehaviour	BasedOnExitCode NoAction ProgramReboot ForceReboot
Maximum allowed run time (min)	MaxExecuteTime	Any integer value
Estimated installation time (min)	ExecuteTime	Any integer value
Provision this application for all users on the device	ProvisionForAllUsers	True False

Return Values

One of the following values is returned:

- **True**—Property value was successfully set.
- **False**—Property value was not set.

Set-ASSoftwareRepositoryState

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Set-ASSoftwareRepositoryState` cmdlet to perform `CheckOut` and `UndoCheckOut` operations on a Software Repository-enabled Application Catalog.

Note • *The **CheckIn** operation is restricted to the user interface and is not supported through the `Set-ASSoftwareRepositoryState` API.*

Example

The following is the syntax used to check out a package from the Software Repository:

```
Set-ASSoftwareRepositoryState -PackageId nnnn -State state
```

For example:

```
Set-ASSoftwareRepositoryState -PackageId 45 -State CheckOut
```

Parameters

The `Set-ASSoftwareRepositoryState` cmdlet has the following parameters:

Table 23-51 • Set-ASSoftwareRepositoryState Parameters

Parameter	Description
PackageId	(Required) Use to specify the ID number of the package to be checked out or have its checkout canceled.
State	(Required) Use one of the following values to specify the change you want to make to the Software Repository state: • CheckOut • UndoCheckOut

Returns

When executed, one of the following values is returned:

Table 23-52 • Return Values for Set-ASSoftwareRepositoryState

Return Value	Description
0	The state change was successful.
-1	Error: Package directory was not found.
-2	Error: Package copy failure.
-3	Error: Package move failure.
-4	Error: Package add failure.
-5	Error: Package directory exists.
-10	Error: General failure.
-11	Error: Check out failure.
-12	Error: Insert into failure.
-13	Error: Delete failure.
-14	Error: Get latest version failure.
-15	Error: Mismatched connection failure.
-16	Error: Package missing failure.

Table 23-52 • Return Values for Set-ASSoftwareRepositoryState (cont.)

Return Value	Description
-17	Error: Operation canceled.

Set-ASTestState

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

You can use the `Set-ASTestState` cmdlet to set a given test to either run or not run.

Example

The following is the syntax used to specify whether or not a test will run:

```
Set-ASTestState -TestId nnnn -TestState 0
```

For example:

```
Set-ASTestState -TestId 0401 -TestState 0
```

Parameters

The ASTestState cmdlet has the following parameters:

Table 23-53 • ASTestState Parameters

Parameter	Description
TestId	Use to specify the ID number of the test whose test state you want to set. This is the same ID number that identifies the test on the Application Catalog **Select Tests to Execute** dialog box. (Required)
TestState	(Required) Use one of the following values to specify a test state: • **1** = Select test • **0** = Clear the selection of the test

Returns

One of the following values is returned:

- **True**—Operation was successful.
- **False**—Operation was not successful.

Set-ASOptionProperty

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

you can use the `Set-ASOptionProperty` to set the download path for the Package Feed Module.

Example

```
Set-ASOptionProperty -OptionType PackageFeedOptions -PropertyName PackageFeedDownloadPath -
    PropertyValue D:\AR\AdminStudio Shared\PF\[Vendor]\[Product Name]\[Version]
```

Parameters

The cmdlet has the following parameter:

Table 23-54 • Set-ASOptionProperty Parameters

Parameter	Description
PropertyName	Use to specify the below property names that you want to set: • PSTemplateLocation • PSOutputLocation • PSWrapOnImport • EXETemplateLocation • EXEOutputLocation • EXEWrapOnImport • PackageFeedDownloadPath
PropertyValue	Use to specify the property value that you want to set: For Example the below property value is used for the property name PackageFeedDownloadPath: • Folder path • Valid Place Holders

Return value

A success or failure message is returned.

See Also
Get-ASOptionProperty

Set-ASSoftwareTagProperties

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition.*

You can use the `Set-ASSoftwareTagProperties` cmdlet to set the software tag properties.

Example

The following is the syntax used to set the software tag properties:

```
Set-ASSoftwareTagProperties -PackageId 1 -RequireSoftwareEntitlement 1 -SoftwareCreatorRegId
"regid.YYYY-MM.ReversedDomainName,division."
```

Parameters

The `ASSoftwareTagProperties` cmdlet has the following parameters:

Table 23-55 • `ASSoftwareTagProperties` Parameters

Parameter	Description
RequireSoftwareEntitlement	(Required) Use one of the following values to specify a software entitlement: • **1** = Software must be purchased • **0** = Software is free
TagCreatorName	Use to enter a name to identify the creator of this tag file, the default value is Flexera LLC.
TagCreatorRegID	(Required) Use to enter a RegID to identify the creator of this tag file, using the following format: regid.YYYY-MM.ReversedDomainName,optional_division For example: regid.2009-06.com.yourcompany,GlobalProductDivision
SoftwareCreatorName	Enter a name to identify the creator of this package. By default, the value is Unknown. If the value of this field is left as Unknown, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.
SoftwareCreatorRegID	(Required) Use to enter a RegID to identify the creator of this package. By default, the value is Unknown. If the value of this field is left as Unknown, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.
SoftwareLicenseName	Use to enter a name to identify the licensor of this package. By default, the value is Unknown. If the value of this field is left as Unknown, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.

Table 23-55 • `ASSoftwareTagProperties` Parameters (cont.)

Parameter	Description
SoftwareLicensorRegID	(Required) Use to enter a RegID to identify the licensor of this package. By default, the value is Unknown. If the value of this field is left as Unknown, then that exact string will appear in the tag file to indicate that it is not possible to determine the actual value for this field.

Returns

A success or failure message is returned.

Start-ASConversion

Edition *• Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

Use the `Start-ASConversion` cmdlet to start Automatic Application Converter using a given `.aacx` file. This cmdlet is usually used after you have added one or more packages using the `Add-ASPackageForConversion` cmdlet, which returns a path to an `.aacx` file. This `.aacx` file is then passed to the `Start-ASConversion` cmdlet to start the conversion.

Examples

The following are examples of how to use the Start-ASConversion cmdlet:

```
Start-ASConversion -AACSettings "C:\Personal\AAC\test.aacx" -BuildMSI
```

```
Start-ASConversion -AACSettings "C:\Personal\AAC\test.aacx" -OutputPath "C:\Result"
-BuildAppV -BuildXenApp -BuildThinApp -BuildMSI
```

Parameters

The `Start-ASConversion` cmdlet has the following parameters:

Table 23-56 • Start-ASConversion Parameters

Parameter	Description
AACSettings	The Automated Application Converter project file to use for Conversion. Use to specify the Automated Application Converter project file to use during conversion. If it is not supplied, a copy of the project file specified in the platform settings file will be used.
[VMPlatform]	Specify platform to use for automated repackaging, such as **`600Sx64`**, **`501`**, or **`502S`**.
[OutputPath]	Output folder under which all output will be collected.
[BuildAppV]	Specify this parameter to build App-V packages.

Table 23-56 • Start-ASConversion Parameters (cont.)

Parameter	Description
[BuildXenApp]	Specify this parameter to build Citrix XenApp profiles.
[BuildThinApp]	Specify this parameter to build VMWare ThinApp packages.
[BuildMSI]	Specify this parameter to build Windows Installer packages.

Return Values

The path to an `.aacx` file is returned.

Test-ASConflicts

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Test-ASConflicts` cmdlet performs conflict analysis between a source package and target packages.

When using the `Test-ASConflicts` cmdlet:

- The package needs to exist in the Application Catalog.
- You can run the analysis against a list of other PackageIDs, or specify an existing group name to run the analysis against all packages in that group.
- If none of these targets are specified, then the group in which the source package exists will be used for analysis.

You can optionally specify a comma-separated list of rule names to run.

Examples

```
Test-ASConflicts -PackageID 21 -TargetGroups MyApplications -Rules ACE03,ACE04
```

Parameters

The `Test-ASConflicts` cmdlet has the following parameters:

Table 23-57 • Test-ASConflicts Parameters

Parameter	Description
PackageID	Use to specify the source package in the conflict analysis.
[TargetGroups]	Use to specify the group name(s) of the groups against which you want to compare the source package for conflicts.
[TargetPackageIDs]	Use to specify the Package IDs of the packages against which you want to compare the source package for conflicts.

Table 23-57 • Test-ASConflicts Parameters (cont.)

Parameter	Description
[Rules]	Use to specify the rules to evaluate during the conflict analysis.

Return Values

A list of test results is returned.

Test-ASPackage

Edition • *Powershell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

The `Test-ASPackage` cmdlet performs testing on a specified package. Using this cmdlet is equivalent to selecting the package in the Application Catalog tree of the **Analyze** tab and clicking the **Execute Tests** button. When using the AdminStudio PowerShell Cmdlets to perform testing, the tests appropriate to the package that are selected in AdminStudio Application Catalog on the **Select Tests to Execute** dialog box are executed.

Examples

```
Test-ASPackage -PackageId nnn -DetailedResults
```

Parameters

The `Test-ASPackage` cmdlet has the following parameters:

Table 23-58 • Test-ASPackage Parameters

Parameter	Description
PackageId *nnn*	Use to identify the package which needs to be tested.
[DetailedResults]	Add this parameter to return individual result data for the tests that were run. If this parameter is not included, then the cmdlet returns a summary of the test execution and just includes the number of errors and warnings encountered in the test run.

Return Values

A list of test results is returned.

New-ASPackageRequest

Edition • *PowerShell cmdlets are enabled with AdminStudio Enterprise Edition and with Workflow Manager.*

A PowerShell API to be called by an external system. When you send a request through an external system, request will automatically lands in the package backlog in the AdminStudio and will avoid manual communication for the package request.

Examples

```
New-ASPackageRequest -Product 'Firefox' -Source 'FlexraOne' -Vendor 'Mozilla' -Priority '2' -Version
'15.0'
```

Parameters

The `New-ASPackageRequest` cmdlet has the following parameters:

Table 23-59 • New-ASPackageRequest Parameters

Parameter	Description
Product	ProductName of the Package.
Source	Source of the Package.
Vendor	Manufacturer of the Package.
Version	Version of the Package.
Priority	Priority of the Package. ***Note*** • *Priority value should not exceed more than five. Other wise an error message will be shown.*

Return Values

A success or failure message is returned.

24

AdminStudio REST APIs

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

The AdminStudio REST APIs described in this document enable you to integrate AdminStudio capabilities into your application or to build your own user interface.

This chapter includes the following topics:

- About the AdminStudio REST APIs
- Setting Up Connection to Run REST APIs
- REST APIs Reference

About the AdminStudio REST APIs

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

Some of the core tasks that the AdminStudio REST APIs enables you to automate include:

- **Application Catalogs**—Creating a new Application Catalog or upgrading an existing Application Catalog.
- **Importing**—Importing existing packages into the AdminStudio Application Catalog.
  - Supports importing all the formats that are currently supported by Application Catalog.
  - Supports importing individual package into Application Catalog.
- **Application virtualization compatibility**—Checking to see if your packages for suitable for conversion to virtual formats.

- **Application model properties**—Can set the application model properties of an application.
- **Testing**—Testing your packages by running operating system compatibility, best practice validation, and conflict analysis tests on the packages, and viewing test results.
- **Publishing**—Publishing your applications to Microsoft System Center Configuration Manager.

Setting Up Connection to Run REST APIs

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

Perform the following setup connection to run REST API:

- AdminStudioHost.exe.config file

AdminStudioHost.exe.config file

AdminStudioHost.exe.config file should be updated with the ConnectionString details. AdminStudioHost.exe.config file can be found at **C:\Program Files (x86)\AdminStudio\2021\Common**. The file should be updated as follows:

- If machine is in a domain, update as below:

```
<connectionStrings>
<clear/>
<add name="WebServiceCatalog" connectionString="Data Source= SCHLTENG01\MSSQL_5500; User ID=as01test\XXXX;Initial Catalog=XXXXX;Integrated Security=SSPI;"/>
</connectionStrings>
```

- If machine is not in domain, update as below:

```
<connectionStrings>
<clear/>
<add name="WebServiceCatalog" connectionString="Data Source= SCHLTENG01\MSSQL_5500;User ID=sa;Initial Catalog=XXXXX;Password=*****;"/>
</connectionStrings>
```

REST APIs Reference

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

The below table shows the list of all AdminStudio REST APIs.

Table 24-1 • AdminStudio REST APIs

API Name	Description
Adding App Portal Catalog Item Keywords to the Application Catalog	It will add App Portal catalog item keywords to the Application Catalog.
Getting Package Deployment Type	It will give details of the deployment type of package specified.
Get-ASTestDetails	It will display the details of all application compatibility rules.
Get-ASKeywords	It will return a list of App Portal catalog keywords.
Get-ASTestState	It will return the test state (selected or not selected) of a given test.
Get-ASApplicationID	it will return the ApplicationID for a given PackageID.
Get-ASApplicationDetails	It will return all the available packages in the Application Catalog with respect to the given parameters.
Get-ASVirtualReadiness	It will obtain the virtualization readiness status of a given package.
Get-ASPackageTestSummary	It will return a summary of various tests performed for the package.
Get-ASProperty	It will return the value for a property name specified.
Get-ASCatalogItem	It will return a list of the content based on GroupID or ApplicationID specified.
Get-ASPackageCodes	It will return package codes of the specified package.
Invoke-ASPublish	It will publish a package to a deployment server.
Invoke-ASImportPackage	It will import specified package into the Application Catalog.
New-ASPackageRequest	When you send a request through an external system, request will automatically lands in the package backlog in the AdminStudio.
New-ASCatalog	It will create a new Application Catalog.
New-ASDistributionConnection	It will setup new distribution connection.
Remove-ASApplication	It will delete a application using specified ApplicationID.

Table 24-1 • AdminStudio REST APIs

API Name	Description
Remove-ASGroup	It will delete a group using specified GroupID
Remove-ASPackage	It will delete a package using specified PackageID
Resolve-ASPackage	It will run application compatibility fixes on a package
Set-ASTestState	It will set a given test to either run or not run.
Set-ASProperty	It will set the application model properties of a package.
Set-ASCatalog	It will upgrade an existing Application Catalog from one version to another.
Test-ASPackage	It will execute all the rules selected in the Test center on a specified package.

Adding App Portal Catalog Item Keywords to the Application Catalog

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will add App Portal catalog item keywords to the Application Catalog (as individual records in the ASKeywords table).

Table 24-2 • Add-ASKeywords API Information

Request Type	Description
API	`http://localhost:8086/catalog/keyword/?keyword={Keyword}`
Method	POST
Parameters	List all parameters in the following format: • **Keyword** = Use to add App Portal catalog item keywords to the Application Catalog.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-2 • Add-ASKeywords API Information

Request Type	Description
Response	Example: `http://localhost:8086/catalog/keyword/?keyword=test` <response> <HasFault>false</HasFault> <headers> <httpHeaders> <Host>localhost:8086</Host> <Connection>keep-alive</Connection> <Accept>*/*</Accept> <User-Agent>PostmanRuntime/7.26.10</User-Agent> <Accept-Encoding>gzip, deflate, br</Accept-Encoding> <Postman-Token>a6ba7704-25a4-44ed-bf55-5177e64c809c</Postman-Token> <Content-Length>0</Content-Length> </httpHeaders> <request></request> </headers> <summary /> <data>true</data> </response>

Getting Package Deployment Type

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will give details of the deployment type of package specified.

Table 24-3 • Get-DeploymentType API Information

Request Type	Description
API	`http://localhost:8086/catalog/deploymenttypes/{PackageID}`
Method	GET
Parameters	List all parameters in the following format: • **PackageID** = Specify the Package ID present in the AdminStudio catalog.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-3 • Get-DeploymentType API Information

Request Type	Description
Response	<response> <HasFault>false</ HasFault><headers><httpHeaders><Host>localhost:8086</ Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User- Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>63e000a8-f05e-445b-9666- a282cd827964</Postman-Token></httpHeaders><request></request></ headers><summary /><data><data><InSoftwareRepository>false</ InSoftwareRepository><SubscribedText>Not associated with any subscribed database</SubscribedText><ConflictResultsText>There is no conflict information persisted for this product.</ ConflictResultsText><TransformsText>This product has 1 transform(s). (C:\Users\Administrator\Desktop\Publish tosccm\TestData\msi\FoxitReader941_enu_Setup_SoftwareId.mst)</ TransformsText><InstallCommandLine>msiexec / i"FoxitReader941_enu_Setup.msi" / qnTRANSFORMS="FoxitReader941_enu_Setup_SoftwareId.mst"</ InstallCommandLine><UnInstallCommandLine>msiexec /x {13B5DA92-1D08- 11E9-ACFF-000C296BF2A5} /q</ UnInstallCommandLine><PackagePath>C:\Users\Administrator\Desktop\Publ ish to sccm\TestData\msi\FoxitReader941_enu_Setup.msi</ PackagePath><InstallationLocation></ InstallationLocation><Transforms>C:\Users\Administrator\Desktop\Publi sh tosccm\TestData\msi\FoxitReader941_enu_Setup_SoftwareId.mst</ Transforms><TransformFilePaths>C:\Users\Administrator\Desktop\Publish to sccm\TestData\msi\FoxitReader941_enu_Setup_SoftwareId.mst</ TransformFilePaths><PatchesText>This package does not have any patches.</PatchesText><ParentApplications /><ParentCount>1</ ParentCount><LanguageDisplayName>English (United States)</ LanguageDisplayName><IsSigned>false</IsSigned><GetAllInstallBehavior /><RowID>13</RowID><PackageCode>{3A149962-8850-4B5D-B144- FCBB09CC8D9A}</PackageCode><ProductCode>{13B5DA92-1D08-11E9-ACFF- 000C296BF2A5}</ProductCode> <ProductVersion>9.4.1.16828</ ProductVersion><UpgradeCode>{9D148992-FACF-4107-84A3-C48F19CF0B57}</ UpgradeCode><ProductName>Foxit Reader</ ProductName><Manufacturer>Foxit Software Inc.</ Manufacturer><ProductLanguage>1033</ ProductLanguage><FileName>C:\Users\Administrator\Desktop\Publish to sccm\TestData\msi\FoxitReader941_enu_Setup.msi</ FileName><Comments>This installer database contains the logic and data required to install Foxit Reader.</Comments><ImportDate>2021-04- 12T07:46:40.473</ImportDate><FileModifiedDate>2019-03- 14T09:16:41.11</FileModifiedDate><Flags>0</ Flags><DisplayedProductName>Foxit Reader</ DisplayedProductName><OriginalMsiFileName>FoxitReader941_enu_Setup.ms i</OriginalMsiFileName><SoftwareRepositoryFlag>0</ SoftwareRepositoryFlag><OriginalPackageLocation>C:\Users\Administrato r\Desktop\Publish to sccm\TestData\msi</OriginalPackageLocation></ data></data></response>

Get-ASTestDetails

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will display the details of all application compatibility rules.

Table 24-4 • Get-ASTestDetails API Information

Request Type	Description
API	http://localhost:8086/testcenter/tests/{InternalTestID}/details
Method	GET
Parameters	List all parameters in the following format: • **InternalTestID** = Use to specify the ID of the application compatibility or Microsoft ICE test
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-4 • Get-ASTestDetails API Information

Request Type	Description
Response	Example: http://localhost:8086/testcenter/tests/1002/state/false <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>56b65b1e-aebb-4bf3-974e-340473c067ec</Postman-Token></httpHeaders><request></request></headers><summary /><data><TestDescription>Since Windows Vista, security has been increased which causes all applications to run by default with standard user privileges (even when the logged-on user is a member of an Administrator group). As a result, unmanifested Control Panel (.cpl) files might fail. A manifest file is a simple .xml file containing settings that inform the operating system how to handle the program when it is launched.</TestDescription><TestBriefDescription>The Windows Installer database is scanned for the presence of unmanifested Control Panel (.cpl) files.</TestBriefDescription><TestSubchecks>3</TestSubchecks><ManualFixDescription>Control Panel (.cpl) files should be embedded in .exe files including a manifest that specifies the privilege level required to execute the application. Where this is not feasible, an external manifest file can be created. In the latter case, the manifest file must be co-located with the .cpl file and named the same as the full filename of the .cpl file, with .manifest extension (e.g. <application_name>.cpl.manifest).</ManualFixDescription><BasicFixDescription>For each unmanifested Control Panel (.cpl) file, a manifest file specifying privilege level "highestAvailable" is added in a Windows Installer transform.</BasicFixDescription><AdvancedFixDescription>For each unmanifested Control Panel (.cpl) file, a manifest file specifying privilege "requireAdministrator" is added in a Windows Installer transform.</AdvancedFixDescription><SeverityType>Yellow</SeverityType><MoreInfoLink>http://msdn.microsoft.com/en-us/library/Aa480152#appcomp_topic4</MoreInfoLink><HasAdvancedFix>true</HasAdvancedFix><HasBasicFix>true</HasBasicFix></data> </response>

Get-ASKeywords

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will return a comma-delimited list of App Portal catalog item keywords in the Application Catalog's ASKeywords table.

Table 24-5 • Get-ASKeywords API Information

Request Type	Description
API	http://localhost:8086/catalog/keywords
Method	GET
Parameters	Not applicable.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Response	<response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>f53ea1a2-7c55-4524-bf02-6745fda411e8</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><Keyword>test</Keyword><Oid>1</Oid></data><data><Keyword>rock</Keyword><Oid>2</Oid></data></data> </response>

Get-ASTestState

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will return the test state (selected or not selected) of a given test.

Table 24-6 • Get-ASTestState API Information

Request Type	Description
API	http://localhost:8086/testcenter/tests/{InternalTestID}/state
Method	GET
Parameters	List all parameters in the following format: • **InternalTestID** = Use to specify the ID number of the test that you are checking the test state

Table 24-6 • Get-ASTestState API Information

Request Type	Description
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Response	Example: http://localhost:8086/testcenter/tests/1002/state <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>8ffb6c26-8b8b-429b-8b64-0919f8247772</Postman-Token></httpHeaders><request></request></headers><summary /><data>true</data> </response>

Get-ASApplicationID

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will return the ApplicationID for a given PackageID.

Table 24-7 • Get-ASApplicationID API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/applicationID/{PackageID}
Method	GET
Parameters	List all parameters in the following format: • **PackageID** = Specify the PackageID of the package that you need the ApplicationID for.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-7 • Get-ASApplicationID API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/deploymenttypes/applicationID/13 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>8b3a65bc-a143-4921-aa79-083d5d15c4d1</Postman-Token></httpHeaders><request></request></headers><summary /><data>28</data> </response>

Get-ASApplicationDetails

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request with out parameters, it will return all the available packages in the Application Catalog.

Table 24-8 • Get-ASApplicationDetails (with out parameters) API Information

Request Type	Description
API	http://localhost:8086/catalog/packages/
Method	GET
Parameters	Not Applicable.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-8 • Get-ASApplicationDetails (with out parameters) API Information

Request Type	Description
Response	<response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>17353504-4185-4e23-81e7-d4df56512916</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><PackageId>3</PackageId><ProductName>Picasa</ProductName><Manufacturer>Google Inc.</Manufacturer><Version>2.7.0</Version><PackageType>MSI</PackageType><ApplicationId>10</ApplicationId><PackageCatalogPath>Applications/WFM/Picasa</PackageCatalogPath></data><data><PackageId>4</PackageId><ProductName>Postman</ProductName><Manufacturer>Postman</Manufacturer><Version>8.0.6</Version><PackageType>EXE</PackageType><ApplicationId>11</ApplicationId><PackageCatalogPath>Applications/WFM/Postman</PackageCatalogPath></data><data><PackageId>55</PackageId><ProductName>Wireshark</ProductName><Manufacturer>Wireshark development team</Manufacturer><Version>3.0.0.0</Version><PackageType>MSIX</PackageType><ApplicationId>45</ApplicationId><PackageCatalogPath>Applications/Wireshark</PackageCatalogPath></data></data> </response>

When you send this request with parameters, it will return the details of matching packages in the Application Catalog.

Table 24-9 • Get-ASApplicationDetails API Information

Request Type	Description
API	http://localhost:8086/catalog/packages/?productname={xyz}&manufacturer={xyz}&version={0.0.0}
Method	GET
Parameters	List all parameters in the following format: • **ProductName** = ProductName of the Package in the Application Catalog • **Manufacturer** = Manufacturer of the Package in the Application Catalog • **Version** = Version of the Package in the Application Catalog
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-9 • Get-ASApplicationDetails API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/packages/?ProductName=Picasa&Manufacturer=Google Inc.&Version=2.7.0 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>631d242d-9a95-4ae8-bf9e-e05328d66f3e</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><PackageId>3</PackageId><ProductName>Picasa</ProductName><Manufacturer>Google Inc.</Manufacturer><Version>2.7.0</Version><PackageType>MSI</PackageType><ApplicationId>10</ApplicationId><PackageCatalogPath>Applications/WFM/Picasa</PackageCatalogPath></data></data> </response>

Get-ASVirtualReadiness

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will obtain the virtualization readiness status of a given package.

Table 24-10 • Get-ASVirtualReadiness API Information

Request Type	Description
API	http://localhost:8086/testcenter/virtualizationreadiness/deploymenttypes/{PackageID}/details
Method	GET
Parameters	List all parameters in the following format: • **PackageID** = Specify the package ID of package that you are testing.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-10 • Get-ASVirtualReadiness API Information

Request Type	Description
Response	Example: http://localhost:8086/testcenter/virtualizationreadiness/deploymenttypes/59/details <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>996b368f-9fed-444e-a331-c4a07340fad8</Postman-Token></httpHeaders><request></request></headers><summary /><data>32768</data> </response>

Get-ASPackageTestSummary

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will return a summary of various tests performed for the package that is specified using the -PackageId parameter.

Table 24-11 • Get-ASPackageTestSummary API Information

Request Type	Description
API	http://localhost:8086/testcenter/deploymenttypes/{PackageID}/details
Method	GET
Parameters	List all parameters in the following format: • **PackageID** = ID of the package that will be returned.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-11 • Get-ASPackageTestSummary API Information

Request Type	Description
Response	Example: http://localhost:8086/testcenter/deploymenttypes/59/details <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>0bde6c12-eae3-4ca4-94ce-b48a042a99a3</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><CategoryName>Windows Server 2019</CategoryName><TotalExecuted>54</TotalExecuted><TotalErrors>0</TotalErrors><TotalWarnings>1</TotalWarnings><AutoFixable>0</AutoFixable><Suppressed>0</Suppressed> <OverallAssessment>4</OverallAssessment><CategoryId>122</CategoryId><TestType>79</TestType><IsVirtualized>0</IsVirtualized><Recommendation>0</Recommendation></data></data> </response>

Get-ASProperty

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will return the value for a property name specified.

Table 24-12 • Get-ASProperty API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/{id}/properties/{propertyname}/?distributionconnectionname={distributionconnectionname}
Method	GET

Table 24-12 • Get-ASProperty API Information

Request Type	Description
Parameters	List all parameters in the following format: • **PackageID** = ID of the package that will be returned. • **DistributionConnectionName**= Name of the Distribution Connection. • **PropertyName** = Name of the property. ***Note*** *• The name of the property is fetched from the Property_PluginId table for the corresponding distribution system to set properties related to other Distribution System except ConfigMgr.*
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Response	Example: 1. http://localhost:8086/catalog/deploymenttypes/59/properties/publisher <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>5019eff7-29fe-4fd3-84ad-bec6af893f76</Postman-Token></httpHeaders><request></request></headers><summary /><data>Microsoft</data> </response> 2. http://10.80.149.84:8086/catalog/deploymenttypes/17/properties/PackageName/?distributionconnectionname=Altiris <response> <HasFault>false</HasFault><headers><httpHeaders><Cache-Control>no-cache</Cache-Control><Accept-Encoding>gzip, deflate</Accept-Encoding><Host>10.80.149.84:8086</Host><Accept>*/*</Accept><Accept-Language>en-US,en;q=0.9</Accept-Language><Postman-Token>d7a023c4-d770-2530-ce07-14496de449fc</Postman-Token><User-Agent>Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36</User-Agent><Connection>keep-alive</Connection></httpHeaders><request></request></headers><summary /><data>TortoiseSVN1.11.28</data> </response>

Get-ASCatalogItem

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it returns a list of the content based on GroupId or ApplicationID specified.

Table 24-13 • Get-ASCatalogItem API Information

Request Type	Description
API	**GroupID**: http://localhost:8086/catalog/groups/{GroupID} **ApplicationID** http://localhost:8086/catalog/applications/{ApplicationID}
Method	GET
Parameters	List all parameters in the following format: • **GroupID** = Specify the GroupID of the application. • **ApplicationID** = Specify the ApplicationID of the application.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-13 • Get-ASCatalogItem API Information

Request Type	Description
Response	GroupID: Example:http://localhost:8086/catalog/groups/1 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>00913a77-4563-4100-ad7d-9c8778c6fbc9</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><RowID>45</RowID><GroupName>Wireshark</GroupName><Description>Application Group</Description><Comments>Launch Skype</Comments><Properties>16</Properties><CreatedDate>2021-04-15T14:29:39.8</CreatedDate><ModifiedDate>2021-04-15T14:29:39.8</ModifiedDate></data><data><RowID>48</RowID><GroupName>TestGroup</GroupName><Description></Description><Comments></Comments><Properties>0</Properties><CreatedDate>2021-04-15T07:47:24.337</CreatedDate><ModifiedDate>2021-04-15T07:47:24.337</ModifiedDate></data></data> </response> ApplicationID: Example: http://localhost:8086/catalog/applications/45 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>96cfcaf4-9ee3-41e0-a38e-6bd7c2b53e79</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><ChildGroups /><ChildApplications /><ChildApplicationsAsApplications /><GroupPackages /><RowID>55</RowID><GroupName /><Description /><Comments>Launch Skype</Comments><Properties>0</Properties><CreatedDate>0001-01-01T00:00:00</CreatedDate><ModifiedDate>0001-01-01T00:00:00</ModifiedDate></data></data> </response>

Get-ASPackageCodes

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, It will return package codes of the specified package.

Table 24-14 • Get-ASPackageCodes API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/packagecodes/{PackageID}
Method	GET
Parameters	List all parameters in the following format: • **PackageID** = Specifies the package ID
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-14 • Get-ASPackageCodes API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/deploymenttypes/packagecodes/63 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>6975952d-74ce-4da1-b62c-b32b1f50a054</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><InSoftwareRepository>false</InSoftwareRepository><SubscribedText>Not associated with any subscribed database</SubscribedText><ConflictResultsText>There is no conflict information persisted for this product.</ConflictResultsText><TransformsText>This package does not have any transforms.</TransformsText><InstallCommandLine></InstallCommandLine><UnInstallCommandLine></UnInstallCommandLine><PackagePath>C:\repack\FoxitReader941_enu_Setup\App-VPackage\Foxit Reader\Foxit Reader.appv</PackagePath><InstallationLocation></InstallationLocation><PatchesText>This package does not have any patches.</PatchesText><ParentCount>1</ParentCount><ApplicationsText>Only the current application references this product</ApplicationsText><AssociationsText>This package does not have any associations.</AssociationsText><LanguageDisplayName></LanguageDisplayName><AppVFB1Size /><AppVTotalSize /><AppVCompressed /><AppVClientVersion /><AppVVersionID /><AppVPackageId /><AppVPackageVersion /><AppVUrlProtocol /><AppVUrlHost /><AppVUrlPort /><AppVUrlPath /><AppVOS /><AppVUrlComplete>Not Available</AppVUrlComplete><AppVPackageType>1</AppVPackageType><IsSigned>false</IsSigned><RowID>63</RowID><PackageCode>{7D5BB368-F287-4423-ADE0-6C4540FD1518}</PackageCode><ProductCode>{13b5da92-1d08-11e9-acff-000c296bf2a5}</ProductCode><ProductVersion>9.4.1.16828</ProductVersion><UpgradeCode></UpgradeCode><ProductName>Foxit Reader</ProductName><Manufacturer>Foxit Software Inc.</Manufacturer><ProductLanguage>0</ProductLanguage><FileName>C:\repack\FoxitReader941_enu_Setup\App-VPackage\Foxit Reader\Foxit Reader.appv</FileName><Comments>No description entered</Comments><ImportDate>2021-04-16T16:16:23.76</ImportDate><FileModifiedDate>2021-04-16T16:16:18.193</FileModifiedDate><Flags>24</Flags><DisplayedProductName>Foxit Reader (MSI => App-V5)</DisplayedProductName><OriginalMsiFileName>Foxit Reader.appv</OriginalMsiFileName><SoftwareRepositoryFlag>0</SoftwareRepositoryFlag><OriginalPackageLocation>C:\repack\FoxitReader941_enu_Setup\App-VPackage\Foxit Reader</OriginalPackageLocation></data></data> </response>

Get Status of a Request

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will give details of the requested transaction (ReceiptID) given.

Table 24-15 • Get Status of a Request API Information

Request Type	Description
API	http://localhost:8086/message/transactions/{ReceiptID}/details
Method	GET
Parameters	List all parameters in the following format: • **ReceiptID** = Specify the receipt id from the postman responses.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-15 • Get Status of a Request API Information

Request Type	Description
Response	Example: http://localhost:8086/message/transactions/146f8997-b96c-4927-8483-12e32a31ab32/details <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>c3ae5f7c-ccbf-4ae6-b2f1-3526ec508c3f</Postman-Token></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1"><ReceiptID>146f8997-b96c-4927-8483-12e32a31ab32</ReceiptID><FileName /><GroupPath /><ParentReceipt /><PackageID>0</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>5</State><StateDescription>Error in Transaction</StateDescription><Messages><TransactionID>146f8997-b96c-4927-8483-12e32a31ab32</TransactionID><CreateDateTime>2021-04-19T01:07:18.9174237-07:00</CreateDateTime><ReadDateTime>2021-04-19T01:07:54.2949074-07:00</ReadDateTime><Text>Upgrade of catalog "XXXXX" Failed : Catalog is at the same level as the requested upgrade</Text><MsgID>0</MsgID><Type>3</Type></Messages><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-19T08:07:54.293908Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>3488</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt></data></data> </response>

Invoke-ASPublish

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will publish a package to a deployment server.

Table 24-16 • Invoke-ASPublish API Information

Request Type	Description
API	http://localhost:8086/catalog/publish/{ApplicationID}?TargetGroup={Applications\Group1}&ConnectionName={Distribution Server Name}
Method	POST
Parameters	List all parameters in the following format: • **ApplicationID** = Specify the ApplicationID of the application you are publishing. To get ApplicationID, see Get-ASApplicationID. • **TargetGroup** = Specify the target group on the deployment server that you want to publish this application to. • **ConnectionName** = Use to specify named connection to a deployment system.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-16 • Invoke-ASPublish API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/publish/28?TargetGroup=Applications\TestGroup&ConnectionName=SCCM 2010 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>1a7becba-db8b-4932-8851-401700d738ba</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1"><**ReceiptID>5738f44c-346c-4c88-8104-2223121c47b8</ReceiptID>**<FileName /><GroupPath /><ParentReceipt /><PackageID>0</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData><SystemName /><Server /><UserName /><Password /><SiteName /><ApplicationID>28</ApplicationID><ShareDirectory /><TargetGroup>Applications\TestGroup</TargetGroup><ConnectionName>SCCM 2010</ConnectionName><DependentApplication>false</DependentApplication><PluginID /><CustomData /></DeploymentData><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-16T12:37:52.4517933Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>6084</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt></data></data> </response> ***Note*** • *By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

Invoke-ASImportPackage

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, It will import specified package into the Application Catalog.

Table 24-17 • Invoke-ASImportPackage API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/Groups/{GroupName}/?FilePath=C:\Testdata\orca\New folder\orca.msi
Method	POST
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Parameters	List all packages in the catalog in the following format: • **FilePath** = Path of the Package that is imported into the Application Catalog. • **GroupName** = Name of the Group under which packages to be imported.

Table 24-17 • Invoke-ASImportPackage API Information

Request Type	Description
Response	<response> <HasFault>false</HasFault> <headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding> <Postman-Token>26f2941c-0af0-4f84-a31c-356c2c60ece1</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /> <data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1">**<ReceiptID>2aefaa25-76e9-4831-897a-3cd5a4ee4789</ReceiptID>**<FileName>C:\Testdata\orca\New folder\orca.msi</FileName><GroupPath>TestGroup</GroupPath><ParentReceipt /><PackageID>0</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /> <DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /> <NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-15T14:41:53.6258261Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>1624</value></Handle><SafeWaitHandle> <IsInvalid>false</IsInvalid></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>true</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt></data></data></response></response> ***Note*** • *By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

Invoke-ASAppVUpgrade

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, It will import specified package into the Application Catalog.

Table 24-18 • Invoke-ASAppVUpgrade API Information

Request Type	Description
API	
Method	POST
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Parameters	List all packages in the catalog in the following format: • **FilePath** = Path of the Package that is imported into the Application Catalog. • **GroupName** = Name of the Group under which packages to be imported.
Response	

New-ASPackageRequest

A REST API to be called by an external system. When you send a request through an external system, request will automatically lands in the package backlog in the AdminStudio and will avoid manual communication for the package request. The package details will be inserted in the Backlog tab of the AdminStudio.

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

You can use the New-ASPackageRequest API to add Package Request catalog item keywords to the Application Catalog.

This API provide details related to Package request.

Table 24-19 • New-ASPackageRequest API Information

Request Type	Description
API	http://localhost:8086/catalog/packagerequest/ ?{Product=XYZ}&{Vendor=XYZ}&{Version=0.0.0}&{Priority=n}&{Source=XYZ}
Method	POST
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-19 • New-ASPackageRequest API Information

Request Type	Description
Parameters	List all packages in the catalog in the following format: • **Product** = ProductName of the Package. It is Mandatory parameter. • **Source** = Source of the Package • **Vendor** = Manufacturer of the Package • **Version** = Version of the Package • **Priority** = Priority of the Package ***Note*** • *Priority value should not exceed more than 5. If the value of Priority is empty or invalid, then the default value will be used as a Priority. The default value of the Priority will be 3.*

Table 24-19 • New-ASPackageRequest API Information

Request Type	Description
Response	<response> <HasFault>false</HasFault> <headers> <httpHeaders> <Content-Length>0</Content-Length> <Accept-Encoding>gzip, deflate, br</Accept-Encoding> <Host>10.80.149.30:8086</Host> <Accept>*/*</Accept> <ConnectionInfo>"PROVIDER=SQLNCLI11;Data Source=10.00.000.00;Initial catalog=AS_2Feb;user ID=xx;password=xxxxxx;IntegratedSecurity=SSPI"<ConnectionInfo> <Postman-Token>027e8317-56cd-4705-84b7-bb9556cbe143</Postman-Token> <User-Agent>PostmanRuntime/7.26.8</User-Agent> <Connection>keep-alive</Connection> </httpHeaders> <request></request> </headers> <summary /> <data> <data> <Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1">**<ReceiptID>be2c718b-91d7-413c-b2a8-4a45ebac96bb</ReceiptID>** <FileName /> <GroupPath /> <ParentReceipt /> <PackageID>0</PackageID> <TestIDs /> <PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID> <PackageFeedId>0</PackageFeedId> <FixTransformPath /> <DeploymentData /> <WrapData /> <UpdatePackageData /> <NonPluginTestData /> <PackageConversionData /> <PackageFeedRequestData /> <PackageFeedAutomationRequestData /> <BacklogItemRequestData> <Product>Microsoft ODBC Driver</Product> <Vendor>Microsoft</Vendor> <Version>17.6.1.1</Version> <Source>FlexeraOne</Source> <Priority>4</Priority> </BacklogItemRequestData> <State>0</State> <StateDescription>New Transaction</StateDescription> <CustomData /> <Progress> <TransactionID>00000000-0000-0000-0000-000000000000</TransactionID> <Start>0</Start> <End>0</End> <Current>0</Current> <StepSize>0</StepSize> </Progress> <LastMessageID>-1</LastMessageID> <Release>false</Release> <TimeCreated>2021-02-05T06:33:11.4756398Z</TimeCreated> <CancelTokenSource> <IsCancellationRequested>false</IsCancellationRequested> <Token> <IsCancellationRequested>false</IsCancellationRequested> <CanBeCanceled>true</CanBeCanceled> <WaitHandle> <Handle> <value>624</value> </Handle> <SafeWaitHandle> <IsInvalid>false</IsInvalid> <IsClosed>false</IsClosed> </SafeWaitHandle> </WaitHandle> </Token> </CancelTokenSource> <ExistingPackageId>0</ExistingPackageId> <EnforceDuplicateDetection>false</EnforceDuplicateDetection> <GenerateSoftwareTag>false</GenerateSoftwareTag> </Receipt> <ClientID> <UniqueID>00000000-0000-0000-0000-000000000000</UniqueID> <Database>XXXX</Database> <Server>10.00.000.00</Server> <User>sa</User> </ClientID> </data> </data> </response> ***Note*** *• By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

New-ASCatalog

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will create a new Application Catalog.

Table 24-20 • New-ASCatalog API Information

Request Type	Description
API	http://localhost:8086/catalog/{Name}
Method	POST
Parameters	List all parameters in the following format: • **CatalogName** = Use to enter a name for the new Application Catalog. Upon successful creation, you will be automatically connected to the new Application Catalog.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-20 • New-ASCatalog API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/NewCat <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>0cf346be-7f5f-418c-a19b-df0336cdbad2</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1"><**ReceiptID>0164bf63-b8e9-4a78-8f90-6fd3a630061a</ReceiptID>**<FileName /><GroupPath /><ParentReceipt /><PackageID>0</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-19T06:41:48.3666699Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>7848</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt></data></data> </response> ***Note*** • *By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

New-ASDistributionConnection

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will setup new distribution connection.

Table 24-21 • New-ASDistributionConnection API Information

Request Type	Description

API	http://localhost:8086/catalog/distributionconnections
	{
	"Name" : "ConfigMgr",
	"PluginID" : "13",
	"ServerAddress" : "00.00.000.000",
	"SiteCode" : "XXX",
	"DistributionWindowsAuthentication" : "false",
	"DistributionUser" : "XX\\XXXXX",
	"DistributionPassword" : "XXXXX",
	"SharePath" : "\\\\00.00.000.000\\Publish",
	"ShareWindowsAuthentication" : "false",
	"ShareUserName" : "XX\\XXXXX",
	"SharePassword" : "XXXXX",
	}

Method	POST

Table 24-21 • New-ASDistributionConnection API Information

Request Type	Description

Parameters	List all parameters in the following format:
	• **Name** = Use to specify the name of this new named connection to a distribution system.
	• **PluginID** = Use to specify the plug-in with which the distribution system is associated. This parameter is mapped to the object identifier (OID) of the ASCMSupportedPackageTypes database table; you can provide any value available in this table to identify the plug-in.
	• **ServerAddress** = Use to specify the distribution server address.
	• **SiteCode** = When connecting to a ConfigMgr (Formerly called as System Center Configuration Manager) distribution system, use to specify the site code.
	• **DistributionWindowsAuthentication** = Use to specify whether the distribution connection should use Windows Authentication or a custom user name and password. Available options are.
	  ◦ False = Do not use Windows Authentication
	  ◦ True = Use Windows Authentication
	• **DistributionUser** = If not using Windows Authentication, use this parameter to specify the user name to use when connecting to the distribution system.
	• **DistributionPassword** = If not using Windows Authentication, use this parameter to specify the password to use when connecting to the distribution system.
	• **SharePath** = Use to specify the path to which packages are published.
	• **ShareWindowsAuthentication** = Use to specify whether the share path connection should use Windows Authentication or a custom user name and password. Available options are.
	  ◦ False = Do not use Windows Authentication
	  ◦ True = Use Windows Authentication
	• **ShareUserName** = If not using Windows Authentication, use this parameter to specify the user name to use when connecting to the share path.
	• **SharePassword** = If not using Windows Authentication, use this parameter to specify the password to use when connecting to the share path.

Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-21 • New-ASDistributionConnection API Information

Request Type	Description
Response	<response> <HasFault>false</HasFault><headers><httpHeaders><Content-Length>483</Content-Length><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Host>localhost:8086</Host><Content-Type>application/json</Content-Type><Accept>*/*</Accept><Postman-Token>bf0dee20-81f0-4ba2-99ab-a623e1d71fc5</Postman-Token><User-Agent>PostmanRuntime/7.26.10</User-Agent><Connection>keep-alive</Connection></httpHeaders><request></request></headers><summary /><data>true</data> </response>

Remove-ASApplication

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will delete a package using its ApplicationID.

Table 24-22 • Remove-ASApplication API Information

Request Type	Description
API	http://localhost:8086/catalog/applications/{ApplicationID}
Method	DELETE
Parameters	List all parameters in the following format: • **ApplicationID** = Specifies the ID for the application which is being removed.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-22 • Remove-ASApplication API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/applications/11 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>e2354cf1-ac57-4d16-849c-9719ae52014f</Postman-Token></httpHeaders><request></request></headers><summary /><data>1</data> </response>

Remove-ASGroup

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will delete a group using its GroupID.

Table 24-23 • Remove-ASGroup API Information

Request Type	Description
API	http://localhost:8086/catalog/groups/{GroupID}
Method	DELETE
Parameters	List all parameters in the following format: • **GroupID** = Specifies the Group ID for the group which is being removed.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-23 • Remove-ASGroup API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/groups/12 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>97c3cb01-ebe7-49a3-b5ed-6c695de6050d</Postman-Token></httpHeaders><request></request></headers><summary /><data>1</data> </response>

Remove-ASPackage

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will delete a package using its PackageID.

Table 24-24 • Remove-ASPackage API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/{PackageID}
Method	DELETE
Parameters	List all parameters in the following format: • **PackageID** = Specifies the package ID for the package which is being removed.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-24 • Remove-ASPackage API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/deploymenttypes/60 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>c0b54198-4410-4ff6-8349-3fae723586d1</Postman-Token></httpHeaders><request></request></headers><summary /><data>1</data> </response>

Resolve-ASPackage

Edition *• REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will run application compatibility fixes on a package. This only picks up issues that are fixable.

Table 24-25 • Resolve-ASPackage API Information

Request Type	Description
API	http://localhost:8086/testcenter/deploymenttypes/{PackageID}
Method	PUT
Parameters	List all parameters in the following format: • **PackageID** = Specifies the package ID for the package on which fixes need to be run.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-25 • Resolve-ASPackage API Information

Request Type	Description
Response	Example: http://localhost:8086/testcenter/deploymenttypes/59 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>32a6bf78-6b52-4258-846a-747b8be39d23</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1">**<ReceiptID>bfccb1df-e6a5-43cb-9ac6-a73f91b44d93</ReceiptID>**<FileName /><GroupPath /><ParentReceipt /><PackageID>59</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-16T07:35:40.5512132Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>4444</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt><ClientID><UniqueID>00000000-0000-0000-0000-000000000000</UniqueID><Database>XXXX</Database><Server>10.00.000.000</Server><User>sa</User></ClientID></data></data> </response> ***Note*** *• By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

Set-ASTestState

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will set a given test to either run or not run.

Table 24-26 • Set-ASTestStateAPI Information

Request Type	Description
API	http://localhost:8086/testcenter/tests/{InternalTestID}/state/{State}
Method	PUT
Parameters	List all parameters in the following format: • **InternalTestID** = Use to specify the ID number of the test whose test state you want to set • **State** = Use to specify the test state either True or False.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs
Response	Example: http://localhost:8086/testcenter/tests/1002/state/false <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>6839882d-d21e-4934-90f9-ca33a8489447</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data>true</data> </response>

Set-ASProperty

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will set the application model properties of a package.

Table 24-27 • Set-ASProperty API Information

Request Type	Description
API	http://localhost:8086/catalog/deploymenttypes/{id}/properties/{propertyname}/?value={value}&distributionconnectionname={distributionconnectionname}
Method	POST
Parameters	List all parameters in the following format: • **PackageID** = Package ID number. • **DistributionConnectionName** = Name of the Distribution Connection. • **Value** = Value of application model property. • **PropertyName** = Name of application model property. ***Note*** • *The name of the property is fetched from the Property_PluginId table for the corresponding distribution system to set properties related to other Distribution System except ConfigMgr.*
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-27 • Set-ASProperty API Information

Request Type	Description
Response	Example: 1. http://localhost:8086/catalog/deploymenttypes/59/properties/Publisher/?value=TestPublisher <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>16058260-d59e-45d4-bf96-eb59098952ec</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data>true</data> </response> 2.http://localhost:8086/catalog/deploymenttypes/17/properties/PackageName/?value=Tortoise&distributionconnectionname=Altiris <response> <HasFault>false</HasFault><headers><httpHeaders><Host>10.80.149.84:8086</Host><Accept-Language>en-US,en;q=0.9</Accept-Language><Connection>keep-alive</Connection><Origin>chrome-extension://fhbjgbiflinjbdggehcddcbncdddomop</Origin><Postman-Token>bfcbfb4c-0c84-f634-042c-5cb35cbc53e8</Postman-Token><Accept-Encoding>gzip, deflate</Accept-Encoding><Accept>*/*</Accept><User-Agent>Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36</User-Agent><Content-Length>0</Content-Length><Cache-Control>no-cache</Cache-Control></httpHeaders><request></request></headers><summary /><data>true</data> </response>

Test-ASPackage

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will execute all the rules selected in the Test center on a specified package.

Table 24-28 • Test-ASPackage API Information

Request Type	Description
API	http://localhost:8086/testcenter/deploymenttypes/{PackageID}
Method	POST
Parameters	List all parameters in the following format: • **PackageID** = Use to identify the package which needs to be tested
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-28 • Test-ASPackage API Information

Request Type	Description
Response	Example: http://localhost:8086/testcenter/deploymenttypes/47 <response> <HasFault>false</HasFault><headers><httpHeaders><Content-Length>0</Content-Length><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Referer>http://localhost:8086/testcenter/deploymenttypes/47</Referer><Host>localhost:8086</Host><Accept>*/*</Accept><Postman-Token>4153cbf3-bfd2-4f58-af93-10744b41ce28</Postman-Token><User-Agent>PostmanRuntime/7.26.10</User-Agent><Connection>keep-alive</Connection></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1"><**ReceiptID>71b866b7-ca14-4a6d-aae7-66254f79c127</ReceiptID>**<FileName /><GroupPath /><ParentReceipt /><PackageID>47</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-16T06:40:36.8576114Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>5388</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt> <ClientID><UniqueID>00000000-0000-0000-0000-000000000000</UniqueID><Database>XXXX</Database><Server>10.00.000.000</Server><User>sa</User></ClientID></data></data> </response> ***Note*** *• By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

Set-ASCatalog

Edition • *REST APIs are enabled with AdminStudio Enterprise Edition.*

When you send this request, it will upgrade Application Catalog from the previous releases.

Table 24-29 • UpgradeCatalog API Information

Request Type	Description
API	http://localhost:8086/catalog/{CatalogName}
Method	PUT
Parameters	List all parameters in the following format: • **CatalogName** = Use to upgrade a name for the new Application Catalog.
Connection Info	Provides the connection information of the package as mentioned in Setting Up Connection to Run REST APIs

Table 24-29 • UpgradeCatalog API Information

Request Type	Description
Response	Example: http://localhost:8086/catalog/AS_2020 <response> <HasFault>false</HasFault><headers><httpHeaders><Host>localhost:8086</Host><Connection>keep-alive</Connection><Accept>*/*</Accept><User-Agent>PostmanRuntime/7.26.10</User-Agent><Accept-Encoding>gzip, deflate, br</Accept-Encoding><Postman-Token>c058071e-872b-46b1-a664-1ad0d4f3f890</Postman-Token><Content-Length>0</Content-Length></httpHeaders><request></request></headers><summary /><data><data><Receipt xmlns:json="http://james.newtonking.com/projects/json" json:id="1"><**ReceiptID>2ac8f04a-ca43-40fc-a3d5-bef01e9592d9</ReceiptID>**<FileName /><GroupPath /><ParentReceipt /><PackageID>0</PackageID><TestIDs /><PluginGUID>00000000-0000-0000-0000-000000000000</PluginGUID><PackageFeedId>0</PackageFeedId><FixTransformPath /><DeploymentData /><WrapData /><AppAttachData /><UpdatePackageData /><NonPluginTestData /><PackageConversionData /><PackageFeedRequestData /><PackageFeedAutomationRequestData /><BacklogItemRequestData /><State>0</State><StateDescription>New Transaction</StateDescription><CustomData /><Progress><TransactionID>00000000-0000-0000-0000-000000000000</TransactionID><Start>0</Start><End>0</End><Current>0</Current><StepSize>0</StepSize></Progress><LastMessageID>-1</LastMessageID><Release>false</Release><TimeCreated>2021-04-19T08:12:44.5418173Z</TimeCreated><CancelTokenSource><IsCancellationRequested>false</IsCancellationRequested><Token><IsCancellationRequested>false</IsCancellationRequested><CanBeCanceled>true</CanBeCanceled><WaitHandle><Handle><value>8012</value></Handle><SafeWaitHandle><IsInvalid>false</IsInvalid><IsClosed>false</IsClosed></SafeWaitHandle></WaitHandle></Token></CancelTokenSource><ExistingPackageId>0</ExistingPackageId><EnforceDuplicateDetection>false</EnforceDuplicateDetection><GenerateSoftwareTag>false</GenerateSoftwareTag></Receipt></data></data> </response> ***Note*** *• By using ReceiptID, you can see the transaction messages for the request. For more details, see Get Status of a Request.*

Index

Symbols

-? 922
.aacx files 1079
.aot 934
  converting to Repackager project 934
  difference from .axt file 934
.axt 934
  converting to Repackager project 934
  difference from .aot file 934
.cab 1705
.cer file 2157
.cub 1696, 1699
.inc 922, 933, 1027, 1033
  converting to Repackager project 933
.ini Files Tab 1022
.ipf 934
  converting to Repackager project 934
.irp 932, 1033
  creating 932
.isl 937
  converting to Repackager project 937
.ism 131, 816, 922, 938, 1027
  building in Repackager 938
.msi 938
  building in Repackager 938
.osc 527
.pvk file 2161
.spc file 2160
.spy 1033
.txt 937
  converting to Repackager project 937
.wse 937
  converting to Repackager project 937
<Machines> 1081
<Options> 1081
<PackageList> 1081
<Packages> 1081
<Results> 1081

Numerics

0x800A151 222
-10000 - Process Cancelled By User 1321
-10001 - Suite File Missing 1321
-10002 - Suite File is Duplicate 1321
-10003 - Application File Missing 1322
-10004 - INI File Missing 1322
11000 - Excluding TCPIP Registry Entries 1322
11001 - Fail on VMware 1323
11003 - Control Panel Applet - Citrix 1323
11004 - Control Panel Applet - ThinApp 1323
11005 - QuickTime 7.4.1 Causes Fatal Error 1324
11006 - Adobe Distiller Exclude AdobePDFSettings 1324
11007 - Exclude URL Shortcut 1324
64-bit applications 855
-9000 - Unknown Exception 1268
-9001 - Unknown COM 1269
-9002 - Error Opening Package 1269
-9003 - Error Saving Package 1269
-9004 - Process Cancelled By User 1270
-9005 - Error Creating Temporary Folder 1270
-9006 - Error Decompressing Package 1271
-9007 - File With Extension Not Found 1271
-9008 - Error Extracting Icon 1272
-9009 - Unknown Provider 1272
-9010 - Invalid Target File Name 1272
-9011 - Error Reading MSI Table 1273
-9012 - Unexpected Error in Method 1273
-9013 - Type Library Not Found 1274

-9014 - ShellExecute Failed 1274
-9015 - Unable to Determine Full Path for Driver 1274
-9016 - Contents of Table Ignored 1275
-9017 - .NET 1.x Assembly Not Supported 1276
-9018 - Custom Actions Warning 1276
-9019 - Conditionalized Components 1277
-9020 - Directory With Null Parent Error 1278
-9021 - Unable to Extract COM Data 1278
-9022 - Complus Table Error 1279
-9024 - FileSFPCatalog 1279
-9026 - LaunchCondition Table Warning 1279
-9027 - LockPermissions Table Warning 1280
-9028 - MoveFile Table Error 1281
-9029 - MsiDriverPackages Table Error 1281
-9030 - ODBCTranslator Table Warning 1282
-9031 - RemoveFile Table Warning 1282
-9032 - RemoveIniFile Table Warning 1283
-9033 - RemoveRegistry Table Warning 1283
-9036 - ISCEInstall Table Error 1284
-9037 - ISComPlusApplication Table Error 1284
-9038 - ISPalmApp Table Error 1285
-9039 - ISSQLScriptFile Table Error 1285
-9040 - ISVRoot Table Error 1286
-9041 - ISXmlFile Table Error 1286
-9051 - Package Decompression Canceled 1287
-9100 - CreateInstance of Package Object Failed 1287
-9101 - Create Operation of Package Object Failed 1287
-9102 - Failed to Write Header Information 1288
-9103 - Citrix Finalization Failed 1288
-9104 - Citrix Save Failed 1289
-9105 - Error Initializing Citrix Writer 1289
-9106 - Error Initializing Citrix Package 1289
-9107 - Error Writing Citrix File Entries 1290
-9108 - Error Determining Source File Path 1290
-9109 - Error Writing Citrix Folder Entries 1290
-9110 - Error Writing Citrix Registry Entries 1291
-9113 - Error Writing Citrix INI File Entries 1291
-9114 - Error Writing Citrix Shortcuts 1292
-9115 - Error Saving Citrix Profile 1292
-9116 - Error Creating Empty Citrix Profile 1292
-9117 - Error Creating Intermediate Folder 1293
-9118 - Error Initializing Citrix Profile 1293
-9119 - Error Creating Default Target in Citrix Profile 1293
-9120 - Error Deleting File From Profile 1294
-9121 - Failed to Copy File into Citrix Profile 1294
-9122 - Target Does Not Exist in Citrix Profile 1294
-9124 - No Shortcuts Created for this Profile 1295
-9125 - Error Writing Citrix File Type Associations 1295
-9126 - Failed to Sign Profile Using Certificate 1296
-9127 - Could Not Create Script Execution 1296
-9128 - Duplicate Shortcut 1296
-9129 - Duplicate Shortcut Names 1297
-9130 - Duplicate Shortcut Targets 1297
-9131 - Unable to Resolve Installer Variable 1298
-9132 - 16 Color Shortcut Icon Not Found 1298
-9133 - Shortcut Icon Not Found 1298
-9134 - Failure to Extract Icon from Executable 1299
-9135 - Shortcut Target is 16-Bit 1299
-9136 - Some Files May Not Be Decompressed 1300
-9137 - Destination Directory Cannot Be Found 1300
-9138 - DuplicateFile table warning 1301
-9139 - 64-bit executables 1301
-9200 - ThinApp Must Be Installed 1302
-9201 - Extension for Shortcut Files Must Be .exe 1302
-9202 - No Applications Were Created 1303
-9203 - ThinApp Tool is Missing 1303
-9204 - Duplicate Shortcut 1303
-9205 - Identically-Named Shortcut Already Exists, But With Different Command Line Parameters 1304
-9206 - Identically-Named Shortcut Already Exists But With a Different Target 1304
-9207 - Error During Build Process (vregtool.exe) 1304
-9208 - Error Occurred During Build Process (vftool.exe) 1305
-9209 - Error Occurred During Build Process (tlink.exe) 1305
-9210 - 64-Bit Executables (ThinApp) 1306
-9300 - Unhandled Exception During AdviseFile Operation 1306
-9301 - Unhandled Exception During AdviseRegistry Operation 1306
-9302 - Unhandled Exception During Command Action 1307
-9303 - Unhandled Exception During Alter File Action 1307
-9304 - Unhandled Exception During Alter Registry Action 1307
-9305 - Unhandled Exception During Create Action 1308
-9306 - Unhandled Exception During Execution of Rules Engine 1308
-9401 - Error Initializing App-V Writer 1308
-9402 - Error Initializing App-V Package 1309
-9403 - Error Writing App-V File Entries 1309
-9404 - Error Writing App-V Folder Entries 1309
-9405 - Error Writing App-V Registry Entries 1310
-9406 - Error Writing App-V INI File Entries 1310
-9407 - Error Writing App-V Shortcuts 1310
-9408 - Error Writing App-V File Type Data 1311
-9409 - Error Saving App-V Data 1311
-9410 - Error Determining Source File Path 1311
-9411 - OSD File Template Could Not Be Extracted 1312
-9412 - OSD File Could Not Be Saved 1312
-9413 - App-V OSD Real Save 1312
-9414 - Local App-V Application Should Not Be Specified as a Dependency of the Primary Application 1313
-9415 - Dependency Application Was Not Found 1313
-9416 - Invalid Primary Application Directory 1313
-9417 - Dependency Application's OSD File Contains an Invalid HREF Value 1314
-9418 - Error While Privatizing Side-By-Side Assemblies 1314
-9419 - Error Inserting Watermark 1315
-9420 - Errror During App-V Package Upgrade 1315
-9421 - 16-Bit Application 1315
-9422 - Package Cannot Be Opened 1316

-9423 - No Shortcuts Detected 1316
-9424 - Windows 8 or Windows 2012 OS Required 1317
-9500 - Shortcut Missing 1317
-9600 - Error Initializing Symantec Writer 1318
-9601 - Error Writing Symantec Folder Entries 1318
-9602 - Error Writing Symantec Shortcuts 1319
-9603 - Error Creating Target File for Symantec Package 1319
-9604 - Error Writing Symantec File Entries 1319
-9605 - Error Writing Symantec Registry Entries 1320
-9606 - Error Writing Symantec INI File Entries 1320

A

aacx.exe 1233
Abort Result setting 1484
About AdminStudio dialog 141
About Application Manager 1854
About Exclusions Editor dialog box 1026
About QualityMonitor dialog 2206
About Repackager dialog box 988
Account 173
Account Administration page 173
Account Details page 174
accounts 176
  Account Administration page 173
  creating 155
  deleting 159
  deleting a group account 159
  deleting a user account 159
  disabling an account 159
  filtering by status 155
  importing 156
  importing from Active Directory 176
  managing 153
  updating 158
  viewing 174
ACE Rule Properties dialog 1855, 1856, 1857, 1858
  Additional Information tab 1856
  DLL Information tab 1858
  General Information tab 1855
  Where Clause tab 1857
ACE Tests tab 704
  in Application Manager Options dialog box 704
ACE200 1448
ACE201 1448
ACE202 1449
ACE203 1449
ACE204 1449
ACE205 1450
ACE206 1450
ACE207 1451
ACE208 1451
ACE215 1452
ACE216 1452
ACE217 1452
ACE218 1452
ACE219 1452
ACE220 1453
ACEs 1990
  creating user-defined ACEs 2124
  custom source and target packages ACEs 2123
  custom source-only package ACEs 2123
  deleting user-defined ACEs 2132
  editing user-defined ACEs 2131
  how to create custom ACEs 2124
  location of ACE files 2134
  types of user-defined ACEs 2123
  user-provided DLL based ACEs 2124
  viewing ACE metrics 2132
ACT Summary tab 1813
activation 77
  errors 103
  for AdminStudio 83
  obtaining activation code for 89
  overview 85
  ports used in 77
  returning a license for InstallShield 100
  setting up a FlexNet license server 101
  silent 77
  site accessed 105
  through a Web page 90
  through the Internet 90
  troubleshooting 102
Active Directory
  controlling access to ThinApp applications 1615
activities 2318
Activity Report 2318
  activities displayed in 2318
  creating 2318
Activity report
  creating 2316
Add AppLink Reference dialog box 1638, 1667
  entering relative paths 1668
Add Ignore Table dialog 1858
Add New Tool dialog 142
Add Packages 1167
Add Tool Wizard 143, 150, 151
  Command-Line Configurations panel 151
  Command-Line Properties dialog 143
  Tool Properties panel 150
  Welcome panel 150
Add/Remove Program settings 1725
  changing 1725
Add/Remove Programs 1726
  disabling Modify button in 1726
  disabling Remove button in 1726
  disabling Repair button in 1726
Add/Remove Programs view 1725, 1762

changing properties in 1725
Add-ASKeywords 2368
Add-ASPackageForConversion 2369
Additional Errors setting 1487
Additional Information panel 1874, 1875
Rules Wizard 1874, 1875
Additional Information Tab 1856
additional server locations 1724
configuring 1724
Additional Setup Programs dialog box 870, 918
Repackaging Wizard 870, 918
administration
copying roles 199
creating directory service connections 160
creating new accounts 155
creating new roles 198
deleting accounts 159
deleting directory service attributes 165
deleting directory service connections 162
deleting roles 200
importing accounts 156
managing directory service attributes 163
managing directory service connections 160
managing directory services 160
reviewing accounts 174
reviewing directory service connections 179
setting up directory service attributes 163
updating accounts 158
updating directory service connections 162
updating roles 198
Administrative Install panel 2260
Distribution Wizard (Package) 2260
administrative installation 2241
creating with Distribution Wizard (Package) 2241
administrator company 154
AdminStudio 119, 120, 122, 123, 134, 135, 138, 139, 150
account permissions 248
activation 77
Add Tool Wizard 150
adding tools 150
checking for updates 123
client tools permissions 193
configuring the interface 120
creating new Application Catalog 224
Customer Experience Improvement Program (CEIP) 145
database server permissions 850
deactivating 78
default Application Catalog 230
disconnecting from Application Catalog 261
installing on a different machine 78
Interface reference 135
launching applications 120
menus 139
minimum permissions 850
minimum privileges 850
Process Assistant 854
Projects tab 138
setting shared location 120
setting task page help location 122
specifying default Application Catalog 230
specifying required databsae permissions to user or group 214
specifying the location of the AdminStudio Shared Directory 120
Start Page 136
Tool Properties dialog 147
toolbar 139
troubleshooting App Portal connection 409
unable to create App Portal catalog item 409
upgrading legacy databases 229
using 1691
using the Interface 119
virtualization 1041
AdminStudio Application Catalog 224, 230, 261, 1069, 1103, 1195
creating new 224
disconnecting from 261
specifying default 230
AdminStudio Automated Application Converter Log Report 1183
viewing 1136
AdminStudio Automated Application Converter Log report 1079
viewing 1184
viewing debug messages 1185
AdminStudio databases 229
upgrading 229
AdminStudio Enterprise Server
login troubleshooting 222
standalone Application Catalogs 213
AdminStudio Interface 120, 123, 135, 138, 141, 150
configuring the interface 120
configuring to stay on top 123
dialogs 141
reference 135
using 119
wizards 150
Workflows Templates tab 138
AdminStudio options 144
AdminStudio Platform API
snapin 2360
AdminStudio Shared Directory 120, 144
Global Exclusions list 120
AdminStudio Snapin 2360
AdminStudio Test Configuration Wizard 1782, 1861
Compliance Level Panel 1861
OS Snapshot(s) Panel 1863
AdminStudio user account 249
permissions on App Portal 249
permissions on FlexNet Manager Platform 249

permissions on SQL Server 249
permissions on System Center 2012 Configuration Manager 249
AdminStudio Workflow Manager. See Workflow Manager
advanced conversion options 946
configuring Repackager 946
Advanced Options dialog 2166, 2167
Digital Signature tab 2167
Manifest Options tab 2166
Advanced Options view 946
Advanced Settings view 1015
advertisement 1703
feature 1703
AirWatch Deployment Data tab 454, 609
AirWatch Server 454
AirWatch Deployment Data tab 609
distributing applications to 2233
managing package deployment data 454
support for 233
All Merge Modules view 634
All Reports page 2327
Allow application execution to the following user groups 1615
Allow Local Interaction setting 1474
Allow user to view and interact with program installation 580
ALLUSERS 1027
alternate-language repackaging on clean machines 860
Altiris
Altiris Integration Panel 2263
XML template 2265
Altiris Deployment Data tab 451, 607
specifying command line settings 608
specifying package information 607
Altiris Server 451
Altiris Deployment Data tab 607
managing package deployment data 451
specifying package deployment settings 452
AMS. See Workflow Manager
analysis options 528
configuring in OS Snapshot Wizard 528
Analysis Options dialog 817
Analysis Options dialog box 920
Repackaging Wizard 920
Analyze
about tests 1778
ACT Summary tab 1813
benefits of 1774
creating custom mobile tests 1868
icon legend 1815
integrating with Microsoft ACT 1813
Mobile Test Wizard 1868
overview 1772
performing compatibility and best practices testing 1794
performing conflict testing 1811
reference 1842
resolutions 2121
resolving issues 1838
selecting automatic fix preferences 1786
selecting tests to execute 1780
setting compliance level for OS and browser compatibility tests 1782
status icons 1815
suppressing issues 1837
tasks you perform using 1772
viewing ACT test results 1813
viewing and filtering test results 1814
viewing application virtualization compatibility test results 1824
viewing conflict test results 1833
viewing detailed package results 1819
viewing group or application test results 1817
Viewing MSIX Conversion Compatibility Test Results 1823
viewing OS and browser compatibility test results 1821, 1824
viewing remote application publishing compatibility test results 1828
viewing results in 1773
viewing virtualization and Windows Installer best practices test results 1830, 1832
Analyze Application View 1845
Analyze Deployment Type View 1845
Application Virtualization Compatibility tab 1849
Inter-Application Conflicts tab 1852
OS and Browser Compatibility tab 1847
Virtualization and Windows Installer Best Practices tab 1850
Analyze Group View 1843
Analyze tests 1879
Android mobile risk assessment tests 2030
Apple best practices 2010
Apple iOS 1973
Apple mobile risk assessment 2055
Apple OS X 1973
application conflict tests 2082
application virtualization compatibility 1981
Azure application services 2109
best practices and risk assessment 1988
choosing virtual formats to display in test results 1987
desktop risk assessment 2077
Google Android 1976
installer analysis 1981
Mac OS risk assessment 2077
Microsoft App-V best practices 2000
Microsoft App-V conflict 2104
mobile risk assessment 2012
operating system compatibility 1881
package data conflict 2082
reference 2120
remote application publishing compatibility 2108
remote desktop services 2113

risk assessment 2012
web deploy best practices 2010
Windows Installer best practices (ACE) 1990
Windows Installer Internal Consistency Evaluators (ICEs) 1988
Windows mobile risk assessment tests 2013
Windows Phone tests 1970
Windows risk assessment 2080
Windows Server and desktop 1882
Android mobile risk assessment tests 2030
anonymous
Customer Experience Improvement Program (CEIP) 145
anonymous authentication 167
anti-virus software 978
excluding directories when using Snapshot method 861
-app 922
App Portal
categories dialog box 402
checking Default Category setting 409
connecting to 727
enabling auto creation of catalog item 401
entering catalog metadata 403
integrating with AdminStudio (with FlexNet Manager Platform) 245
integration with AdminStudio (without FlexNet Manager Platform) 250
keywords 407
managing metadata 400
overview of support 400
permissions required by AdminStudio user account 249
reviewing catalog item title 403
selecting a catalog item template 406
setting brief description and long description 403
specifying catalog item category 404
specifying catalog item keywords 407
synchronizing Application Catalog applications with 245
templates 406
App Portal Information tab 401
Append Package Version setting 1410
Apple best practices tests 2010
Apple mobile risk assessment tests 2055
Application Catalog 224, 230, 259, 261
AdminStudio default 230
connecting to 220, 221
connecting to existing 220
creating new 224
disconnecting from 261
integration with InstallShield Editor 1684
required permissions on 214
searching 259
specifying a default for an AdminStudio enterprise 230
specifying default 230
specifying required database permissions 214
standalone 213
upgrading 229
upgrading 5.0 or 5.5 or 6.0 Application Catalogs 229
upgrading legacy 229
using the Software Repository 525
Application Configuration view 1761
Tuner 1761
Application Conversion Project Wizard 1064, 1191
using 1066
Application Conversion Project Wizard Complete panel 1076
Application Conversion Wizard 1065, 1181, 1214
using 1133, 1134
application isolation 958, 961, 2153, 2155, 2156, 2157, 2160, 2161, 2162, 2163
assemblies 2156
certificate store 2161
certificates 2157
code signing 2160
concept 2153
digital signatures 2157
filtering file listings when manually configuring 2163
manifests 958, 2157
methods 2155
modifying default recommendations 2162
private key 2161
setting assembly naming conventions 2162
software publishing credentials 2160
using Repackager vs. Application Isolation Wizard 2155
Application Isolation Wizard 2153, 2155, 2157, 2160, 2161, 2164, 2165, 2166, 2169, 2170
Advanced Options dialog 2166
Application Isolation Progress panel 2166
Application Manifest Properties dialog 2170
Assembly Properties dialog 2169
AssemblyType property 2170
CertificateFile property 2170
CertificateName property 2170
code signing technologies 2160
command-line options 2170
Company property 2170
comparison to isolation using Repackager 2155
Completing the Application Isolation Wizard panel 2166
configuration files 2170
digital signatures 2157, 2167
Division property 2170
IsolatedComponents property 2170
Isolation Method panel 2165
isolation methods 2155
launching 2155
Manifest and Assembly Design dialog 2168
Manifests property 2170
NewComponents property 2170
private key 2161
PVKFile property 2170
reference 2164

setting manifest options 2166
software publishing credentials 2160
SPCFile property 2170
Summary panel 2165
TimeStamp property 2170
using Repackager to isolate Windows Installer packages 2155
Welcome panel 2164
Windows Installer File Selection panel 2165
application life cycle 246
Application Link 1637
Application Manager 207, 262, 624, 1787, 1812, 1839, 1841, 1843, 1845, 1853, 1854, 1860, 1866, 1867, 1872, 1873, 1874, 1875, 1876, 1877
adding groups 262
All Merge Modules view 634
All Patches View 2145
Associated Patches View 2145
Associated Patches view 1854
automatically resolving conflicts 1839
changing the default validation file in 1787
Components view 635
Conflict Wizard 1812, 1866, 1867
Consolidated Patch Report 2148
copying packages to multiple groups 263
deleting groups 262
deleting packages 264
Dependencies View 2145
Dependencies view 624
Dependency view 635
editing groups 263
Enterprise Policy view 642
Exclusion view 635
extended attribute description file 473
extended attributes 472
extended attributes and Workflow Manager 475
Extended Attributes view 623
Files view 636
Files/Components view 626
Group view 550
icons 538
Impact Analysis View 2145
Import Wizard 784, 813, 814
importing 265, 289, 305, 306
importing merge modules 304
importing OS snapshots into 305
importing other setup types into 290
importing packages in multiple configurations into 306
INI File Changes view 627, 641
manually resolving conflicts 1841
menus 530
Merge Module Import Wizard 816
Merge Module view 634, 640
Merge Modules view 631
modifying groups 263
moving groups in 264
moving OS snapshots in 264
moving products in 264
organizing products into groups 262
OS Snapshot view 640
Output Window 548
Patch Impact view 1853
patches 637
Patches Group View 637
Patches tab 637, 639, 2145
Patches view 637, 639, 2145
performing patch impact analysis 2143
Product view 538, 636
Registry view 627, 641
removing groups 262
Reports tab 361
resolving conflicts 1838
Rules Wizard 1872, 1873, 1874, 1875, 1876, 1877
scanning packages for dependencies 624
searching the Application Catalog 259
shortcut menus 543
Shortcuts view 629, 642
Tables view 633, 642
toolbar 530
user permissions 850
using extended attributes 472
viewing patch and patch impact information 2145
viewing patch impact analysis results 2143
Application Manager dialogs
Find 671
Group Properties 680
Application Manager extended attributes 472
using 472
Application Manager groups 262, 263
adding 262
deleting 262
editing 263
modifying 263
removing 262
Application Manager Options dialog 680, 681, 682, 683, 703, 704
Duplicate Package tab 683
Flexera Service Gateway tab 727
Application Manager Output Window 548
Application Manager views
All Merge Modules 634
Components 635
Dependency 635
Enterprise Policy 642
Exclusion 635
Files 636
Files/Components 626
Group 550
INI File Changes 627, 641

Merge Module 634, 640
Merge Modules 631
OS Snapshot 640
Products 636
Registry 627, 641
Shortcuts 629, 642
Tables 633, 642
Application Manifest Properties dialog 2170
application manifests 958, 2157
application object template files 934
Application Sync 1639
Application View
App Portal Information tab 401
application virtualization 1044
application virtualization compatibility tests 1780, 1981
running 1794
viewing results 1824
applications 2161
distributing to AirWatch Server 2233
distributing to Citrix XenApp Server 1148, 2232
distributing to Symantec Altiris Management Server 1148, 2232
distributing to System Center 2012 Configuration Manager 1148
isolating 2161
moving in Application Manager 264
organizing into groups in Application Manager 262
Applications page 1527, 1650
creating a new application shortcuts 1509, 1628
excluding or deleting a shortcut 1629
including an existing shortcut 1629
AppLink 1637
adding reference 1638
Required and optional links 1638
AppLink Reference dialog box
relative links 1639
AppLink references 1665
AppLink settings 1664
collisions and order of import 1666
entering a relative path 1668
example references 1669
required and optional linked applications 1666
required vs. optional 1668
Required vs. Optional applications 1666
AppLink Settings dialog box 1638, 1664
deleting a reference 1639
AppSync settings 1639
benefits of 1639
Clear Sandbox option 1640
configuring 1640
expiration 1670
Expire Period 1671
Expire Period option 1641
Frequency of update 1640
frequency of update 1640
Message displayed 1640
messages 1641
setting expiration settings 1640
URL of update 1640
Use Application Expiration 1671
Warning Frequency 1672
Warning Frequency option 1641
Warning Message 1672
Warning Period 1671
Warning Period option 1641
AppSync Settings dialog box 1669
App-V
specifying client runtime drive 1228
specifying deployment settings 442
App-V 4.6 1050
App-V 4.x Application Launcher 1137
App-V 4.x applications
testing using App-V 4.x Application Launcher 1137
App-V 5
browser extensions 1050
App-V Append Version property 1163
App-V Application Launcher
for testing with packages edited in Virtual Package Editor 1445
App-V applications 1011
about 1045, 1046
automatically creating from Repackager 1043
components of the package 1046
creating 64-bit App-V applications 1050
creating using InstallShield Microsoft App-V Assistant 1042
editing OSD file 1051
editing OSD file to identify location of App-V server 1051
evaluating AdminStudio's App-V support 80
integration of Windows Services 1052
location of generated files 949
methods to convert Windows Installer packages 1041
start up and shut down sequences 1052
support for Windows services 1052
App-V Assistant 948, 1042, 1493, 1522, 1527
about 1498
adding an existing folder to an App-V package 1504
adding diagnostic tools to an App-V package 1502, 1533
adding files to an App-V package 1504
application features requiring pre- or post-conversion actions 1523
Applications page 1527
benefits of using 1495
Build Options page 1529
building a Windows Installer package with build output 1517
building an App-V package 1519
building App-V package in Direct Edit mode 1516
compared with Microsoft App-V Sequencer 1496
controlling the display of predefined folders 1506

creating 64-bit App-V applications 1050
creating an App-V package 1500
creating new application shortcut executables 1509
deployment server 1502
Dynamic Suite Composition page 1528
enabling App-V package building when editing a Windows Installer package 1516
error messages 1523
File Mapping dialog box 1534
Files page 1527
Home page 1523
HTTP protocol 1526
including an existing shortcut 1510
integration with Project Assistant and Installation Designer 1492
managing files and folders 1503, 1504
modifying build options 1515
modifying registry settings 1511
modifying shortcuts 1509
Package Information page 1524
Package Isolation Options dialog box 1535
Package Optimizations dialog box 1518, 1538
performing dynamic suite composition 1514
reference 1523
Registry Isolation Options dialog box 1536
Registry page 1527
renaming a shortcut 1511
selecting releases to build 1515
specifying deployment server 1502
specifying general settings 1501
specifying OS requirements 1501
specifying primary application directory 1506
specifying upgrade information 1502
support for Windows services 1500
supported InstallShield project types 1499
App-V Client 1046
App-V Comments property 1160
App-V Compression property 1165
App-V Data Type setting 1472
App-V Dynamic Suites property 1165
App-V Feature Block 1 setting 1471
App-V GUID setting 1471
App-V package 1405
adding shortcuts to 1430
editing 1405
extracting files from 1425
file extensions 1434
including debug tools with 1447
registry isolation 1426
scripts for 1439
App-V Package Optimization property 1162
App-V Package Upgrade Settings dialog box 1533
App-V packages
about 1495, 1497
adding diagnostic tools to 1502, 1533
adding existing folder 1504
adding files to 1504
adding or deleting registry keys and values 1512
benefits of using App-V Assistant to create 1495
building 1519
building a Windows Installer package with build output 1517
building from the command line 1522
components of 1497
compressing packages for 1516
conversion error and warning messages 1523
creating with InstallShield 1493
defining shortcuts 1509
excluding vs. deleting a shortcut 1511
feature blocks 1518
files included in 1497
how transforms are included 1499
including additional Windows Installer packages 1517
including an existing shortcut 1510
inheritance of isolation options from folders to files 1508
inheritance of isolation options in the registry 1513
managing files and folders 1504
modifying registry entries 1511, 1512
overview of 1494
package optimizations 1518
renaming a shortcut 1511
selecting application shortcuts 1509
setting isolation options 1508
shortcut requirements 1509
specifying package name 1501
steps to create with App-V Assistant 1498
support for Windows services 1500
App-V Root Folder Name property 1165
App-V Runtime Drive property 1165
App-V Sequencer 1046
App-V Server 441, 443, 1046
specifying advanced deployment settings 442
App-V Server Host property 1129, 1164
App-V server location
in OSD file 1051
App-V Server Path property 1129, 1165
App-V Server Port property 1129, 1164
App-V Server Protocol property 1130, 1165
App-V Supported OS property 1162
App-V Upgrade Package property 1163
App-V Version setting 1471
ARPCOMMENTS 1762
ARPCONTACT 1762
ARPHELPLINK 1762
ARPHELPTELEPHONE 1762
ARPNOMODIFY 1762
ARPNOREMOVE 1762
ARPNOREPAIR 1762
ARPURLINFOABOUT 1762

ARPURLUPDATEINFO 1762
assemblies 961, 2156, 2157, 2160, 2161, 2162, 2164
  digital signatures 2157
  in application isolation 2156
  private key 2161
  servicing published shared 2164
  setting naming convention 961, 2162
  signing 2160
  software publishing credentials 2160
assemblies and manifests 2155, 2156
  isolation method 2155
assembly files 956
assembly manifests 958, 2157
Assembly Properties dialog 2169
AssemblyType 2170
Associate with Workflow Manager Application dialog 653
Associated Patches view 1854
associated tools 130
  running in projects 130
attributes 183
  creating directory services attributes 183
authentication
  for Application Catalog or Microsoft SCCM Server 1104
authoring packages 1683
Automated Application Converter 1039
  about 1060
  App-V client runtime drive 1228
  benefits of 1061
  best practices 1254
  command line parameters 1233
  compress the MSI wrapper option 1227
  creating 64-bit App-V applications 1050
  getting started 1064
  launching 1063
  location of output 1053
  Machines tab 1170
  menus and toolbars 1180
  opening a new or existing project 1065
  overview diagram 1062
  preparing virtual machines 1086
  problems connecting to HyperV image 1256
  project files 1079
  reference section 1151
  selecting rules when adding packages from a directory 1200
  setting default project properties 1149
  setting project options 1149
  support for Windows services 1052
  supported operating systems 1062
  supported virtual machines 1063
  user interface 1152
  using in evaluation mode 1083
  using list features 1186
  viewing debug messages 1258
  wizard comparison 1064, 1192
  workflow diagram 1062
  wrapper MSI 1226
Automated Application Converter Log
  viewing debug messages 1258
Automated Repackaging on Virtual Machines panel 1075, 1135
automatic issue resolution 1838
automatic login 173
  using directory services 173
Automatic Repackaging on Virtual Machines panel 1209
Azure application services tests 2109

B

-b 922
batch convert option
  evaluating 80
batch converter 1039
best practices 527, 1254
  OS Snapshot 527
best practices and risk assessment tests 1988
best practices tests 1778
Browse for Folder dialog box 1109, 1218
Browse local machine 509, 1072, 1096, 1203, 1205
Browse local machine and network 1069
browser extensions
  in App-V 5.0 1050
Build App-V Packages 1223
Build Citrix Profiles 1223
Build Options page 1529, 1653
Build Settings page 1594
  digitally signing a Citrix profile 1579
Build ThinApp application 1655
Build ThinApp Packages 1223
Build Virtual Package 1530, 1656
Build Windows Installer Packages 1223
Build Wrapper MSI setting 1411
BuildCompressed 1027
building App-V packages from the command line 1522
building profiles using command line 1605
building ThinApp applications using command line 1673
buying InstallShield 89

C

CAB file 1058
CAB files 1705
  Tuner 1705
cancel 1181
Captured Installation view 999
capturing OS snapshots 527, 528
catalog item
  specifying categories 404
catalog items 401
Categories dialog box 402, 404, 654

CEIP (Customer Experience Improvement Program 145
certificate name in the store 2160
CertificateFile 2170
CertificateName 2170
certificates 2157, 2161
  certificate store 2161
  private key 2161
-cf 922
checking for updates 144
checklist 1739, 1740
Choose Registry Key dialog box 1025
Citrix 947, 1012, 1055
  Citrix profile 938, 1012
  Citrix XenApp 947, 1055
Citrix Assistant 948, 1042, 1547, 1605
  about 1550
  adding an existing folder to a profile 1563
  adding diagnostic tools to profile 1556, 1598
  adding files to a Citrix profile 1562
  adding or deleting registry keys and values from a Citrix profile 1577
  adding pre-launch and post-exit scripts to profile 1560
  application features requiring pre- or post-conversion actions 1605
  Build Settings page 1594
  building a Citrix profile 1580
  building Citrix profile in Direct Edit mode 1580
  controlling the display of predefined folders 1566
  creating a Citrix profile 1554
  creating new folder 1565
  creating new profile shortcut 1573
  deleting files and folders 1566
  Diagnostic Tools dialog box 1556, 1598
  digitally signing a profile 1579
  enabling Citrix profile building when editing a Windows Installer package 1580
  error messages 1605
  excluding or deleting a profile shortcut 1574
  Home page 1583
  how requirements are applied at runtime 1559
  how transforms are included 1554
  including an existing profile shortcut 1574
  inheritance of isolation options 1570
  inheritance of isolation options in registry 1578
  integration with Project Assistant and Installation Designer 1492
  managing files and folders 1562
  modifying build settings 1578
  modifying registry settings 1576
  modifying shortcut settings 1570
  moving files and folders 1565
  overview 1547
  overview of 1547
  overview of Citrix profiles 1053, 1551
  overview of isolation options 1568
  Profile Files page 1589
  Profile Information page 1584
  Profile Registry page 1593
  Profile Requirements page 1586
  Profile Shortcuts page 1591
  reference 1582
  Registry Isolation Options dialog box 1602
  renaming a shortcut 1575
  Script Execution dialog box 1597
  selecting releases to build 1579
  Service Packs Requirement dialog box 1604
  setting isolation options 1567
  setting isolation options for folders and files 1569
  setting isolation options on files 1599
  setting isolation options on folders 1601
  setting language requirements 1558
  setting operating system and service pack requirements 1557
  setting registry isolation options 1577
  setting Service Pack requirements 1604
  specifying profile information 1555
  specifying profile name and version 1555
  specifying profile requirements 1557
  specifying whether users can update applications 1555
  steps to create a Citrix profile 1548
  supported InstallShield project types 1554
  when to exclude or delete shortcuts 1575
Citrix profile 938, 947, 1012, 1055, 1056
  about custom actions 1325
  about services 1325
  and COM+ applications 1327
  automatically creating from Repackager 1043
  benefits of deploying 1058
  CAB file 1058
  creating using InstallShield Citrix Assistant 1042
  features requiring pre-or post-conversion actions 1325
  hash file key for digital signatures 1058
  hashes file 1058
  ignored tables 1325
  location of generated files 1057
  methods to convert Windows Installer packages 1041
  overview of 1053, 1057
  overview of Citrix XenApp 1056
  profile manifest file 1058
Citrix profiles 1549
  about 1551
  adding diagnostic tools to 1556, 1598
  adding existing folder 1563
  adding files to 1562
  adding or deleting registry keys and values 1577
  adding pre-launch and post-exit scripts 1560
  adding scripts 1560
  benefits of deploying 1552

building using command line 1605
CAB file 1551
Citrix Assistant 1547
conditions when Shortcut should be excluded or deleted 1575
conversion error and warning messages 1605
creating with InstallShield 1547
digitally signing 1579
hashes.txt file 1551
how requirements are applied at runtime 1559
how shortcuts are implemented 1571
how transforms are included 1554
Icons file 1551
icons.bin file 1551
including an existing shortcut 1574
inheritance of isolation options from folders to files 1570
inheritance of isolation options in the registry 1578
managing files and folders 1562
moving files and folders 1565
myapp.profile file 1551
overview of 1551
overview of Citrix Assistant 1547
overview of Citrix XenApp 1549
overview of isolation options 1568
Profile Manifest file 1551
renaming a shortcut 1575
Scripts folder 1551
setting file isolation options 1599
setting folder isolation options 1601
setting isolation options 1567
setting isolation options for folders and files 1569
setting language requirements 1558
setting operating system and service pack requirements 1557
setting registry isolation options 1577
setting Service Pack requirements 1604
shortcut requirements 1573
shortcuts and the isolation environment 1571
specifying whether users can update applications 1555
steps to create with Citrix Assistant 1548
Citrix XenApp 1055, 1056, 1547, 1549
about 1549
about COM+ applications 1327
and services 1326
benefits of 1058
managing package deployment data 447
overview of 1056, 1549
overview of Citrix profiles 1053, 1057, 1551
specifying deployment settings 448
Citrix XenApp Assistant. See Citrix Assistant.
Citrix XenApp Profiles (*.profile) 1075, 1209
Citrix XenApp Server 447
distributing applications to 1148, 2232
specifying advanced deployment settings 450
support for 233
class IDs 2184
checking 2184
Class IDs view 2218
clean machines 859, 860
alternate-language repackaging on 860
Repackager 859
Clear Sandbox 1640
Close wizard to configure packages and machines 1073
code signing 2160
certificate name in the store 2160
credentials 2160
Collect Product Information panel 912
Repackaging Wizard 912
Column selector 1183
columns 1188
changing order 1188
selecting on the Packages and Machines tabs 1183
COM In Process Enabled 1475
COM Objects Interaction 1164, 1229, 1474
COM Out of Process Enabled 1475
COM+ applications
and Citrix profile 1327
and Citrix XenApp 1327
command line 897, 1605, 1673, 2202
running QualityMonitor 2202
running Repackaging Wizard from 897
command line parameters 1233
Command Line property 1157
Command tab 1741
command-line configurations 148
creating 148
deleting 148
editing 148
Command-Line Configurations panel 151
Add Tool Wizard 151
command-line options 922, 1723, 2170
Application Isolation Wizard 2170
Repackager 922
when to use Dialogs view instead 1723
command-line properties 143
Add Tool Wizard 143
Command-Line Properties dialog 143
DevLocation variable 143
InstallLocation variable 143
ProjectName variable 143
SharedPoint variable 143
SourcePackage variable 143
TargetDir variable 143
TargetFileName variable 143
Command-Line Properties dialog 143
Add Tool Wizard 143
COMMapping 1027
Comments 1762

comments 1726
adding and editing setup property 1726
companies
administrator and consumer 154
Company 2170
Company property 1157
Complete Analysis 1783
Completing the Application Isolation Wizard panel 2166
compliance levels
complete analysis 1782
industry standard analysis 1782
industry standard with auto-fixes 1782
component settings options 947, 1017
Components view 635
Compress Wrapper MSI setting 1411
Compressed property 1158
Compressed setting 1474
Compression 1228
compression of ThinApp applications 1627
Compression Type 1656
configuration files 2170
Application Isolation Wizard 2170
Configuration tab 148
Tool Properties dialog 148
configuring command lines 151
conflict resolution 1811
conflict tests 2082
running 1811
viewing results 1833
Conflict Wizard 1812, 1866, 1867
identifying conflicts 1812
Summary panel 1867
Target Information panel 1867
conflicts 1811, 1812, 1839, 1841
automatically resolving in Application Manager 1839
identifying with Conflict Wizard 1812
manually resolving in Application Manager 1841
resolving 1838
solution using virtualization 1044
conflicts tests 1779
ConflictSolver
Consolidated Patch Report 2148
Patch Impact Analysis Wizard 2143, 2149
Patch Impacts View 2145
performing patch impact analysis 2143
viewing patch impact analysis results 2148
ConflictSolver. See Application Manager.
Connect to a Microsoft System Center Configuration Manager Server panel 2260
Connect to an AdminStudio Application Catalog panel 1104, 1195
Connect to Application Catalog dialog 658, 735
Connect to Machine 1102, 1176
connections
creating 233
Consolidated Patch Report 2148
consumer company 154
Contact Person 1762
Contents 1182
context file 1150
Control Access via Active Directory 1615, 1647
converting 933, 934, 937
.aot 934
.axt 934
.inc 933
.ipf 934
.isl 937
.txt 937
.wse 937
InstallScript MSI 895
InstallShield log files 937
legacy setups 933
Novell ZENworks projects 934
Repackager 3.x output 933
SMS projects 934
WinINSTALL projects 937
Wise Installation projects 937
converting packages 1066
Copy In 1178
Copy Out 1178
Copy Role page 202
copying 263
packages to multiple Application Manager groups 263
roles 199
Create a new transform file option 1745
Create Report dialog box 989
CreateSetupExe 1027
creating
accounts 155
directory service connections 160
roles 198
creating new workflows 129
credentials 2160
-cs 922
custom actions
and Citrix profiles 1325
custom installation 1702
preventing feature display 1702
Custom Report 2317
Custom report
creating 2316
custom reports
creating 2316
custom setups 1722
disabling 1722
Custom SQL Query Report 2322
Custom SQL Query report
creating 2316

Custom Stored Procedure report
creating 2316
Custom Stored Procedures Report 2304, 2323
custom tables 682
customer
Customer Experience Improvement Program (CEIP) 145
Customer Experience Improvement Program (CEIP) 145
canceling membership 145
joining 145
Customize dialog box 1741

D

data sources 1715
adding new 1715
data type appearance 929
changing Repackager 929
database
assigning required permissions 214
Database Name 1196
DDE Application setting 1469
DDE Command setting 1469
DDE Ifexec setting 1469
DDE Topic setting 1469
deactivating AdminStudio 78
debug log 123
generating 123
debug messages
viewing 1185, 1258
Default Destination 1701
Default Organization 1701
Default.ini 1033
Deleted Files view 1006
Deleted Registry Entries view 1007
Deleted.isr 1033
deleting
accounts 159
directory service connections 162
groups 159
roles 200
users 159
deleting added setups 871
dependencies 950
detecting in Repackager 950
viewing in Application Manager 2145
dependencies of a virtual package 1418
Dependencies view 624, 1463
Dependency view 635
deploying SMS 1730
deployment data 419
Deployment Status 2198
install or configure products or features 2198
reinstall components 2200
reinstall features 2199
verify files 2197
view properties 2197
Deployment Status view 2226
deployment testing 2183
deployment tests 2217
executing all from command line 2217
executing all from Interface 2217
running silently 2217
description file 473
Application Manager extended attributes 473
Description property 1701
editing for features 1701
description property 1702
desktop risk assessment tests 2077
Destination Group panel 813
Import Wizard 813
destination variable 1702
DeviceDriverMSIX001 1977
DevLocation 143
diagnostic tools
adding to App-V package 1533
adding to Citrix profile 1556, 1598
adding to ThinApp application 1616, 1659
Diagnostic Tools dialog box 1556, 1598, 1616, 1659
dialog 1723
editing properties for 1723
Dialog Properties dialog 1743
dialog sequences 1721
restoring 1721
dialogs 141, 142, 144, 147, 1721, 1722, 1723, 1742, 1743, 2166
About AdminStudio 141
Add New Tool 142
AdminStudio Interface 141
AdminStudio Options 144
Advanced Options 2166
Command-Line Properties in Add Tool Wizard 143
Dialog Properties dialog 1743
hiding during UI sequences 1721
suppressing License Agreement 1722
suppression issues 1723
Tool Properties 147
Transform Summary 1742
working with in Tuner 1721
Dialogs view 1721, 1722, 1723, 1743, 1762
Dialog Properties dialog 1743
disabling custom setups from 1722
editing dialog properties in 1723
hiding dialogs during UI sequences from 1721
restoring dialog sequences from 1721
suppressing License Agreement dialog from 1722
when to use over MSI command-line options 1723
digital certificates 2157, 2161
certificate store 2161
Digital Signature tab 2167

Advanced Options dialog 2167
digital signatures 2157, 2160, 2161, 2167
Application Isolation Wizard 2167
private key 2161
software publishing credentials 2160
digitally signing
a Citrix profile 1579
Direct Editor 1731, 1732, 1766
adding a new record 1732
adding a new row 1732
editing packages 1731
finding and replacing in 1732
from Validation tab 1732
directories 973
exclusions 973
Directories and Files Excluded During Analysis dialog box 1024
directories and subdirectories 965
excluding 965
directory
package selection rules 1200
directory exclusions 973
directory services 183
automatic login 169
creating connection to 160
deleting attributes 165
deleting connection to 162
Directory Services Attributes Administration page 183
importing accounts 156
importing users 176
LDAP attributes 183
managing 160
managing attributes 163
managing connections to 160
managing directory services configurations 160
setting up attributes 163
updating connection to 162
viewing connection to 179
Directory Services Account/Group Add page 176
Directory Services Attributes Administration page 183
Disable Log Monitor Tracing 1656
Disable Modify Button 1762
Disable Remove Button 1762
Disable Repair Button 1762
disabling 1726
Modify button in Add/Remove Programs 1726
Remove button in Add/Remove Programs 1726
Repair button in Add/Remove Programs 1726
disabling a user account 159
disconnecting from Application Catalog 261
distribution 1728, 2240
copying to FTP server 1728
copying to network location 1728
preparing packages 2240
Distribution Output panel 2267
Distribution Wizard (Package) 2267
Distribution Summary panel 2267
Distribution Wizard (Package) 2267
Distribution System tab 233
distribution systems 233
creating multiple named connections 233
Distribution Type panel 2259
Distribution Wizard (Package) 2259
Distribution Wizard
Altiris Integration Panel 2263
Altiris XML Template 2265
Distribution Wizard (Package) 2240, 2241, 2242, 2243, 2250, 2259, 2260, 2263, 2266, 2267
Administrative Install panel 2260
Connect to a Microsoft System Center Configuration Manager Server panel 2260
creating administrative installations using 2241
distributing packages to FTP servers 2242
distributing packages to Microsoft System Center 2007 Configuration Manager 2244
distributing packages to network locations using 2243
Distribution Output panel 2267
Distribution Summary panel 2267
Distribution Type panel 2259
FTP Location panel 2263
Installation Package panel 2266
LANDesk Integration Panel 2266
Network Location panel 2266
preparing packages for distribution using 2240
publishing packages to Microsoft System Center 2007 Configuration Manager 2244
script-based setup.exe 2250
Welcome panel 2259
ZENworks Configuration Management 2246, 2268
Distribution Wizard (Package) for ZENworks Configuration Management 2246, 2268
Division 2170
DLL Hell 2153
avoiding 2153
DLL Information tab 1858
DLL-Based ACEs panel 1877
Rules Wizard 1877
documentation tool 902
Duplicate Package options 683
Duplicate Package tab 683
Dynamic Dependency Scanner 957
Dynamic Suite Composition 1418
configuring for an App-V package 1418
dynamic suite composition 1514
Dynamic Suite Composition page 1528

E

Edit Keywords dialog box 670

Edit menu 1180
Edit Registry Key dialog 1026
Edit Registry Key Dialog Box 1026
editing packages 1731
  Direct Editor 1731
Enable Browser Helper Objects 1475
EnablePathVariables 1027
Enforce Security Descriptors setting 1474
engines 853
  including InstallScript engines with installation 853
enhancement
  Customer Experience Improvement Program (CEIP) 145
enhancing packages 1683
ensuring package quality 2175
Enterprise Policy
  view 642
Error 0x800A1518 222
Error -4308 - VM failed to start up 1258
Error -4309 - VM failed to shut down 1258
Error -4310 - Failed to connect to VM 1259
Error -4312 - Failed to prepare Repackager 1259
Error -4313 - Failed to access the package 1260
Error -4314 - Failed to copy repackaged output from virtual machine 1261
Error -4315 - Failed to send command to VM 1261
Error -4316 - Failed getting response from VM 1262
Error -4317 - Failed running pre-snapshot 1262
Error -4318 - Failed running post-snapshot 1263
Error -4319 - Failed running package installation 1263
Error -4320 - Failed creating folder on VM 1264
Error -4333 - Preparing command-line... 1264
Error -4380 - Failed to prepare AppV 1265
Error -4388 - Failed preparing for pre-snapshot 1265
Error -4389 - Failed connecting to server 1266
Error -4390 - Failed connecting to image 1266
Error -4391 - Failed to reboot 1267
Error -4395 - Failed to create VM directory 1267
Error -4409 - Failed to delete package cache folder 1268
error control level 1719
  setting 1719
Error Control setting 1486
Errors property 1177
evaluating
  AdminStudio's Microsoft App-V support 80
  batch convert option 80
evaluation files 1696, 1699
evaluation mode 1083
Event setting 1483
Excluded Processes dialog box 919
excluding files 964
exclusion list 1033
  specifying directory 120
exclusion lists 2190
Exclusion view 635
exclusions 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 1020, 1021, 1022, 1023
  all files in a directory 965
  all shortcuts in a directory 967
  anti-virus software 978
  configuring in Repackager 964
  directories and subdirectories 965
  directory 973
  editing existing file 973
  editing existing INI file 975
  editing existing registry 977
  external configuration file in Repackager 968
  file 971
  files 1021
  INI file 974
  INI file sections 967, 974
  INI files 966, 1022
  modifying Repackager external configuration file 969
  OS snapshot 969
  OS Snapshot Wizard 970
  OS Snapshot Wizard global 970
  project 969
  registry 1023
  registry data 976
  registry keys 966
  registry values 966
  removing existing file 973
  removing existing INI file 976
  removing registry 977
  Repackager 970
  Repackager file 964
  Repackager global 970
  repackaging 969
  resetting to default values 1020
  shortcuts 967
  shortcuts from subdirectories 967
  specific file extensions 972
  with specific file extensions 972
Exclusions Editor 969, 970, 971, 972, 973, 974, 975, 976, 977, 1020, 1021, 1022, 1023, 1024, 1025, 1026
  About dialog box 1026
  Choose Registry Key dialog box 1025
  Directories and Files Excluded During Analysis dialog box 1024
  directory exclusions 973
  Edit Registry Key dialog box 1026
  editing existing file exclusions 973
  editing existing INI file exclusions 975
  editing existing registry exclusions 977
  excluding INI file sections 974
  File Exclusion Information dialog box 1024
  files 971
  Files tab 1021
  INI File Exclusion Information dialog 1025

INI files 974
INI Files tab 1022
Keys Excluded During Registry Analysis dialog box 1025
menus 1020
OS Snapshot Wizard exclusions 970
reference 1020
registry data 976
Registry tab 1023
removing existing file exclusions 973
removing existing INI file exclusions 976
removing registry exclusions 977
Repackager exclusions 970
resetting exclusions to default values 1020
specific file extensions 972
Exclusions tab 997
Execute Tests button 1773
Expand App-V Package 1163, 1229
experience
Customer Experience Improvement Program (CEIP) 145
expiration 1640, 1670
Expiration Message 1672
Expire Period 1671
Explore 1181
exporting
reports 2309
Expression Builder dialog 1858
extended attributes 472, 473, 475
and Workflow Manager 475
description file in Application Manager 473
in Application Manager 472
using in Application Manager 472
Extended Attributes view 623
Application Manager 623
external configuration file 968, 969
modifying Repackager 969
specifying in Repackager 968
ExtraHKCRPermanent 1027

F

FAQ 134
feature 1700
changing visibility 1700
setting initial state of 1700
feature advertisement 1703
Feature Block 1 Size setting 1474
feature blocks 1518
feature properties 1702
features 1701
editing description for 1701
Features view 1701, 1750
Description property 1701
FILE 1526
file associations 2185
checking 2185
File Associations view 2218
File Exclusion Information dialog box 1024
file exclusions 964, 973
editing existing 973
removing existing 973
Repackager 964
File Isolation Options dialog box 1599
File Locations tab 1741
File Mapping dialog box 1534
File menu 1180
files 1704, 1705
adding 1704
displaying from base Windows Installer package 1704
excluding 964
exclusion of specific extensions 972
exclusions 971, 1021
preventing installation of from the Windows Installer package 1705
removing added 1705
storing in CAB in Tuner 1705
Files & Folders page 1648
creating a new folder 1620
deleting files and folders from a ThinApp application 1621
files and folders 1704
Files and Folders view 1001, 1704, 1705, 1751
adding files in 1704
displaying files from the base Windows Installer package 1704
preventing installation of files from the Windows Installer package 1705
removing added files from 1705
Files page 1527
Files tab 1021
Exclusions Editor 1021
Files view 636
Files/Components view 626
Filesystem 1166, 1228
filtering test results 1837
Find dialog 671
finding and replacing 1732
using Direct Editor 1732
Finishing INI File Import panel 1768
Import REG File Wizard 1768
Finishing Registry Import panel 1769
Import REG File Wizard 1769
First Action Delay setting 1487
First Error setting 1487
Flexera Service Gateway 245
connecting to 727
connection settings 727
output messages 250
FlexNet Manager Platform
connecting to 727

integrating with AdminStudio 245
permissions required by AdminStudio user account 249
synchronizing Application Catalog applications with 245
FlexNet Manager Suite
searching 252
floating licenses 101
Folder Isolation Options dialog box 1601, 1660
frequency 144
frequently asked questions 134
FTP Location panel 2263
Distribution Wizard (Package) 2263
FTP server 1728
copying installation to 1728
FTP servers 2242
distributing packages to using Distribution Wizard (Package) 2242
Full isolation option 1624
Full VFS Write Mode 1164, 1230, 1475

G

General Information panel 1873
Rules Wizard 1873
General Information tab 1855
General Information view in Virtual Package Editor 1472
General tab 995
Generated property 1177
generating reports 2309
generic transforms 1695, 1742
creating 1695, 1742
Get-ASApplicationDeploymentSummary 2387
Get-ASApplicationDetails 2372
Get-ASApplicationID 2372
Get-ASAppPortalCategories 2373
Get-ASAppPortalTemplates 2374
Get-ASConfigPlatform 2386
Get-ASDeploymentSystemPackageTree 2388
Get-ASPackage 2391
Get-ASPackageTestSummary 2392
Get-ASProperty 2394
Get-ASTestDetails 2395
Get-ASTestState 2395
Get-ASVirtualReadiness 2397, 2400
Global Exclusions Editor
exclusion list 120
Global Exclusions list 120
Group Dependencies setting 1486
Group Properties dialog 680
Group view 550
grouping 1188
lists 1188
groups 262, 263, 264
adding in Application Manager 262
copying packages to multiple groups 263
deleting 159
deleting from Application Manager 262
editing properties in Application Manager 263
moving in Application Manager 264
organizing products into in Application Manager 262
removing from Application Manager 262
sharing packages between multiple groups 263
Guest Agent 1219
Guest Password property 1100, 1174
Guest Username property 1100, 1174
GuestAgent Path property 1101, 1174

H

Hard Time-Out property 1158
hashes file 1058
hashes.txt 1058
help files 128, 2185
associating with tasks 128
checking 2185
Help Files view 2218
Help Telephone 1762
Help URL 1762
Help view 1748
Home 167, 217, 252, 259, 262, 263, 264, 296, 317, 371, 372, 375, 379, 384, 407, 1063, 1687, 2072, 2141, 2244
Host setting 1476
HTTP 1526
HTTPS 1526
HyperV
connection problems 1256
Hyper-V server 1203

I

ICEs 1699, 1988
icons 538
Application Manager 538
in Application Manager 538
used on Results tab 1078
used on Selected Package List panel 1070
icons in Analyze 1815
IIS
setting Anonymous Authentication option 167
Import Conflict Options panel 1768, 1769
Import INI File Wizard 1768
Import REG File Wizard 1769
Import INI File panel 1768
Import INI File Wizard 1768
Import INI File Wizard 1767, 1768
Import Conflict Options panel 1768
Import INI File panel 1768
Welcome panel 1768
Import REG File Wizard 1768, 1769

Finishing INI File Import panel 1768
Finishing Registry Import panel 1769
Import Conflict Options panel 1769
Import Registry File panel 1769
Welcome panel 1769
Import Registry File panel 1769
Import REG File Wizard 1769
Import Tab 548
Import Wizard 784, 813, 814
Destination Group panel 813
Summary panel 814
importing 265, 289, 290, 304, 305, 306
directory service accounts 156
directory service groups 156
Marimba NCP files 265
merge modules 265, 304, 305
Microsoft security patch files 2137
MSI packages 265, 289
OS Snapshots 265, 306
OS snapshots 305
other setup types 290
specifying duplicate package identifiers in Application Manager 306
transforms 289
Windows Installer packages 265, 289
Include App-V Launcher setting 1410
Include SFT in Wrapper MSI setting 1411
including additional MSIs in virtual package 1530, 1635, 1656
Index 1182
Industry Standard Analysis 1783
Industry Standard Analysis With Auto-Fixes 1783
INI file 1767
Import INI File Wizard 1767
INI file actions 1713
modifying 1713
INI File Changes view 627, 641
INI File Exclusion Information dialog box 1025
INI file exclusions 975, 976
editing existing 975
removing existing 976
INI file keys 1713, 1715
modifying 1713
removing 1715
INI file sections 967, 974
excluding 967, 974
INI file values 1713
modifying 1713
INI files 966, 974, 1022, 1711, 1712, 1713, 1714
adding 1712
adding new keys 1713
adding sections to 1713
exclusions 966, 974, 1022
importing existing 1712
removing 1714
removing sections from 1714
INI Files tab 1022
Exclusions Editor 1022
INI Files view 1005, 1712, 1713, 1714, 1715, 1757
adding INI files in 1712
adding new INI file keys 1713
adding sections to INI files in 1713
importing existing INI files into 1712
modifying INI file keys 1713
removing INI file keys from 1715
removing INI files from 1714
removing sections from INI files from 1714
Initial Configuration Complete panel 1073, 1207
initial state 1700
setting for features 1700
initial state property 1702
Install Microsoft App-V Client option 1138
install monitoring 860
installation 1728
copying to FTP server 1728
copying to network location 1728
installing AdminStudio on a different machine 78
Installation Designer
opening 1493
installation monitoring 860
excluding processes from 919
Installation Package panel 2266
Distribution Wizard (Package) 2266
-installdir 922
InstallLocation 143
InstallScript engines 853
InstallScript MSI 895
converting 913
converting to Basic MSI with InstallScript support 895
converting to Repackager project 895
InstallScript MSI Conversion Output panel 915
Repackaging Wizard 915
InstallScript MSI Identified panel 913
Repackaging Wizard 913
InstallScript Scan 853, 895
InstallShield
about the virtualization Assistants 1491
Citrix Assistant 1547
integration of App-V Assistant 1492
integration of Citrix Assistant 1492
integration of ThinApp Assistant 1492
integration of virtualization Assistants 1492
ThinApp Assistant 1606
InstallShield 2008 Professional
compared to InstallShield Editor 1689
InstallShield Editor 956, 1683
customizing and authoring packages 1683
Dynamic Dependency Scanner 957
editing generated Repackager projects 956

features 1689
integration with Application Catalogs 1684
vs. InstallShield 2008 Professional 1689
InstallShield Editor project 938
building in Repackager 938
InstallShield log files 937
converting to Repackager project 937
inter-application conflict tests
running 1811
inter-application conflicts tests 1779
viewing results 1833
Interface 119
using AdminStudio 119
Interm directory 1054, 1612
internal consistency evaluators 1699
Internet activation for InstallShield 90
invalid Windows Installer package 1698
handling 1698
Invoke-ASConvertPackageEx 2403
Invoke-ASImportAppFromDeploymentSystem 2404
Invoke-ASImportPackage 2405
Invoke-ASPublish 2361, 2411, 2412
Invoke-ASWrapPackage 2407
-is 922, 923
isolated components 956
Isolated Components Design dialog 2169
IsolatedComponents 2170
isolating applications 2153
Isolating Windows Installer Packages Using Application Isolation Wizard 957
isolation 956, 2153, 2155, 2161, 2162, 2163
applications 2161
filtering file listings when manually configuring 2163
methods 2155
modifying default recommendations 2162
reasons not to do 2153
reasons to do 2153
Isolation Method panel 2165
Application Isolation Wizard 2165
isolation methods 2155
assemblies and manifests 2155
Windows Installer isolated components 2155
isolation options
in ThinApp Assistant 1624
inheritance from folders to files in a Citrix profile 1570
inheritance from folders to files in a ThinApp application 1626
inheritance from folders to files in an App-V package 1508
inheritance in the registry in a Citrix profile 1578
inheritance in the registry in a ThinApp application 1633
inheritance in the registry in an App-V package 1513
overview of 1568, 1623
setting for folders and files in Citrix Assistant 1569
setting for folders and files in ThinApp Assistant 1625
setting in App-V Assistant 1508
setting in Citrix Assistant 1567
setting in registry for Citrix profile 1577
setting in registry for ThinApp application 1632
setting in ThinApp Assistant 1622
Isolation Options dialog box 990
Isolation setting for a registry key 1478
isolation tests 2193, 2194
performing 2193, 2194
Isolation Tests view 2225
isrepackager.context.ini 1151
ISRepackager.ini 1033, 1035
isrepackager.ini 120
ISRIsolation 961
issnapshot.ini 818
issue resolution
automatic 1838
manual 1838

K

Keys Excluded During Registry Analysis dialog 1025
keywords 407
Keywords dialog box 679

L

LANDesk Integration Panel 2266
Distribution Wizard (Package) 2266
language requirements
setting in Citrix profile 1558
Launch Conflict Wizard button 1773
Launch Package for Testing 1097
Launcher 1163, 1226
LDAP attributes 183
directory services 183
legacy setups 853, 865, 933
converting 933
repackaging 853, 865
License Agreement dialog 1722
suppressing 1722
LimitedUI 1027
list features 1186
lists
changing column order 1188
changing columns displayed 1187
customizing 1188
grouping 1188
resizing 1188
sorting 1187
load order group 1719
setting 1719
Location panel 1770
Packaging Wizard 1770

lockdown and runtime tests 2192, 2193
performing 2192
running in restricted environments 2193
Lockdown and Runtime Tests view 2220
lockdown testing 2191
Log Monitor 1635
Log Monitor tracing options 1635
log report 1183
logging in 169
automatically 169
forgetting your password 169
login 165, 168, 169

M

Mac OS risk assessment tests 2077
Machine Import Wizard 1176
Machine property 1099, 1173
Machines tab 1170
editing virtual machine properties 1098
properties 1173
selecting columns to display 1183
shortcut menu commands 1176
viewing machine information 1172
Mandatory setting 1464
manifest
in Citrix profile 1058
Manifest and Assembly Design dialog 2168
manifest files 956
manifest options 2166
Application Isolation Wizard 2166
Manifest Options tab 2166
Advanced Options dialog 2166
Manifests 2170
manifests 958, 2157, 2172
application 958, 2157
assembly 958, 2157
checking in QualityMonitor 2188
checking with QualityMonitor
QualityMonitor
checking manifests 2188
examples of 2172
testing 2188
Manifests view 2219
Mapped Network Drive Changes go to Sandbox 1647
Marimba NCP files 265
importing 265
menus 139, 530, 986, 1020, 1180, 1736, 2204
AdminStudio 139
Application Manager 530
Application Manager context 543
Exclusions Editor 1020
QualityMonitor 2204
Repackager 986
Merge Child Keys option 1427
Merge Module Import Wizard 816
Summary panel 816
Merge Module view 634, 640
merge modules 304
importing 265, 304, 305
Merge Modules view 631
Merged isolation option 1624
Method Selection panel 909
advanced settings 919
Repackaging Wizard 909
methods 165
Microsoft ACT 1813
Microsoft ACT database
entering connection settings 259
Microsoft Application Compatibility Toolkit 1813
entering connection settings 259
Microsoft Application Virtualization (App-V). See App-V packages.
Microsoft Application Virtualization. See App-V applications.
Microsoft App-V
managing package deployment data 441, 443
specifying deployment settings 442
Microsoft App-V applications. See App-V applications.
Microsoft App-V Assistant. See App-V Assistant.
Microsoft App-V best practices tests 2000
Microsoft App-V conflict tests 2104
Microsoft App-V Packages (*.sft) 1075, 1209
Microsoft App-V Sequencer
compared with App-V Assistant 1496
Microsoft App-V Server 441, 443
specifying advanced deployment settings 442
Microsoft Hyper-V Server 508, 1063, 1072, 1096, 1205
Microsoft patches
Consolidated Patch Report 2148
performing patch impact analysis 2143
Properties dialog box 2151, 2152
viewing information in Application Manager 2145
Microsoft Problem Steps Recorder 902
Microsoft SCCM Server 1103
Microsoft security patch files
importing 2137
Microsoft security patches
importing 2135
Microsoft System Center 2007 Configuration Manager 2244, 2260
deployment data 419
publishing packages to 2244
using Distribution Wizard (Package) to prepare packages for 2244
Microsoft System Center 2012 Configuration Manager
deployment data 419
MIF 1730
instructing SMS to create file 1730
Minimum Client Version setting 1473

-mm 922, 923
mobile apps
  creating custom mobile tests 1788
mobile risk assessment tests 2012
Mobile Test Wizard 1788, 1868
-mode 922, 923
Modify button 1726
  disabling in Add/Remove Programs 1726
modifying INI file actions 1713
modifying INI file keys 1713
modifying INI file values 1713
-mp 922
-ms 922, 923
MSI command-line options 1723
  when to use Dialogs view instead 1723
MSI Doctor 2195
  install or configure products or features 2198
  reinstall components 2200
  reinstall features 2199
  verify files 2197
  view properties 2197
MSI packages 306
  building in Repackager 938
  handling invalid 1698
  importing 265
  prevalidating 1697
MSI/EXE Packages 1805
MSIs
  about repackaging MSIs 1132
MSIX 1795
MSIX conversion compatibility tests
  viewing results 1823
MSIXtests 1977
MultiUserShortcuts 1027

N

Named Objects Interaction 1164, 1229, 1474
nested .msi custom action 1036
network location 1728
  copying installation to 1728
Network Location panel 2266
  Distribution Wizard (Package) 2266
network locations 2243
  distributing packages to using Distribution Wizard (Package) 2243
New Workflow Project Wizard 129, 151, 152
  creating workflows with 129
  reference 151
  Source Package panel 152
  Target Directory and File Name panel 152
  Welcome panel 151
  Workflow Selection panel 151
New Workflow Project wizard 131
  example using 131
new workflows 129
  creating 129
New-ASCatalog 2414
New-ASDistributionConnection 2416
NewComponents 2170
notes 127
  creating for tasks 127
Notify Flexera Software App Portal on publish of current Application 402
Novell ZENworks projects 934
  .aot and .axt files 934
  application object template files 934
  converting to Repackager project 934
NT service arguments 1718
NT service dependencies 1718
  setting 1718
NT service description 1718
  setting 1718
NT service display name 1718
  setting 1718
NT service type arguments 1718
NT services 1717
NT Services view 1718, 1719, 1720, 1759
  setting the NT service arguments in 1718
  setting the NT service dependencies in 1718
  setting the NT service description in 1718
  setting the NT service display name in 1718
  setting the NT service error control level in 1719
  setting the NT service load order group in 1719
  setting the NT service overall install result in 1719
  setting the NT service start name and password in 1720
  setting the NT service start type in 1720
  setting the NT service type in 1720

O

-o 922, 923
ODBC data source attributes 1716, 1717
  adding new 1716
  editing 1716
  removing 1717
ODBC data sources 1717, 2188
  checking 2188
  removing existing 1717
ODBC Data Sources view 2219
ODBC driver attributes 1716, 1717
  adding new 1716
  editing 1716
  removing 1717
ODBC drivers 2189
  checking 2189
ODBC Drivers view 2220
ODBC resources 1715

ODBC Resources view 1715, 1716, 1717, 1758
adding new data sources 1715
adding new ODBC data source attributes 1716
adding new ODBC driver attributes 1716
editing ODBC data source attributes 1716
editing ODBC driver attributes 1716
removing existing ODBC data sources from 1717
removing ODBC data source attributes from 1717
-of 922, 923
offline activation for InstallShield 90
ON/OFF button 1837
-onp 923
Open a recent transform file option 1747
Open an existing transform file option 1748
Open Project panel 1063, 1193
Open QualityMonitor Project dialog 2210
operating system compatibility tests 1881
operating systems 1062
Optimize each test run 1775
Optimize for Offline Use 1519
Optimize for Streaming 1518
optimizing testing 1775
opt-in or opt-out of Customer Experience Improvement Program (CEIP) 145
Options dialog 144, 680, 681, 682, 683, 703, 704, 2210
in AdminStudio 144
Options dialog box 1741
Options.ini file 1027, 1033
options.ini file
OtherComponentFileExtensions 1027
support for user-defined extensions 1027
OS and browser compatibility tests
running 1794
selecting tests by selecting an OS snapshot 1783
setting auto fix preferences 1786
setting compliance level 1782
viewing results 1821, 1824
OS compatibility tests 1779
OS requirements 1015
OS security patch files
importing 2135, 2137
OS Snapshot
display limits 681
Shortcuts view 642
OS snapshot 969
exclusions 969
OS Snapshot Summary panel 817
OS Snapshot Wizard 817
OS Snapshot view 640
OS Snapshot Wizard 527, 528, 816, 817, 970
Analysis Options dialog 817
best practices 527
concept 527
configuring analysis options 528
exclusions 970
OS Snapshot Summary panel 817
performing OS Snapshots 528
Performing Snapshot panel 817
Project Information panel 816
reference 816
Welcome panel 816
OS Snapshots
importing 265, 306
OS snapshots 264, 305, 527
capturing 527
importing 305
moving in Application Manager 264
selecting OS and browser compatibility tests using 1783
taking 527
OSD file 1051
editing 1051
osguard.cp file 1464
other setup types
importing 290
OtherComponentFileExtensions 1027
OtherFilesNewComponents 1027
Output Cache Path property 1101, 1174
output directory
setting 1209
output formats 1075
Output Path 1223
Output Tab 548
Output Window 548, 1740
Import Tab 548
Output Tab 548
Package Auto Import Tab 548
Patch Impact Tab 548
Search Results Tab 548
Output window 1076, 1182
overall install result 1719
setting 1719
Override Child Keys option 1427

P

package 1697, 1698
handling invalid Windows Installer 1698
prevalidating MSI 1697
Package Auto Import Tab 548
package content 1703
configuring 1703
package conversion options 946, 1016
package data conflict tests 2082
package definition file 1728
creating 1728
Package GUID setting 1473
Package Import Wizard 1064, 1103, 1211
Package Information page 1524

Package Isolation Options dialog box 1535
Package Optimization 1226
Package Optimizations
  Optimize for Offline Use 1519
  Optimize for Streaming 1518
Package Optimizations dialog box 1538
Package property 1156
package quality 2175
Package Summary panel 1770
  Packaging Wizard 1770
Package Validation view 1697, 1748
  prevalidating a Windows Installer package 1697
Package Version setting 1473
Package view 1728, 1765
  deploying to FTP server 1728
  deploying to network location 1728
Package.DAT 1053, 1611
package.ini 1673
packages 264, 1691, 1727, 1731, 2240
  adding from a Microsoft SCCM Server 1103
  adding from an AdminStudio Application Catalog 1103
  adding from local machine or network 1107
  adding to project 1103, 1107
  copying to multiple Application Manager groups 263
  customizing 1691
  deleting from Application Catalog 264
  editing with Direct Editor 1731
  launching for testing 1137
  managing 1103
  preparing for distribution 2240
  preparing in Tuner for distribution 1727
  rules for adding from a directory 1109
  selection rules when adding from a directory 1109
  selection rules when adding packages from a directory 1200
  sharing between multiple Application Manager groups 263
Packages tab 1103, 1152
  Connect to Machine 1102
  editing package properties 1111
  editing properties 1111
  how virtual or repackaged packages are listed 1079
  icons used on 1168
  properties 1154
  selecting columns to display 1183
  Status column 1155, 1168
  viewing package information 1154
packaging 1769
  Packaging Wizard 1769
Packaging Wizard 1728, 1769, 1770
  deploying to FTP server 1728
  deploying to network location 1728
  Location panel 1770
  Package Summary panel 1770
  Setup.exe panel 1770
  SMS panel 1770
parameters 1731
  Setup.ini 1731
password 169, 1720
  forgetting your password 169
  resetting your password 169
  setting NT service 1720
patch impact analysis 2143
Patch Impact Analysis Wizard 2135, 2143, 2149
  OS Snapshots panel 2151
  Source Patches panel 2150
  Summary Information panel 2151
  Target Products panel 2151
  Welcome panel 2149
Patch Impact Tab 548
Patch Impact view 1853
patches 289
  Patches Group View 637
  Patches View 637, 639
  performing patch impact analysis 2143
  viewing information in Application Manager 2145
  viewing patch impact analysis results 2148
Patches Group View 637
Patches tab 637, 2145
  Patches Group View 637
  Patches View 639
Patches View 639
Path property 1100, 1157, 1173
Path to Executable File setting 1485
-pc 922, 924
performing OS Snapshots 528
Performing Snapshot panel 817
  OS Snapshot Wizard 817
PermanentSystemFiles 1027
PermanentSystemFilesSubfolders 1027
permissions 134, 154, 195
  AdminStudio 248
  AdminStudio client tools 193
  AdminStudio user account 249
  assigning to Application Catalog user 214
  copying roles 199
  creating new roles 198
  database server 850
  db_datareader 214
  db_datawriter 214
  deleting roles 200
  execute 214
  minimum for AdminStudio 850
  required permissions on Application Catalog databases 214
  updating roles 198
  Workflow Manager 190
Place packages under the following folder 1075, 1135
Place virtualized packages under the following folder 1209
Platform column 509, 1073, 1097
Platform property 1099, 1173

ports
port 443 (HTTPS) 77
port 80 (HTTP) 77
port 8443 77
used in activation 77
postvalidation 1696, 1698
Postvalidation view 1698, 1764
postvalidating transforms 1698
PowerShell commands 2365, 2450
-pp 922, 924
predefined folders
controlling display of in App-V Assistant 1506
controlling display of in Citrix Assistant 1566
controlling display of in ThinApp Assistant 1621
prevalidation 1696, 1697, 1698
handling invalid packages 1698
Prevalidation view 1748
primary application directory 1506
explicitly set 1507
location of shortcut in ProgramFilesFolder 1507
location of shortcut not in ProgramFilesFolder 1507
ProgramFilesFolder 1507
value of INSTALLDIR variable 1507
privacy
Customer Experience Improvement Program (CEIP) 145
private key 2161
Problem Steps Recorder 902
procedures and tasks 1700, 1704, 1705, 1706, 1707, 1708, 1709, 1710, 1711, 1712, 1713, 1714, 1715, 1716, 1717, 1718, 1719, 1720, 1721, 1722, 1723, 1724, 1725, 1726, 1727, 1745, 1747, 1748
Process Assistant 854
Process Template Editor 127, 128
adding tools from 128
reordering tasks in 127
Product Info and Update URL 1762
Product Information view 2216
product properties 1701, 1702
default destination 1701
default organization 1701
destination variable 1702
Product Properties view 1701, 1702, 1749
changing destination variable 1702
setting the default destination 1701
setting the default organization 1701
Product property 1157
Products view 636
Profile Files page 1589
creating a new folder 1565
deleting files and folders from a profile 1566
Profile Information page 1584
profile manifest file 1058
Profile Registry page 1593
Profile Requirements page 1586
Profile Shortcuts page 1591
creating a new profile shortcut 1573
excluding or deleting a profile shortcut 1574
including an existing shortcut 1574
profiles 1053, 1057, 1551
prog IDs 2186
checking 2186
Prog IDs view 2218
project files 1079
components of 1082
creating a new 1065
creating new 1065
major elements 1082
major elements used in 1081
opening 1065
opening existing 1066
setting default properties 1149
Project Information panel 816
OS Snapshot Wizard 816
Project Options dialog box 1149, 1221
project path 872, 883, 891, 901
specifying in Repackaging Wizard 872, 883, 891, 901
Project Properties dialog box 995, 997
Exclusions tab 997
General tab 995
Project Wizard 1181
ProjectName 143
projects 124, 129, 130, 131, 134, 138, 151, 152
creating 129
creating new 151
deleting 130
executing 130
filtering 130
integration with Workflow Manager workflows 138
New Workflow Project Wizard 151
running associated tools in 130
saving 131
setting global default virtual conversion settings 1236
Source Package panel 152
using 129
Projects Report 2312
Projects SLA Report 2313
Projects tab 138
ProjectTemplate 1027
properties 150, 949, 1702, 1723, 1725, 1726, 1727
adding custom setup 1726
changing Add/Remove Programs 1725
configuring tools in AdminStudio 150
editing dialog 1723
feature 1702
modifying setup 1727
removing custom setup 1727
viewing Repackager project 949
Properties Dialog 2151

Properties dialog box 1741
Microsoft patches 2151, 2152
Properties Tab
Tool Properties dialog 147
Properties tab 147
Properties window 1098
Protocol setting 1475
pseudo-tokens 1876
Publisher URL 1762
publishing
packages to Microsoft System Center 2007 Configuration Manager 2244
purchasing InstallShield 89
Purpose column 1073, 1097
Purpose property 1100, 1174
-pv 922, 924
PVKFile 2170

Q

QualityMonitor 2175, 2177, 2180, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2191, 2202, 2204, 2205, 2215
checking class IDs 2184
checking deployment status 2195
checking file associations 2185
checking help files 2185
checking manifest files 2188
checking ODBC data sources 2188
checking ODBC drivers 2189
checking prog IDs 2186
checking services 2186
checking shortcuts 2187
checking type libraries 2187
Component Properties dialog 2206
creating exclusion lists 2190
deployment testing 2183
dialogs 2205
excluding system files 2190
exclusion lists 2190
Feature Properties dialog 2207
Files dialog 2209
Install or Configure Feature Dialog 2208
Install or Configure Product Dialog 2208
install or configure products or features 2198
lockdown and runtime testing 2191
menus and toolbar 2204
MSI Doctor 2195, 2197, 2198, 2199, 2200
Product Properties dialog 2212
reinstall components 2200
Re-install Feature Dialog 2213
reinstall features 2199
Re-install Product Dialog 2213
running from the command line 2202
Test Cases 2177
test reports 2202
verifying files 2197
viewing deployment status properties 2197
viewing Test Item details 2180
views 2215
QualityMonitor projects 2176, 2177
creating new 2176
opening existing 2177

R

reboot handling 924
Reboot Message setting 1487
reference 135, 151, 816, 1151
AdminStudio Interface 135
New Workflow Project Wizard 151
OS Snapshot Wizard 816
REG files 1708
importing 1708
Registry 1166, 1229
registry 976, 977, 1023, 1768
exclusions 976, 1023
Import REG File Wizard 1768
modifying in App-V Assistant 1512
modifying in ThinApp Assistant 1631
removing exclusions 977
registry entries 1706
Registry Entries view 1003
registry exclusions 976, 977
editing existing 977
removing global 977
registry information 1708
removing 1708
Registry Isolation Options dialog box 1536, 1602, 1662
registry keys 966, 1706
and App-V package 1512
and Citrix profile 1577
and ThinApp application 1632
creating 1706
excluding 966
Registry page 1651
Registry tab 1023
Exclusions Editor 1023
registry values 966, 1707
creating 1707
excluding 966
Registry view 627, 641, 1706, 1707, 1708, 1752
creating a registry key in 1706
creating a registry value in 1707
importing REG files in 1708
removing registry information from 1708
reinstallation 78
remote application publishing compatibility test results 1828
remote application publishing compatibility tests 2108

remote desktop services tests 2113
Removable Disk Changes go to Sandbox 1647
Remove 1176
Remove button 1726
  disabling in Add/Remove Programs 1726
Remove Selected 1167
Remove_ASPackage 2424
Remove-ASApplication 2422
Remove-ASGroup 2423
reordering 127
  tasks 127
Repack.ini file 1035
Repack.log 1033
Repackage 1178
Repackaged Output view 938, 1008, 1012
  building InstallShield Editor projects 938
  building MSI packages 938
  creating virtual applications 1043
Repackager 853, 859, 862, 922, 928, 929, 932, 933, 934, 937, 946, 949, 950, 951, 955, 956, 964, 965, 966, 967, 968, 969, 970, 985, 986, 988, 989, 995, 998, 999, 1001, 1003, 1004, 1005, 1006, 1007, 1008, 1012, 1015, 1017, 1019, 1027, 1036, 1037
  About Repackager dialog 988
  additional setup programs 870
  Advanced Settings view 1015
  and anti-virus software 861
  automatically building a Citrix profile 947
  automatically building a ThinApp application 947
  automatically creating a Windows Installer package after creating the Editor project 948
  automatically running best practice tests upon build 943, 1011
  building a Citrix profile 938, 1012, 1055
  building a ThinApp applications 1052
  building a VMware ThinApp application 1011
  building an App-V package 1011
  building Citrix profile automatically at project build 947
  building InstallShield Editor projects 938
  building MSI packages 938
  building ThinApp application automatically at project build 947
  Captured Installation view 999
  changing data type appearance 929
  clean systems 859
  command-line options 922
  component settings options 947, 1017
  configuring advanced conversion options 946
  configuring exclusions 964
  conversion problems 1036, 1037
  converting .axt 934
  converting .inc 933
  converting .ipf 934
  converting .isl 937
  converting .txt 937
  converting .wse 937
  converting InstallShield log files 937
  converting legacy setups 933
  converting Novell ZENworks projects 934
  converting Repackager 3.x output 933
  converting SMS projects 934
  converting WinINSTALL projects 937
  converting Wise Installation projects 937
  Create Report dialog box 989
  creating projects 932
  creating reports 951
  Deleted Files view 1006
  Deleted Registry Entries view 1007
  detecting dependencies 950
  dialog boxes 988
  editing generated projects in InstallShield Editor 956
  Error Building Table File error 1037
  excluding all files in a directory 965
  excluding all shortcuts in a directory 967
  excluding directories and subdirectories 965
  excluding files 964
  excluding INI file sections 967
  excluding INI files 966
  excluding registry keys 966
  excluding registry values 966
  excluding shortcuts 967
  excluding shortcuts from subdirectories 967
  exclusion list 1033
  exclusions 970
  file exclusions 964
  Files and Folders view 1001
  files associated with 1033
  INI Files view 1005
  installing on a clean machine 864
  installing on clean machine 864
  ISDEV fatal error -5023 1037
  isolating applications 2155
  Isolation Options dialog box 990
  launching 928
  launching Automated Application Converter 1063
  launching remotely 862
  location of isrepackager.ini 120
  menus 986
  modifying external configuration file 969
  Options dialog box 992
  Options.ini file 1027
  OtherComponentFileExtensions 1027
  package conversion options 946, 1016
  Package Information view 1012
  Project Properties dialog box 995
  Registry Entries view 1003
  remote repackaging on Windows NT4 862
  Repackaged Output view 1008, 1043

repackaging legacy setups 865
running automated best practice tests against the .msi package at build 943, 1011
saving projects 955
setting digital signature options 990
setting manifest options 990
Setup Intent Wizard 1017
Shortcuts view 1004
SMS conversion problems 1037
specifying external configuration file 968
support of user-defined extensions in options.ini 1027
ThinApp applications 1052
toolbar 986
troubleshooting 1036, 1037
viewing properties 949
views 998
virtualization 1011, 1012
VMware Repackaging Wizard 1019
WinINSTALL Conversion dialog box 998
WinINSTALL conversion problems 1036
Repackager 3.x output 933
converting to Repackager project 933
Repackager Cache Path property 1101, 1174
Repackager error building table file 1037
Repackager projects 932, 955
creating 932
saving 955
Repackager Start Page 985
repackaging 853, 860, 862, 865, 969, 1133
a Windows Installer package 898
alternate-language 860
an installation from a self-extracting .exe 868
and anti-virus software 978
excluding processes from 919
exclusions 969
legacy setups 853, 865
remotely on Windows NT4 862
support for 64-bit applications 855
viewing conversion results 1136
Windows Installer packages 1132
Repackaging an InstallScript MSI Setup to a Basic MSI Setup 895
Repackaging Method property 1158
repackaging methods 860
selecting 876
Repackaging panel 915
Repackaging Using the Installation Monitoring Method 866
Repackaging Using the Snapshot Method 876
Repackaging Wizard 862, 897, 908, 909, 910, 912, 913, 914, 915, 917, 918, 919, 920, 924
additional panels 918
additional setup programs 870
Additional Setup Programs dialog box 918
Analysis Options dialog box 920
Collect Product Information panel 912
deleting added setups 871
Excluded Processes dialog box 919
InstallScript MSI Conversion Output panel 915
InstallScript MSI Identified panel 913
Method Selection panel 909
reboot handling 924
repackaging a Windows Installer package 898
Repackaging panel 915
running from command line 897
running remotely on Windows NT4 862
Set Target Project Information and Capture Settings panel 914
Setup Information dialog box 919
Snapshot Method panel 910
specifying project path 872, 883, 891, 901
Summary panel 917
using 866
Welcome panel 908
Repackaging Wizard Command-Line Options 922
Repacking Wizard
using documentation tool 902
Repair button 1726
disabling in Add/Remove Programs 1726
report 1183
viewing 1136
Reports 2327
queries 2310
searches 2310
use of wildcard character in searches 2310
reports 951, 2327
Activity Report 2318
All Reports page 2327
assigning Role permission to view 2355
creating 2327
creating custom reports 2316
creating in Repackager 951
Custom Report 2317
Custom SQL Query Report 2322
Custom Stored Procedures Report 2304, 2323
editing Role permissions to view 2355
export formats 2309
exporting to Microsoft Excel 2309
filtering 2309
generating 2309
Projects Report 2312
Projects SLA Report 2313
Report Wizard 2348
Reports 2327
viewing 2309
Workflow Requests Summary Report 2312
Reports tab
in Application Manager 361
Reports Wizard 2348
requirements

how they are applied for a Citrix profile 1559
setting language requirements in Citrix profile 1558
Reset Period setting 1487
Reset Sandbox on Exit 1647
Resolve Issues button 1773, 1838
Resolve-ASPackage 2425
resolving conflicts 1838
response transforms 1695
restricted environments 2193
running lockdown and runtime tests 2193
results
viewing 1136
Results panel 1018
Setup Intent Wizard 1018
Results tab 1076, 1136, 1176
icons used on 1078, 1178
properties 1177
report 1183
shortcut menu commands 1179
ribbon
Application Manager 530
risk assessment tests 1779, 2012
Role
assigning permission to view reports 2355
Role Administration page 201
Role Details page 203
roles 154, 198, 201
copying 199
creating 198
deleting 200
system roles 195
updating 198
user roles 195
Root Folder Mapping setting 1473
Root Folder Name setting 1474
RTSP 1475, 1526
RTSPS 1475, 1526
Rules Viewer dialog 1859, 1860
Rules Wizard 1872, 1873, 1874, 1875, 1876, 1877
Additional Information panel 1874, 1875
DLL-Based ACEs panel 1877
General Information panel 1873
inserting Column Names in Error or Display Strings 1876
inserting Product Name in Error or Display strings 1876
pseudo-tokens 1876
Summary panel 1877
Token Grammar 1876
Welcome panel 1873
Where Clause Panel 1877
running associated tools 130
runtime drive 1228
runtime testing 2191

S

sample.mdb 230
sandbox cache 1613, 1614
Sandbox Name 1647
saving 131
projects 131
workflows 131
-sb 922, 924
Scan for Dependencies 624
Application Manager 624
Scanning project panel 1018
Setup Intent Wizard 1018
Script Execution dialog box 1597
script-based setup.exe 2250
creating with Distribution Wizard (Package) 2250
Search 1182
Search Results Tab 548
Second Action Delay setting 1487
Second Error setting 1487
Select Output Formats panel 1074, 1134, 1208
Select Package Installation File dialog box 1108, 1230
Select Package Source panel 1068, 1103, 1107, 1194
Select Packages Installation File dialog box 1230
Select Packages panel 1069, 1105, 1108, 1196
Select Tests to Execute button 1772, 1778
Select Transform dialog box 1231
Select Virtual Machine dialog box 1138, 1139, 1145, 1147, 1232
Select Virtual Machine Image File dialog box 1233
Select Virtual Machine Source panel 507, 1071, 1095, 1202
Select Virtual Machines from a Microsoft Hyper-V Server panel 1203
Select Virtual Machines from a VMware ESX or ESXi Server panel 1204
Select Virtual Machines panel 509, 1072, 1096, 1205
Selected Package List panel 1069, 1105, 1109, 1198
icons used on 1070
sequences 1721
restoring dialog 1721
serial number activation for AdminStudio 83
Server Address property 1102, 1174
Server Host 1227
server locations 1724, 1725
adding 1724
modifying 1724
removing 1725
reordering 1725
Server Locations view 1724, 1725, 1761
adding server locations in 1724
modifying server locations in 1724
removing server locations in 1725
reordering server locations in 1725
Server Name 1196, 1204
Server Password property 1102, 1174

Server Port 1227
Server Protocol 1228
Server Username property 1102, 1174
Service Dependencies setting 1486
Service Is Interactive setting 1485
Service Packs Requirement dialog box 1604
service type 1720
  setting 1720
Service Type setting 1485
ServiceControlEvents 1027
services 2186
  and Citrix profiles 1325
  and Citrix XenApp 1326
  checking 2186
Services view 2220
Set Target Project Information and Capture Settings panel 914
  Repackaging Wizard 914
Set-ASConfigPlatform 2427
Set-ASProperty 2432
Set-ASSoftwareRepository 2439
Set-ASTestState 2441
Setup Cache Path property 1101, 1174
Setup Information dialog box 919
  Repackaging Wizard 919
Setup Intent Wizard 950, 1017, 1018
  Results panel 1018
  Scanning project panel 1018
  Welcome panel 1018
Setup Organization 1699
setup programs
  adding additional in Repackaging Wizard 870
setup properties 1726, 1727
  adding custom 1726
  customizing 1726
  modifying 1727
  removing custom 1727
Setup Properties view 1726, 1727, 1761
  adding and editing comments 1726
  adding custom setup properties 1726
  modifying properties 1727
  removing custom setup properties from 1727
setup property comments 1726
  adding and editing 1726
Setup.exe 1731
  creating for package and transform 1731
Setup.exe panel 1770
  Packaging Wizard 1770
Setup.ini 1731
setups 853, 1722
  deleting added in Repackaging Wizard 871
  disabling custom 1722
  repackaging legacy 853, 865
shared assemblies 2164
  servicing published 2164
shared location 120
SharedCommonFiles 1027
SharedPoint 143
sharing
  packages 263
  packages between multiple Application Manager groups 263
Shell New Enabled setting 1467
shortcut menus 543
  Application Manager 543
shortcuts 967, 1708, 1709, 1710, 1711, 1753, 1755, 1756, 2187
  changing icon for 1709
  changing location for 1710
  changing target for 1710
  checking 2187
  creating 1709
  creating a hot key for 1710
  determining path of changed in Tuner 1711
  excluding 967
  excluding all in a directory 967
  excluding from subdirectories 967
  excluding or deleting from App-V package 1510
  excluding or deleting from Citrix profile 1574
  excluding or deleting from ThinApp application 1629
  including in App-V package 1510
  including in Citrix profile 1574
  including in ThinApp application 1629
  location 1756
  properties 1753
  removing 1711
  target 1755
  when to exclude or delete from Citrix profile 1575
  when to exclude or delete from ThinApp application 1630
Shortcuts view 629, 1004, 1709, 1710, 1711, 1753, 2218
  changing a shortcut's icon 1709
  changing a shortcut's location 1710
  changing a shortcut's target in 1710
  creating a hot key for a shortcut 1710
  creating shortcuts in 1709
  OS Snapshot 642
  removing shortcuts from 1711
silent activation 77
single sign-on 169
SISAuthor=Repackager 1027
SMS 1729
  creating file 1729
  deploying 1730
SMS conversion problems 1037
  Repackager 1037
SMS panel 1770
  Packaging Wizard 1770
SMS project 934
  converting to Repackager project 934
-sn 922, 924
Snapin 2360

snapshot 860
Snapshot method
  and anti-virus software 861
Snapshot Method panel 910
  Repackaging Wizard 910
Snapshot Name property 1100, 1175
snapshots 1092
Soft Time-Out property 1158
SoftGridUserSettings folder 1422
software publishing credentials 2160
Software Repository 525
  and Distribution Wizard 525
  and InstallShield Editor 525
  and Virtual Package Editor 525
  introduction 525
  using 525
sorting lists 1187
Source Package panel 152
  in New Workflow Project Wizard 152
SourcePackage 143
SPCFile 2170
Specifying duplicate package identifiers in Application Manager 306
SQL Server
  permissions required by AdminStudio user account 249
standalone Application Catalog 213
Standard.nir 1033
start name 1720
  setting NT service 1720
Start Page 136
start type 1720
  setting NT service 1720
Start-ASConversion 2444
Startup Type setting 1485
status icons 1815
Steps Recorder 902
Subsequent Action Delay setting 1488
Success Result setting 1484
suiting
  in App-V Assistant 1514
Summary panel 814, 917, 1867, 1877, 2165
  Application Isolation Wizard 2165
  Conflict Wizard 1867
  Import Wizard 814
  Repackaging Wizard 917
  Rules Wizard 1877
suppress button 1837
suppressing issues 1837
suppression issues 1723
SuppressReboot 1731
SuppressWin2k 1731
Symantec Altiris Management Server
  distributing applications to 1148, 2232
Symantec Altiris Server 451
  Altiris Deployment Data tab 607
  managing package deployment data 451
  specifying package deployment settings 452
  support for 233
SysGuard File setting 1464
System Center 2007 Configuration Manager
  support for 233
System Center 2012 Configuration Manager
  distributing applications to 1148
  permissions required by AdminStudio 249
  support for 233
System Configuration view 1751
  Tuner 1751
system roles 195

T

table 1732
  adding a new record using Direct Editor 1732
  adding a new row using Direct Editor 1732
Tables view 633, 642
taking OS snapshots 527
target directory 152
  setting in New Workflow Project Wizard 152
Target Directory and File Name panel 152
  in New Workflow Project Wizard 152
Target Information panel 1867
  Conflict Wizard 1867
Target setting 1480, 1481
TargetDir 143, 152
TargetFileName 143, 152
task page help 122
  setting location in AdminStudio 122
task properties 126
  modifying 126
tasks 126, 127, 128, 139
  associating help files 128
  creating new 126
  creating notes for 127
  deleting 128
  in workflows 139
  renaming 127
  reordering in Process Template Editor 127
templates 406
Terminate Children setting 1480
Test Cases 2177, 2179, 2181, 2182, 2200
  adding comments 2179
  clearing results 2181
  creating custom 2200
  filtering data 2182
  manually setting status 2181
Test Configuration Wizard 1782, 1861
Test Cycle Summary view 2216
Test Item details 2180

viewing 2180
Test Item Information dialog 2214
Test Items 2177, 2178, 2179, 2182
adding comments 2179
manually setting status 2182
running individual 2177
running multiple 2178
Test Progress dialog 2214
test reports 2202
Test Result dialog 2215
test results
filtering 1837
Test Virtualization Readiness 1181
Test-ASConflicts 2445
Test-ASPackage 2446
testing 2183, 2191
deployment 2183
lockdown and runtime 2191
repackaged MSI packages 1137
selecting tests to execute 1778
source packages 1137, 1147
virtual packages 1137
testing packages 1137
ThinApp 947, 1052, 1055
AppLink 1664
Interm directory 1054
Virtual Operating System 1053
ThinApp applications 947, 1052
about 1052, 1607, 1611
adding an AppLink reference 1638
adding diagnostic tools to 1616, 1659
adding existing folder 1619
adding files to 1618
adding or deleting registry keys and values 1632
and Active Directory 1615
Application Link 1637
AppLink 1637, 1664
AppSync 1639
automatically creating from Repackager 1043
benefits of deploying 1055, 1607
building 1642
building a Windows Installer package with build output 1635
building using command line 1673
components of 1053, 1611
compressing 1627
configuration file (package.ini) 1673
controlling access to 1615
conversion error and warning messages 1673
creating using InstallShield ThinApp Assistant 1042
creating with InstallShield 1606
defining shortcut executables 1627
excluding vs. deleting a shortcut 1630
files included in 1053, 1611
how transforms are included 1612
including an existing shortcut 1629
inheritance of isolation options from folders to files 1626
inheritance of isolation options in the registry 1633
Interm directory 1054, 1612
intermediate data files 1054, 1612
launching 1611
linking ThinApp applications 1637, 1664
linking to an application with more than one shortcut 1668
linking to an application with one shortcut 1668
linking to another ThinApp application 1668
location of 1611
managing files and folders 1617
methods to convert Windows Installer packages 1041
modifying registry 1631
moving files and folders 1620
operating system 1607
order of import of linked ThinApp applications 1666
overview of isolation options 1623
overview of ThinApp Assistant 1606
Package.DAT 1053
Package.DAT file 1611
prerequisites for building 1055, 1616
relative paths when defining AppLink references 1638
renaming a shortcut 1630
sandbox cache 1614
sandbox name 1614
sandboxes 1613
security and authorization for linked applications 1667
selecting application shortcuts 1626
Setting AppLink options 1637
setting AppSync Expiration settings 1640
setting folder isolation options 1660
setting isolation options 1622, 1624
setting isolation options for folders and files 1625
setting Log Monitor tracing options 1635
setting registry isolation options 1632
shortcut requirements 1628, 1651
shortcuts 1053
steps to create 1613
steps to create with ThinApp Assistant 1608
ThinApp Assistant 1606
ThinAppPackage directory 1611
updating applications 1639
updating using AppSync 1669
virtual environment 1607
virtual operating system 1053
Virtual Operating System (VOS) 1607
ThinApp Assistant 948, 1042, 1606, 1673
about 1608
about sandboxes 1613
adding an existing folder to a ThinApp application 1619
adding diagnostic tools to a ThinApp application 1616, 1659
adding files to a ThinApp application 1618
adding or deleting registry keys and values from a ThinApp

application 1632
application features requiring pre- or post-conversion actions 1673
Applications page 1650
Build Options page 1653
building a ThinApp application 1642
building a Windows Installer package with build output 1635
building ThinApp application in Direct Edit mode 1634
clearing the cache 1658
Compression Type option 1627
compression types 1627
controlling access to ThinApp applications 1615
controlling the display of predefined folders 1621
creating a ThinApp application 1613
creating new application shortcut executables 1628
creating new folder 1620
default isolation options 1625
deleting files and folders 1621
Diagnostic Tools dialog box 1616, 1659
disabling Log Monitor Tracing 1637
enabling ThinApp application building when editing a Windows Installer package 1634
error messages 1673
excluding or deleting a ThinApp application shortcut 1629
Files & Folders page 1648
Home page 1644
how transforms are included 1612
including an existing shortcut 1629
inheritance of isolation options 1626
inheritance of isolation options in registry 1633
integration with Project Assistant and Installation Designer 1492
Interm directory 1612
isolation options 1624
managing files and folders 1617
modifying build options 1633
modifying registry settings 1631
modifying shortcuts 1626
moving files and folders 1620
overview 1606
overview of 1606
overview of isolation options 1623
Package Information page 1646
package.ini 1673
prerequisites for creating ThinApp application 1616
reference 1643
Registry Isolation Options dialog box 1662
Registry page 1651
renaming a shortcut 1630
sandbox cache 1614
selecting releases to build 1634
setting isolation options 1622, 1624
setting isolation options for folders and files 1625
setting isolation options on folders 1660
setting Log Monitor tracing options 1635
setting registry isolation options 1632
specifying access via Active Directory 1615
specifying general settings 1614
steps to create a ThinApp application 1608
supported InstallShield project types 1612
when to exclude and when to delete shortcuts 1630
Write Copy, Merged, and Full isolation options 1624
ThinApp cache
clearing 1658
ThinApp Log Monitor 1635
ThinApp Virtual Operating System (VOS) 1607
ThinApp Virtualization Suite Not Found 1673
ThinApp VOS 1053
ThinAppPackage directory 1611
Timeout setting 1484
TimeStampAssemblies 2170
Token Grammar 1876
Rules Wizard 1876
Tool Properites panel 150
tool properties 147
viewing 147
Tool Properties dialog 147, 148
Configuration tab 148
Properties tab 147
Tool Properties panel 150
Add Tool Wizard 150
toolbar 139, 530, 986, 1736, 2204
AdminStudio 139
Application Manager 530
QualityMonitor 2204
Repackager 986
toolbars 1180
Toolbars tab 1741
tools 128, 130, 147, 150
adding from Process Template Editor 128
adding in AdminStudio 150
adding using the Add Tool Wizard 150
running associated in projects 130
viewing properties 147
Tools menu 1181
Total File Size setting 1474
transform 1698, 1731
creating Setup.exe for 1731
postvalidating 1698
transform properties 1696
viewing 1696
Transform property 1158
Transform Summary dialog 1742
transforms 1692, 1693, 1694, 1695, 1742
creating generic 1695, 1742
creating new 1693
creating universal 1695, 1742
generic 1695, 1742

importing 289
opening existing 1694
opening recently accessed 1695
response 1695
Tuner 1692
universal 1695, 1742
working with 1692
trial mode 1083
troubleshooting 1247
first things to check 1247
how to test a virtual machine 1255
problems and solutions 1250
Tuner 1691, 1692, 1696, 1705, 1721, 1727, 1731, 1732, 1765, 1766
adding and editing setup property comments 1726
adding custom setup properties 1726
adding files 1704
adding INI files 1712
adding new data sources 1715
adding new INI file keys 1713
adding new ODBC data source attributes 1716
adding new ODBC driver attributes 1716
adding sections to INI files 1713
adding server locations 1724
CAB files 1705
changing a feature's visibility 1700
changing a shortcut's icon 1709
changing a shortcut's location 1710
changing a shortcut's target 1710
changing Add/Remove Programs Properties 1725
creating a hot key 1710
creating a new transform file 1745
creating a registry key 1706
creating a registry value 1707
creating shortcuts 1709
Direct Editor 1732, 1766
disabling custom setups 1722
displaying files from the base Windows Installer package 1704
editing dialog properties 1723
editing ODBC data source attributes 1716
editing ODBC driver attributes 1716
hiding dialogs during UI sequences 1721
importing existing INI files 1712
importing REG files 1708
modifying server locations 1724
modifying setup properties 1727
opening a recent transform file 1747
opening an existing transform file 1748
preparing packages for distribution 1727
preventing installation of files from the Windows Installer package 1705
removing added files 1705
removing custom setup properties 1727
removing existing ODBC data sources 1717
removing INI file keys 1715
removing INI files 1714
removing ODBC data source attributes 1717
removing registry information 1708
removing sections from INI files 1714
removing server locations 1725
removing shortcuts 1711
reordering server locations 1725
restoring dialog sequences 1721
setting initial state of a feature 1700
setting the NT service arguments 1718
setting the NT service dependencies 1718
setting the NT service description 1718
setting the NT service display name 1718
setting the NT service error control level 1719
setting the NT service load order group 1719
setting the NT service overall install result 1719
setting the NT service start name and password 1720
setting the NT service start type 1720
setting the NT service type 1720
Setup.ini 1731
suppressing License Agreement dialog 1722
transforms 1692
validation in 1696
working with dialogs 1721
Tuner reference 1736, 1739, 1740, 1741, 1744, 1745, 1747, 1748
checklist 1739
customization steps 1740
Customize dialog box 1741
Help view 1748
InstallShield Start page 1745
menus 1736
Options dialog box 1741
Output Window 1740
Package Validation view 1748
Properties dialog box 1741
toolbar 1736
user interface reference 1736
View Bar 1739
views 1744
Tuner views 1744, 1745, 1748, 1749, 1750, 1751, 1752, 1753, 1757, 1758, 1759, 1761, 1762, 1763, 1764, 1765, 1766
Add/Remove Programs 1762
Additional Tools 1766
Application Configuration 1761
Dialogs 1762
Direct Editor 1766
Features 1750
Files and Folders 1751
Help 1748
INI Files 1757
Location 1765

NT Services 1759
ODBC Resources 1758
Package 1765
Package Preparation 1763
Package Validation 1748
Postvalidation 1764
Prevalidation 1748
Product Properties 1749
Registry 1752
Server Locations 1761
Setup Properties 1761
Setup.exe 1765
Shortcuts 1753
SMS 1765
System Configuration 1751
type libraries 2187
checking 2187
checking with QualityMonitor 2187
Type Libraries view 2218

U

UI sequences 1721
hiding dialogs during 1721
uninstalling
InstallShield 100
universal transforms 1695, 1742
creating 1695, 1742
Updated.isr 1033
upgrading
5.0 or 5.5 or 6.0 Application Catalogs 229
upgrading AdminStudio databases 229
Use Application Expiration 1671
UseHKCUProxy 1027
UseMergeModules 1027
user accounts 154
User Credentials panel 509, 1073, 1097, 1207
user interface 1152
user interface reference 1736
user roles 195
User-Defined Tests view 2225
Test Case view 2226
users. See accounts.
UseSrcFolder 1027
using accounts 168
using domain accounts 168

V

validation 1696, 1787
changing the default file in Application Manager 1787
Validation tab 1732
launching Direct Editor 1732
variables 143
DevLocation 143
InstallLocation 143
ProjectName 143
SharedPoint 143
SourcePackage 143
TargetDir 143
TargetFileName 143
Vendor Command Line Switches 360
verifying component deployment 2195
verifying files 2197
using MSI Doctor in QualityMonitor 2197
-version 924
version 922
Version GUID setting 1473
version management
using the Software Repository 525
Version property 1157
Versioning 1226
VFS folder 1422
View Bar 1739
View menu 1180
View Report 1181
View Settings tab 1741
viewing tool properties 147
Views 998
views 624, 1744, 1745, 1748, 1749, 1750, 1751, 1752, 1753, 1757, 1758, 1759, 1761, 1762, 1763, 1764, 1765, 1766
Additional Tools 1766
Dependencies 624
Dialogs 1762
Direct Editor 1766
Features 1750
Files and Folders 1751
Help 1748
INI Files 1757
Location 1765
NT Services 1759
ODBC Resources 1758
Package 1765
Package Preparation 1763
Package Validation 1748
Postvalidation 1764
Prevalidation 1748
Product Properties 1749
Registry 1752
Server Locations 1761
Setup Properties 1761
Setup.exe 1765
Shortcuts 1753
SMS 1765
virtual file system (VFS) folder 1422
virtual file system folder 1422
virtual machine
how to test 1255

importing 1095
Virtual Machine Import Wizard 1064, 1213
using 1095
Virtual Machine Platform 1076, 1135
Virtual Machine Preparation Tool 1247
virtual machines
connecting to active machines 1102
editing virtual machine properties 1098
how Automated Application Converter selects the machines to use 1136
list of supported 1063
managing 1083
preparing virtual machines for use with the Automated Application Converter 1086
preparing with Virtual Machine Preparation setup 1092
taking a snapshot 1092
VMware VIX API requirement 1094
Virtual Package Editor 1405
associating targets with a dependency 1419
Cmd.exe 1447
configuring file extension associations 1434
configuring virtual services 1444
creating shortcuts 1430
debug tools 1447
defining targets 1429
editing the virtual registry 1426
editing tips 1413
entry points 1429
extracting files from an App-V package 1425
HREF scripts 1439
including files and folders 1422
opening a virtual package in 1409
Regedit 1447
save options 1410
saving as a new package 1410
saving as an upgrade 1410
SCRIPTBODY scripts 1439
setting environment variables 1432
specifying the application path 1442
starting 1409
using the App-V Application Launcher 1445
viewing package history 1417
virtual packages
manually associating with Windows Installer package 287
virtual services in an App-V package 1444
virtualization 938
about 1489
benefits of 1044, 1045, 1490
benefits of Citrix XenApp 1058
building a Citrix profile using Repackager 1055
building a ThinApp application using Repackager 1052
cancelling 1181
capturing context 1150
context file 1150
diagram of 1490
example diagram 1044
example of 1490
getting started 1041
in InstallShield 1491
including additional MSIs 1530, 1635, 1656
methods to convert Windows Installer packages 1041
overview of 1044
overview of Citrix profiles 1053, 1057, 1551
overview of Citrix XenApp 1056, 1549
viewing conversion results 1136
virtualization and Windows Installer best practices tests
running 1794
viewing results 1830, 1832
virtualization Assistants
integration with Project Assistant and Installation Designer 1492
navigating in 1493
opening the Installation Designer 1493
showing and hiding 1493
Virtualization not recommended 1071, 1106, 1110, 1169
Virtualization not supported 1169
Virtualization Readiness 1199
Virtualization Readiness column 1070, 1106, 1109
Virtualization Readiness property 1156
Virtualization Technology property 1101, 1157, 1175
Virtualize packages with detected settings 1073
virtualizing 1133, 1134
visibility 1702
changing for features 1700
feature 1702
visible property 1702
visiblity 1700
VIX API 1094
VMware 1019
ThinApp application 1011
VIX API requirement 1094
VMware ESX or ESXi Server 508, 1063, 1072, 1096, 1203, 1205
taking a snapshot 1093
VMware Repackaging Wizard 1019, 1020
VMware Virtual Machines panel 1020
Welcome panel 1020
VMware ThinApp 1011
VMware ThinApp application 1011
VMware ThinApp Packages (*.exe) 1075, 1209
VMware Virtual Machines panel 1020
VMware Repackaging Wizard 1020
VMware VIX API 1094, 1248
VMware Workstation 1063
setting Snapshots options 1094
taking a snapshot 1093

W

Warning Frequency 1672
Warning Message 1672
Warning Period 1671
Warnings property 1177
watermark 80
web deploy best practices tests 2010
Welcome panel 150, 151, 152, 908, 1018, 1020, 1768, 1769, 2164, 2259
  Add Tool Wizard 150
  Application Isolation Wizard 2164
  Distribution Wizard (Package) 2259
  Import INI File Wizard 1768
  Import REG File Wizard 1769
  in New Workflow Project Wizard 151
  OS Snapshot Wizard 816
  Patch Impact Analysis Wizard 2149
  Repackaging Wizard 908
  Setup Intent Wizard 1018
  VMware Repackaging Wizard 1020
Welcome to InstallShield Tuner 1745
Welcome to Tuner view
  Create a new transform file option 1745
Where Clause Panel 1877
  Rules Wizard 1877
Where Clause tab 1857
Windows 2000 1762
  Add/Remove Programs 1762
Windows Installer File Selection panel 2165
  Application Isolation Wizard 2165
Windows Installer Internal Consistency Evaluators (ICEs) 1988
Windows Installer isolated components 2155
  isolation method 2155
Windows Installer package 1697, 1698
  and .context.msi 1150
  building in Repackager 938
  handing invalid 1698
  prevalidating 1697
Windows installer package
  repackaging using the Repackaging Wizard 898
Windows Installer packages
  about repackaging 1132
  importing 265, 289
  reasons to avoid repackaging 1132
Windows Installer Packages (*.msi)
  specifying as an output format 1075
Windows registry
  and App-V Assistant 1512
  and Citrix Assistant 1576
  and ThinApp Assistant 1631
Windows risk assessment tests 2080
Windows Services
  integration into an App-V application 1052
Windows services
  App-V support of 1052, 1500
  running within the virtual environment 1052, 1500
Windows XP 1762
  Add/Remove Programs 1762
WindowsNTServiceMSIX002 1978
WinINSTALL Conversion dialog box 998
WinINSTALL conversion problems 1036
  Repackager 1036
WinINSTALL project 937
  converting to Repackager project 937
Wise Installation project 937
  converting to Repackager project 937
wizards 150, 151, 1192
  AdminStudio Interface 150
  Application Isolation Wizard 2166
  Finishing INI File Import panel in Import REG File 1768
  Finishing Registry Import panel in Import REG File 1769
  Import Conflict Options panel in Import INI File 1768
  Import Conflict Options panel in Import REG File 1769
  Import INI File 1767
  Import INI File panel in Import INI File 1768
  Import REG File 1768
  Import Registry File panel in Import REG File 1769
  Location panel in Packaging 1770
  New Workflow Project Wizard 151
  Package Summary panel in Packaging 1770
  Packaging 1769
  Patch Impact Analysis wizard 2149
  Setup.exe panel in Packaging 1770
  SMS panel in Packaging 1770
  Welcome panel in Import INI File 1768
  Welcome panel in Import REG File 1769
workflow 131
  example using the New Workflow Project wizard 131
Workflow Manager 138, 475
  and extended attributes 475
  integration with AdminStudio projects 138
  permissions 154
  roles 154
  types of users 154
workflow requests
  activities recorded in database 2318
Workflow Requests Summary Report 2312
Workflow Selection panel 151
  in New Workflow Project Wizard 151
workflow tasks 139
workflows 124, 125, 126, 129, 131, 134, 138, 139, 151, 152
  creating 124
  creating new 125, 129
  deleting 126
  editing 124
  filtering 125
  New Workflow Project Wizard 151, 152

renaming 125
saving 131
tasks 139
Workflows Templates tab 138
Wrap MSI/EXE Packages 1805
Write Copy isolation option 1624

X

XenApp
specifying deployment settings 448
XenApp Server 447
specifying advanced deployment settings 450
XenApp. See Citrix XenApp
XML file 473
as extended attributes description 473

Z

ZENworks Configuration Management 2246, 2268
preparing for distribution 2246
ZENworks Configuration Management Server Login 2268
ZENworks projects
.aot and .axt files 934
application object template files 934
converting using Repackager 934