

# **Spider Technische Referenz**

**Installation, Konfiguration, Fehlerbehebung**

# **spider**

**Ausgabe 2022-01-17**

Copyright © 2020-2022. Flexera Software LLC. Alle Rechte vorbehalten.

Copyright © 1997-2019. Brainwaregroup. Alle Rechte vorbehalten.

Jedwede von Flexera zur Verfügung gestellte Dokumentation unterliegt dem Urheberrecht und ist Eigentum von Flexera. Flexera übernimmt weder Garantie noch die juristische Verantwortung oder irgendeine Haftung für die Nutzung dieser Informationen, für deren Wirtschaftlichkeit oder fehlerfreie Funktion für einen bestimmten Zweck.

Bei Zusammenstellung dieses Dokuments wurde jede Anstrengung unternommen, die Richtigkeit des Inhalts sicherzustellen. Flexera übernimmt jedoch keine Garantie hinsichtlich dieser Dokumentation und keine gesetzliche Gewährleistung für die marktgängige Qualität und Eignung für einen bestimmten Zweck. Des Weiteren übernimmt Flexera keine Haftung für Fehler oder unbeabsichtigte Schäden bzw. Folgeschäden im Zusammenhang mit der Bereitstellung, Leistung oder Verwendung dieses Dokumentes oder der darin enthaltenen Beispiele. Flexera behält sich das Recht vor, diese Dokumentation ohne vorherige Ankündigung zu ändern.

Alle in diesem Dokument in Beispielen verwendeten Namen, Firmennamen oder Firmen sind fiktiv und beziehen sich weder im Namen noch im Inhalt auf tatsächliche vorhandene Namen, Organisationen, juristische Personen oder Institutionen oder sollen diese darstellen. Jegliche Ähnlichkeit mit vorhandenen Personen, Organisationen, juristische Personen oder Institutionen ist rein zufällig.

<b>Dokumententitel</b>	Spider Technische Referenz - Installation, Konfiguration, Fehlerbehebung
<b>Produktversion</b>	6.4.5
<b>Herstellung und Druck</b>	Flexera Software LLC 300 Park Boulevard Itasca IL 60143, USA <a href="http://www.flexera.com">www.flexera.com</a>
<b>Veröffentlichungsdatum</b>	2022-01-17

Die in diesem Dokument beschriebene Software wird unter den Bedingungen eines Lizenzvertrags zur Verfügung gestellt und darf nur im Einklang mit den Bestimmungen dieser Vereinbarung verwendet werden.

Dieses Dokument darf ohne vorherige schriftliche Genehmigung von Flexera, weder vollständig noch auszugsweise kopiert, fotokopiert, reproduziert oder weiterverarbeitet werden.

# Inhaltsverzeichnis

<b>1</b>	<b>Allgemeines</b>	<b>7</b>
1.1	Typografische Konventionen.....	7
1.2	Hilfestellungen .....	7
<b>2</b>	<b>Einleitung</b>	<b>8</b>
2.1	Reihenfolge bei Neuinstallation .....	8
<b>3</b>	<b>Spider</b>	<b>8</b>
3.1	Systemvoraussetzungen.....	8
3.1.1	Applikationsserver .....	8
3.1.2	Datenbank - Microsoft SQL Server .....	10
3.1.3	Indexing - OneSearch .....	11
3.1.4	Hardware-Anforderungen.....	11
3.1.5	Clients / Browser.....	12
3.1.6	Implementationssystem .....	12
3.1.7	Berechtigungsanforderungen .....	13
3.1.8	Hardware-Skalierungen .....	14
	Szenario 1: Datenbank- und Applikationsserver in einem System .....	15
	Szenario 2: Datenbank und Applikationsserver auf separaten Systemen .....	16
	Szenario 3: Datenbank werden in einem SQL-Cluster betrieben .....	17
	Szenario 4: Einsatz einer Web-Farm.....	18
3.2	Installation.....	19
3.2.1	Start der Installation .....	19
3.2.2	Lizenzvereinbarung .....	20
3.2.3	Neuinstallation oder Aktualisierung .....	21
3.2.4	Lizenzdatei .....	22
3.2.5	Installationsverzeichnis.....	24
3.2.6	Startmenüeintrag.....	25
3.2.7	Datenbankserver.....	26
3.2.8	Index Proxy Server .....	27
3.2.9	Web Logins.....	29
	Spider .....	29
	SpiderAdmin.....	31
3.2.10	Datenbanknamen .....	33
3.2.11	Sprach- und Formateinstellungen.....	36
3.2.12	SMTP Einstellungen .....	37
3.2.13	Administrationskonto .....	38
3.2.14	IIS Konfiguration.....	38
3.2.15	Transferverzeichnis .....	39
3.2.16	Installation .....	40
3.2.17	Aktivieren der Lizenz.....	42
3.3	Erweiterte Konfiguration.....	44
3.3.1	Betrieb mit Domänenkonto .....	44
	SQL Server vorbereiten .....	44
	Domänenkonto für den IIS berechtigen .....	48
	Identität der Applikationspools anpassen .....	51
	Umstellen der Zugangsdaten .....	54
3.3.2	Nachträgliche Einrichtung Index Proxy .....	59
	Konfigurationseinträge.....	59
	IIS Applikation connectionStrings.config.....	59
	Einstellungen in der Registry .....	60
3.3.3	Umstellung auf SSL Betrieb .....	61
	IIS Default Web Site konfigurieren .....	61

	Konfigurationseinträge.....	67
	IIS neu starten .....	68
3.3.4	HTTP(S) Umleitung .....	69
3.3.5	Einrichtung Active Directory Login.....	73
	Einstellungen in Spider .....	73
	IIS Einstellungen - Authentication .....	79
3.3.6	Info Seite - support.config.....	82
3.3.7	Verfügbare Sprachen - culture.config.....	84
3.3.8	Einstellungen in der Config Tabelle.....	85
3.3.9	Anzeige erweiterter Fehlermeldungen .....	85
3.3.10	Verschlüsselung der Konfiguration .....	88
3.3.11	Fehlerbehandlungssystem .....	93
3.4	Aktualisierung .....	94
3.4.1	Start des Updates.....	94
3.4.2	Aktualisierung .....	95
3.4.3	Auswahl der Instanz.....	96
3.4.4	Datenbankserver.....	97
3.4.5	Index Proxy Server .....	99
3.4.6	Updates wählen .....	101
3.4.7	Installation .....	103
<b>4</b>	<b>Spider Web API</b>	<b>104</b>
4.1	Technische Grundlagen.....	104
4.1.1	Rest API .....	104
4.1.2	Versionierung.....	105
4.1.3	Mandantenauswahl .....	105
4.1.4	Request .....	105
4.1.5	Response.....	106
4.2	Authentifizierung .....	106
4.2.1	Generierung eines Token .....	107
4.2.2	Verwendung eines Token .....	107
4.2.3	Gültigkeit eines Token .....	107
4.2.4	Verwaltung der Token.....	107
4.3	Admin Bereich .....	108
4.3.1	Mandator .....	108
4.3.2	User .....	108
4.3.3	Profile.....	108
4.4	Entity Bereich .....	108
4.4.1	Entity-Zugriff .....	108
4.4.2	Entity-History-Zugriff .....	109
4.4.3	Entity-Dokumenten-Zugriff .....	109
4.4.4	Entity-Wiedervorlagen-Zugriff .....	109
4.4.5	Entity-Relationship-Zugriff .....	109
4.5	Search Bereich.....	110
4.5.1	Parametrisierung .....	111
4.6	Report Bereich .....	112
4.6.1	Parametrisierung .....	113
4.7	Meta Bereich .....	114
4.7.1	Mandator .....	114
4.7.2	Admin .....	114
4.7.3	Entity.....	114
4.7.4	Search .....	115
4.7.5	Report .....	115
4.8	Versionsinformationen.....	116
4.8.1	Einfache Versionsinformation.....	116
4.8.2	Erweiterte Versionsinformation .....	116

4.9	Swagger .....	116
4.9.1	OpenAPI Document .....	117
4.9.2	Swagger UI .....	117
4.10	Web API Dokumentation .....	120
4.11	Spider Web API Explorer .....	124
4.11.1	Benutzermerkmale .....	124
4.11.2	Authentifizierung .....	124
4.11.3	Sitzungsparameter (Mandant, Sprache, Version) .....	125
4.11.4	Kommunikation mit der Spider Web API .....	125
	Response .....	125
	Responsemodell .....	125
4.11.5	Web API Explorer Bereiche .....	127
	Search API .....	128
	Report API .....	131
	Entity API .....	134
4.12	Code-Beispiele .....	143
4.12.1	Authentifizierung .....	143
4.12.2	Entity Bereich .....	144
4.12.3	Search Bereich .....	150
4.12.4	Report Bereich .....	152
4.12.5	Meta Bereich .....	154
<b>5</b>	<b>Software Services (Kataloge u. Recognition)</b> .....	<b>161</b>
5.1	Systemvoraussetzungen .....	161
5.1.1	Applikationsserver .....	161
5.1.2	Datenbankserver .....	163
	Benötigte Datenbank .....	163
	Benötigter SQL-Benutzer .....	163
	Übersicht erforderlicher Rechte .....	164
5.2	Installation .....	165
5.2.1	Start der Installation .....	165
5.2.2	Lizenzvereinbarung .....	166
5.2.3	Lizenzdatei .....	167
5.2.4	Installationsverzeichnis .....	169
5.2.5	Datenbankserver .....	170
5.2.6	Datenbank & Data Receiver .....	172
5.2.7	SQL-Login, Service Benutzer .....	172
5.2.8	Zusammenfassung .....	174
5.2.9	Installation .....	175
5.3	Erweiterte Konfiguration .....	177
5.3.1	Data Receiver .....	177
	SSL verschlüsselte Übertragung .....	177
5.3.2	EDC-Monitor .....	178
5.3.3	Betrieb mit Domänenkonto .....	179
	SQL Server vorbereiten .....	179
	Ordnerberechtigungen .....	180
	Dienstkonto anpassen .....	181
	Connection String anpassen .....	181
5.4	Aktualisierung .....	182
5.4.1	Start des Updates .....	182
5.4.2	Lizenzvereinbarung .....	183
5.4.3	Lizenzdatei .....	184
5.4.4	Installationsverzeichnis .....	186
5.4.5	Datenbankserver .....	187
5.4.6	Zusammenfassung .....	189
5.4.7	Installation .....	190

<b>6</b>	<b>Datacenter-Modul (novaratio Appliance)</b>	<b>192</b>
6.1	Sizing der virtuellen Maschine .....	192
6.2	Inbetriebnahme der Appliance .....	193
6.2.1	Importieren der VM .....	193
6.2.2	Betriebssystem User .....	194
6.2.3	Anpassung des Hostnamens .....	194
6.2.4	Anpassung der IP-Adresse .....	194
6.2.5	Erzeugen eines Zertifikats .....	195
6.3	Einrichten des Datacenter-Moduls .....	197
6.3.1	Allgemeiner Hinweis .....	197
6.3.2	Registrierung .....	197
6.3.3	Mandantenzuweisung .....	198
<b>7</b>	<b>DSGVO Anforderungen in Spider</b>	<b>200</b>
7.1	Personenbezogene Daten .....	200
7.1.1	Welche Datenkategorien werden in Spider-Produkten verarbeitet? .....	200
7.1.2	Besondere Kategorien personenbezogener Daten (Artikel 9 Abs. 1 DSGVO)..... Kundenindividuelle Erweiterungen prüfen! .....	200
7.1.3	Verarbeitung personenbezogener Daten (Artikel 4 Nr. 2 DSGVO) .....	201
7.2	Personenbezogene Datenfelder an ausgewählten Spider Datenobjekten .....	201
7.2.1	Personenbezogene Daten am Mitarbeiter (Employee) .....	201
7.2.2	Personenbezogene Daten an Assets .....	202
7.2.3	Personenbezogene Daten am Spider Benutzer (User) .....	203
7.2.4	Personenbezogene Daten an den Kontaktperson (Contact Person) .....	204
7.3	Grundsätze für die Verarbeitung personenbezogener Daten (Artikel 5 DSGVO) .....	205
7.3.1	Angemessenheit und Erheblichkeit von personenbezogenen Daten, sowie die Beschränkung derer Verarbeitung auf das notwendige Maß (Artikel 5, Nr. 1, lit. c DSGVO).....	205
7.3.2	Angemessene Speicherung personenbezogener Daten, sowie deren mögliche Anonymisierung (Artikel 5, Nr. 1, lit. e DSGVO) .....	205
7.3.3	Betroffenenrechte .....	206
	Auskunftsrecht (Artikel 15 DSGVO).....	206
	Berichtigung und Löschung personenbezogener Daten (Artikel 5, Nr. 1, lit. d DSGVO).....	206
	Recht auf Berichtigung (Artikel 16 DSGVO).....	206
	Recht auf Löschung (Artikel 17 DSGVO) .....	206
	Recht auf Einschränkung der Verarbeitung (Artikel 18 DSGVO) .....	207
	Mitteilungspflicht (Artikel 19 DSGVO).....	207
	Recht auf Datenübertragbarkeit (Artikel 20 DSGVO) .....	207
	Widerspruchsrecht (Artikel 21 DSGVO).....	207
	Geeignete Möglichkeiten zum Nachweis der Einhaltung der Anforderungen im Rechenschaftsfall (Artikel 5, Nr. 2 DSGVO).....	208
7.4	Rechtmäßigkeit der Verarbeitung (Artikel 6 DSGVO).....	208
7.5	Datenschutz durch Technikgestaltung und durch datenschutzfreundliche Voreinstellungen (Artikel 25 DSGVO).....	208
7.5.1	Vertraulichkeit (Art 32 Abs. 1 lit. b DSGVO) .....	208
7.5.2	Integrität (Art 32 Abs. 1 lit b DSGVO).....	209
7.5.3	Verfügbarkeit und Belastbarkeit (Art 32 Abs. 1 lit b DSGVO) .....	209
7.6	Bedingungen für die Einwilligung (Artikel 7 DSGVO) .....	209
7.6.1	Self-Service-Funktionen .....	209
7.6.2	Einwilligung bei personenbezogenen Daten aus Fremdsystemen oder für die Erhebung zusätzlicher Daten .....	209
7.7	Informationspflicht (Artikel 13 und 14 DSGVO) .....	210
7.7.1	Verarbeitung von Beschäftigungsdaten.....	210
7.7.2	Direkte Benachrichtigung der Betroffenen .....	210
7.8	Verzeichnis von Verarbeitungstätigkeiten (Artikel 30 DSGVO).....	210
<b>8</b>	<b>Logfiles</b>	<b>210</b>
8.1	Brainware.log .....	211

## Allgemeines

### 1.1 Typografische Konventionen

In diesem Handbuch werden verschiedene Formatierungen verwendet, um bestimmte Begriffe und Aktionen hervorzuheben. Spezielle Hinweise und Tipps werden je nach Gewichtung mit einer anderen Hintergrundfarbe dargestellt.

Formatierung	Beschreibung
<b>Fette Schrift</b>	Elemente in der Software oder im Betriebssystem, wie Menüpunkte, Buttons oder Elemente einer Auswahlliste
<i>Kursivschrift</i>	Hervorhebungen (wichtige Details) und Verweise auf andere Kapitel oder Dokumente
Dreieck Symbol "➤"	Schritt einer Handlungsanweisung
Spitze Klammer ">"	Befehlsmenüabläufe, z.B. <b>Datei &gt; Öffnen</b>
<i>Systemschrift</i>	Verzeichnisse, Code- und Scriptbeispiele
GROSSBUCHSTABEN	Tastenbezeichnungen, z.B. SHIFT, STRG, oder ALT
TASTE+TASTE	Tastenkombinationen, bei welchen der Benutzer eine Taste gedrückt halten muss und eine weitere Taste drückt, z.B. STRG+P oder ALT+F4.

<b>Hinweis</b>	Wird für Hinweise oder Tipps verwendet, welche die Arbeit erleichtern oder für zusätzliche Informationen, die das Verständnis für das Produkt fördern.
----------------	--

<b>Wichtig</b>	Informationen, die der Benutzer beachten sollte, da sonst Probleme oder Mehraufwand im Betrieb entstehen können.
----------------	--

<b>Achtung</b>	Informationen, die der Benutzer beachten muss, um Fehlfunktionen des Systems (Abstürze, Datenverluste, Systemausfall) zu verhindern.
----------------	--

### 1.2 Hilfestellungen

Für zusätzliche Informationen und Unterstützung empfehlen wir die [Flexera Community](https://community.flexera.com/) (https://community.flexera.com/). Hier finden Sie die Produktdokumentation, Download-Links und Zugang zum Support.

# Einleitung

---

Dieses Dokument beschreibt:

- Systemanforderungen
- Installation
- Aktualisierung
- Konfiguration der Produkte
  - Spider
  - Recognition

## 2.1 Reihenfolge bei Neuinstallation

---

Bei einer Neuinstallation ist folgende Installationsreihenfolge zu beachten:

- Spider
- Software Services (Katalog & Recognition)
- Datacenter-Modul (novaratio Appliance)

Für das Auslesen und Übermitteln von Daten aus Inventarsystemen sollte der Spider Data Collector (SDC) verwendet werden. Hinweise zur Installation und zur Einrichtung sind in einem separaten Dokument (Spider Data Collector - UserManual) beschrieben.

# Spider

---

## 3.1 Systemvoraussetzungen

---

### 3.1.1 Applikationsserver

---

#### Betriebssystem

**Empfohlene Version:** Windows Server 2019 Standard

**Unterstützte Versionen:**

- 2022
- 2019
- 2016
- 2012 / 2012 R2

**Unterstützte Editionen:**

- Standard
- Datacenter

#### Microsoft .NET Framework

Microsoft .NET Framework Version 4.7.2

## **Microsoft Internet Information Server (IIS)**

- Internet Information Service (IIS) Manager
- Installierter Rollendienst "IIS Management Scripts and Tools"

### 3.1.2 Datenbank - Microsoft SQL Server

---

- Empfohlene Versionen:** Microsoft SQL Server 2016 Standard (64 bit) oder  
Microsoft SQL Server 2017 Standard (64 bit)
- Unterstützte Versionen:**
- 2019
  - 2017
  - 2016
  - 2014
  - 2012
- Unterstützte Editionen:\***
- **Standard** - für Spider Umgebungen mit einem Mandanten
  - **Enterprise** - für große Multi-Mandanten-Umgebungen
  - **Express\* \*\*** - für sehr kleine Umgebungen, Einschränkungen müssen beachtet werden
- Erforderliche Einstellungen:** Im SQL Server Configuration Manager muss das Protokoll TCP/IP und Named Pipes für den SQL Server aktiviert sein.
- Für die Installation muss der SQL Server im Mixed Mode (SQL Server und Windows Authentication) betrieben werden. Nach der Erstinstallation und der Umstellung der Datenbankzugriffe auf Domain Benutzer kann der Server auf reinen Betrieb mit Windows Authentication umgestellt werden.

---

**Hinweis** \*Die Funktion Always On wird nicht unterstützt.  
\*\*Die Volltextsuche steht nur für die Edition Express mit Advanced Services zur Verfügung.

---

Für die Editionen Spider SAM, ITM oder Enterprise werden bis zu 8 Datenbanken benötigt:

- Core
- Core\_TempDB
- Asset
- Licence
- Contract
- Recognition
- SessionDB
- Integration (optional)

Spider Contract werden 5 Datenbanken benötigt:

- Core
- Core\_TempDB
- Contract
- SessionDB
- Integration (optional)

**Folgendes ist zu beachten:**

- Die Namen können individuell gewählt werden. Der Datenbankname darf nicht mit einer Zahl beginnen und eine Gesamtlänge von 50 Zeichen nicht überschreiten.
- Alle Spider Datenbanken müssen dieselbe Datenbank Kollation aufweisen. Es muss eine Case-Insensitive-Collation (CI-Collation) gewählt werden, bspw. SQL\_Latin1\_General\_CP1\_CI\_AS.

- Die Datenbanken müssen vor der Installation komplett leer sein.
- Für die Spider Core-Temp-Datenbank gilt eine Regel bei der Namensgebung. Der Name der Temp-Datenbank ist der Gleiche, wie der der zugehörigen Core-Datenbank, ergänzt um ein zusätzliches Suffix „\_TempDB“, bspw. Spider-Core\_TempDB.
- Für die Spider Core-SessionState-Datenbank gilt eine Regel bei der Namensgebung. Der Name der Datenbank ist der Gleiche, wie der der zugehörigen Core-Datenbank, ergänzt um ein zusätzliches Suffix „\_SessionState“, bspw. Spider-Core\_SessionState.

### 3.1.3 Indexing - OneSearch

---

Mit der OneSearch-Funktion wird die brainwaregroup Enterprise Services Infrastructure (ESI) als zusätzliche Komponente eingeführt und in zukünftigen Versionen weiter in das Produkt integriert.

Die Indexing Infrastruktur kann auf den Spider Applikationsserver oder einem separaten Server installiert werden.

- ca. 750 MB Installationsdateien (inkl. Embedded Java)
- >= 3GB HDD Platz für Indexdaten, abhängig vom Datenumfang
- 1 bis 1.5 GB Arbeitsspeicher

Weitere Informationen können dem Dokument "Brainwaregroup ESI Release Notes" entnommen werden.

### 3.1.4 Hardware-Anforderungen

---

Für die Hardware Anforderungen des Applikations- und Datenbankserver empfehlen wir ab mittleren Systemgrößen ein individuelles Server-Sizing.

Folgende allgemeine Orientierungswerte gelten für Produktionsinstanzen. Für Testinstanzen können kleinere Konfigurationen vorgesehen werden.

#### Prozessor

<b>Empfehlung:</b>	System mit Quad Core Prozessor mit 2 GHz oder mehr
Kleines System:	1+ Quad Core Prozessor(en)
Mittleres System	2+ Quad Core Prozessoren
Großes System:	4+ Quad Core Prozessoren

#### Arbeitsspeicher

Kleines System:	4 - 8 GB RAM
Mittleres System:	8 - 32 GB RAM
Großes System:	ab 32 GB RAM

#### Festplattenkapazität

Applikationsverzeichnis:	300 MB
Datenbank:	Datenbank- und Transaktionsspeicher sind abhängig von den verwalteten Daten, Dokumenten und Änderungshistorien.

Backupspeicher:	Je nach Backupstrategie sind entsprechend dem Datenvolumen für Datenbanken und Transaktionen Faktor 1-2 Speicherkapazitäten für Backups vorzuhalten.
Kleines System:	5 - 20 GB
Mittleres System::	10 - 50 GB
Großes System:	> 50 GB

### 3.1.5 Clients / Browser

---

#### Unterstützte Browser

<b>Internet Explorer</b>	<ul style="list-style-type: none"> <li>• Version 11</li> <li>• Version 10</li> <li>• <i>Ältere Versionen und Kompatibilitätslevel werden nicht unterstützt.</i></li> </ul>
--------------------------	--

#### Edge

**Firefox** ab Version 38 (ESR)

**Chrome** ab Version 46

---

**Hinweis** Active Scripting (IE) bzw. Java Script (Firefox) müssen aktiviert sein.

---

#### Empfohlene Software

**Microsoft Excel** Betrachten und Bearbeiten von Excel-Dateien

**PDF Viewer** PDF Viewer, bspw. Acrobat Reader

### 3.1.6 Implementationssystem

---

Das Implementationssystem ist ein Arbeitsplatzsystem, auf dem die Spider Implementation durchgeführt werden kann.

Folgende Software wird benötigt:

- Microsoft SQL Server Management Studio
- Microsoft Office (Word, Excel)
- PDF Viewer, bspw. Acrobat Reader
- Spider Administrationstool

Folgende Zugriffe müssen ermöglicht werden:

- Remote-Zugriff auf Applikationsserver
- Zugriff auf die Spider-Datenbanken

### 3.1.7 Berechtigungsanforderungen

Für die Installation, Konfiguration und Realisierung der Applikation sowie den relevanten Schnittstellen werden folgende Zugriffsberechtigungen und Einstellungen benötigt:

Typ	Bezeichnung	Beschreibung
Account	SQL Account für Installationen	Installationsbenutzer. DBO- Berechtigungen nur auf den Spider-Datenbanken.
Account	SQL Accounts "Spider" und "SpiderAdmin"	Standard-Account für Spider. Werden durch das Setup an den Spider Datenbanken berechtigt.
Account	Domain Login-Account für den Applikationsserver	Account mit Administrator Berechtigungen auf Applikationsserver
Account	Schnittstellen Domain-User	Permanenter Service Account ohne Passwort change / expiry Policy für die kontinuierlichen Schnittstellen.
Network Policy	Kommunikation Applikations- und Datenbankserver	Die beiden Server müssen miteinander kommunizieren können, falls diese auf zwei unterschiedlichen Server Plattformen betreiben werden. SQL Standard Port 1433
FW Policy	Konfiguration Firewall	Prüfung der Konfiguration der Firewall, damit die vorgesehenen Zugriffe zu den Fremdsystemen möglich sind (inkl. Schnittstellen zum InventoryScanner).

Die erforderlichen Rechte für die Ausführung der Setups entnehmen Sie bitte der unten stehenden Tabelle

Task	Rechte an	Recht(e)
Erstellen von Logins	Server	ALTER ANY LOGIN
Erstellen von Datenbanken	Server	CREATE DATABASE CREATE ANY DATABASE ALTER ANY DATABASE
Erstellen Extended Properties	Datenbank	ALTER
Einrichten der SessionState Datenbank	Datenbank	msdb -> EXECUTE

#### Benötigte SQL-Benutzer

Wenn der SQL-Installationsbenutzer nicht das Recht hat neue Logins auf dem Datenbankserver anzulegen, müssen diese vorab angelegt werden. Für das Setup werden zwei SQL-Benutzer benötigt. Die Anmeldedaten werden während der Installation abgefragt. Das Setup vergibt dann die jeweiligen Berechtigungen auf den Spider Datenbanken.

##### 1. Spider Login

Das Spider-Login wird als Applikationslogin für Spider benötigt. Die Webanwendung und die Services verwenden diesen Login für den Zugriff auf die Datenbanken.

##### 2. Spider Admin Login

Dieser Benutzer ist für die Verwendung mit dem Admin-Tool vorgesehen.

### 3.1.8 Hardware-Skalierungen

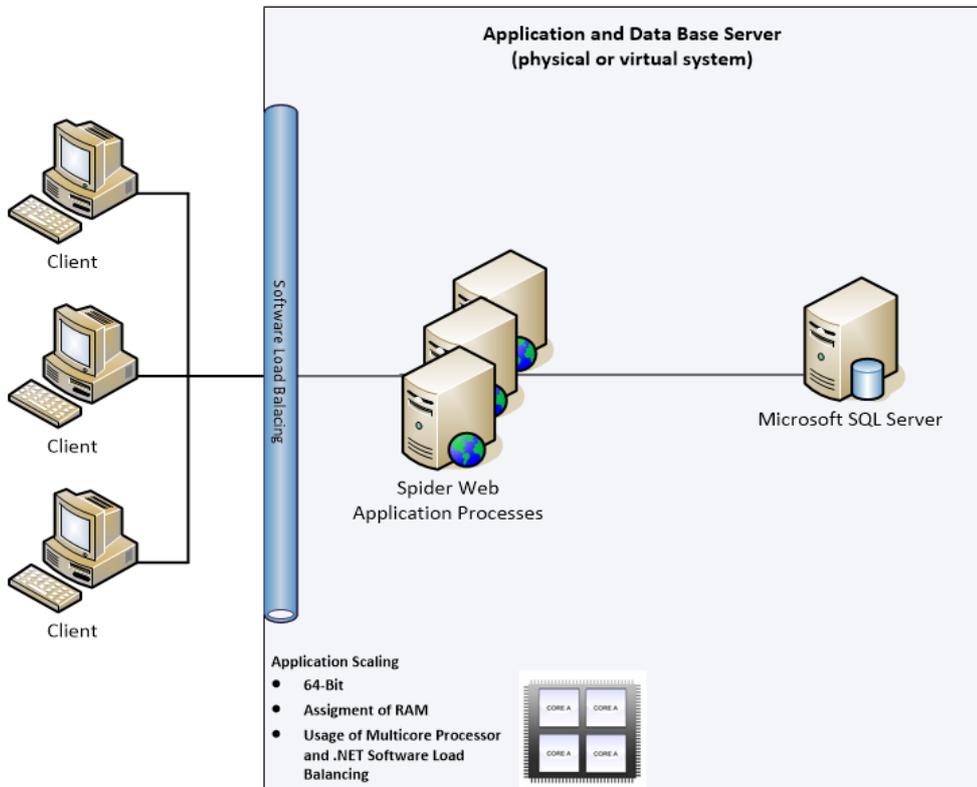
Nachfolgend sind die Systemgrößen und vier verschiedenen Szenarien dargestellt. Es handelt sich dabei um Empfehlungen basierend auf Erfahrungswerten aus zahlreichen Kundenumgebungen. Bei großen Systemen empfiehlt sich ein individuelles Server-Sizing.

#### Systemgrößen

Größe	Produkt	Mengengerüst	Szenario(s)
Klein	Asset	< 5.000 Systeme	1
	Contract	< 2.500 Verträge	
Mittel	Asset	5.000 – 20.000 Systeme	1 oder 2
	Contract	2.500 – 10.000 Verträge	
Groß	Asset	> 20.000 Systeme	2, 3, oder 4
	Contract	> 10.000 Verträge	

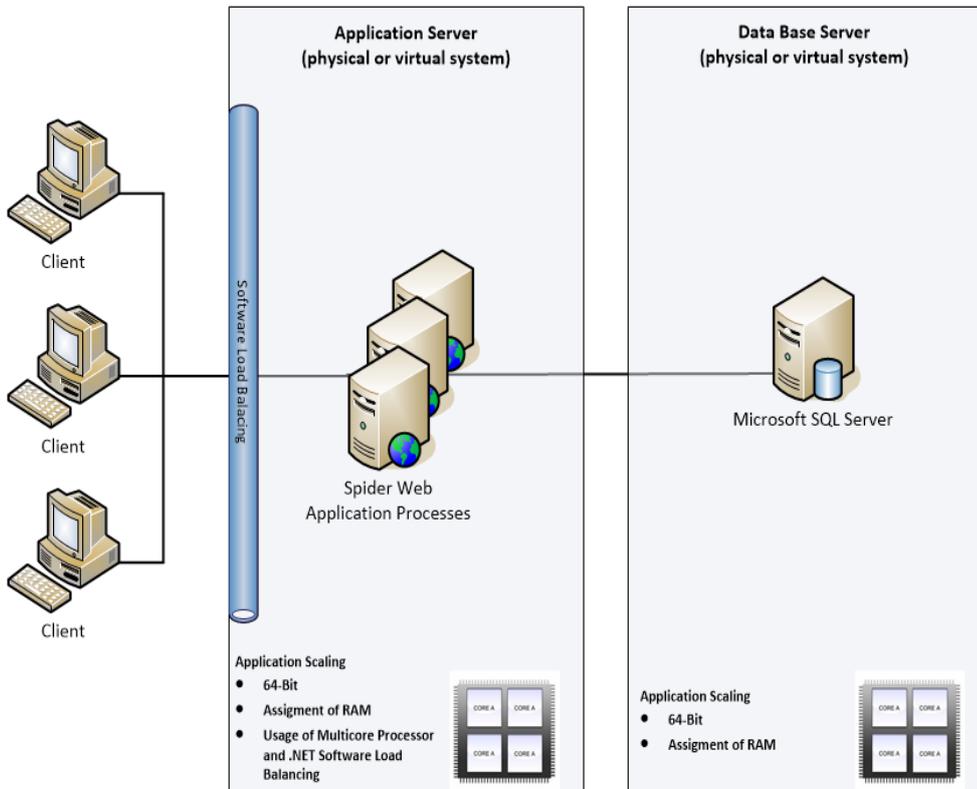
## Szenario 1: Datenbank- und Applikationsserver in einem System

- Datenbank- und Applikationsserver befinden sich auf einem System.
- 64 Bit System für Applikationsserver und Datenbank.
- Mögliche Skalierung über Prozessoren/Cores und Arbeitsspeicher.
- Software Load Balancing (SLB): Anfragen werden auf bereitgestellte Prozesse verteilt.
- Bei einem virtuellen System ist für die Datenbank auf gute I/O-Performance zu achten.



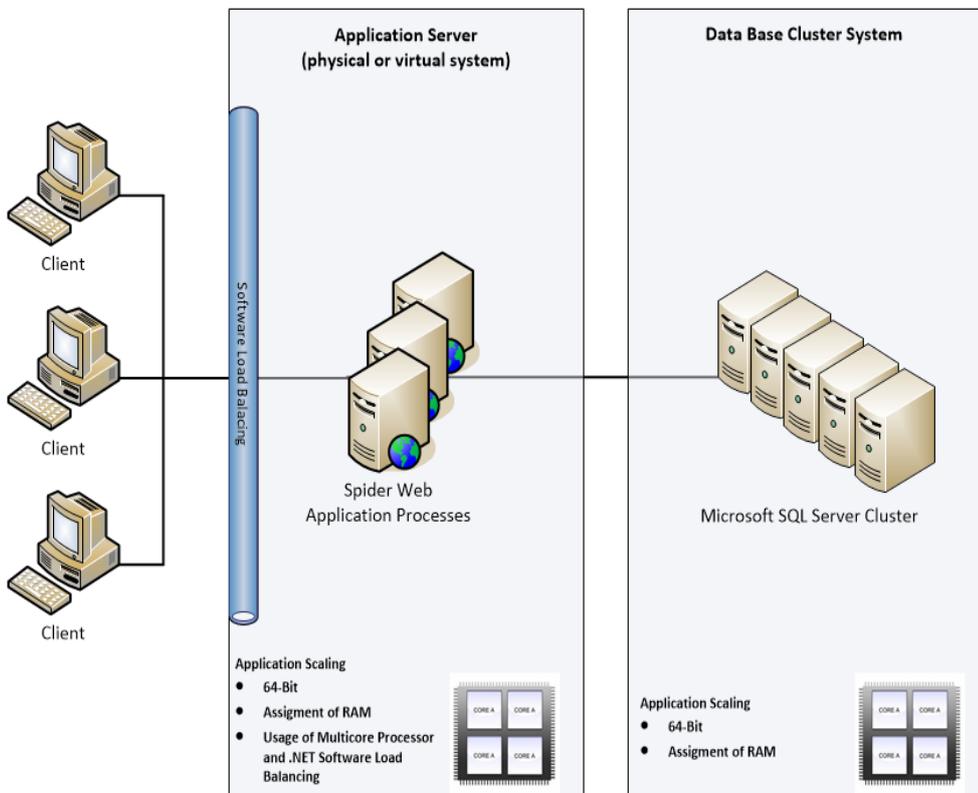
## Szenario 2: Datenbank und Applikationsserver auf separaten Systemen

- Datenbank- und Applikationsserver befinden sich auf unterschiedlichen Systemen.
- 64 Bit Systeme für Applikationsserver und Datenbank.
- Mögliche Skalierung über Prozessoren/Cores und Arbeitsspeicher je System.
- Software Load Balancing (SLB): Anfragen werden auf bereitgestellte Prozesse verteilt.
- Bei einem virtuellen System ist für die Datenbank auf gute I/O-Performance zu achten.



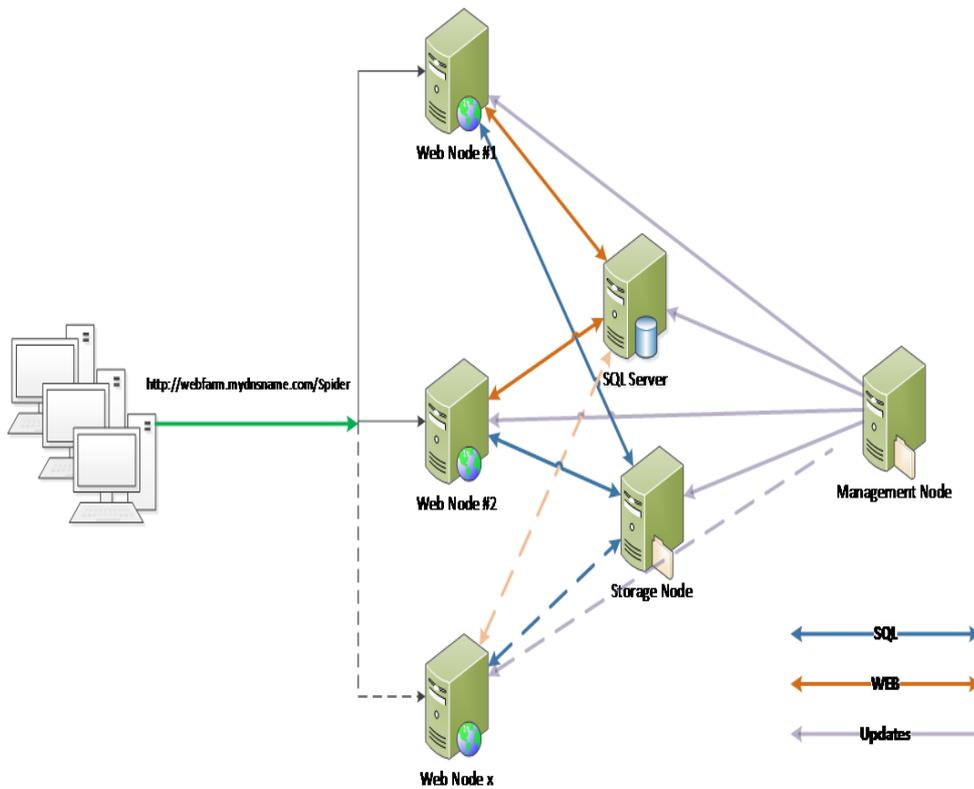
### Szenario 3: Datenbank werden in einem SQL-Cluster betrieben

- Aufbauend auf dem 2. Szenario
- Die Datenbanken werden in einem SQL-Server-Cluster betrieben.
- Mögliche Skalierung über Prozessoren/Cores und Arbeitsspeicher je System.
- Software Load Balancing (SLB): Anfragen werden auf bereitgestellte Prozesse verteilt.



## Szenario 4: Einsatz einer Web-Farm

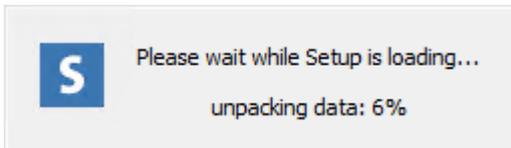
- Aufbauend auf dem 2. oder 3. Szenario
- Es wird eine Web-Farm aus mehreren Web Nodes und einem Management Node betrieben. Auf dem Management Node befinden sich die initiale Spider-Installation, der Lizenzservice und die Spider Services.
- Mögliche Skalierung über Prozessoren/Cores und Arbeitsspeicher je System.
- Kombination von Network Load Balancing in der Web Farm und Software Load Balancing jedes einzelnen Webservers der Farm.
- Software Load Balancing (SLB): Anfragen werden auf bereitgestellte Prozesse verteilt.



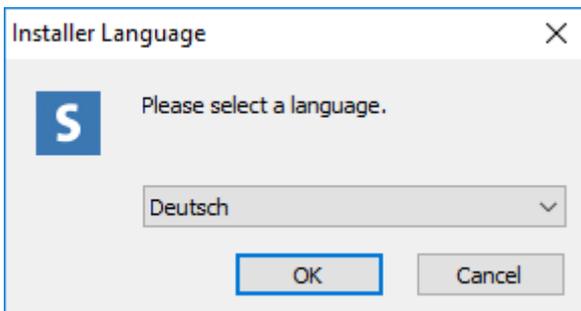
## 3.2 Installation

### 3.2.1 Start der Installation

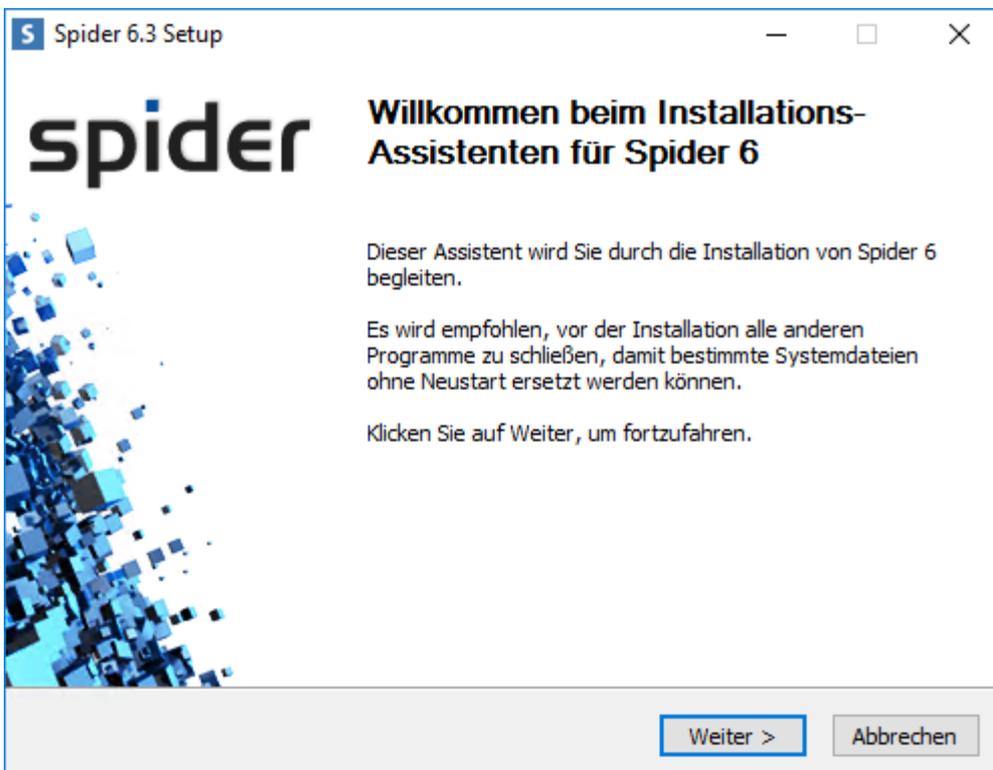
Starten Sie die Installation, indem Sie die Spider 6.x Setup.exe ausführen.



Zunächst werden Sie dazu aufgefordert, die Sprache für den Installationsassistenten auszuwählen. Klicken Sie hierfür auf die gewünschte Sprache in der Auswahlbox und bestätigen Sie Ihre Eingabe mit „OK“.



Anschließend öffnet sich der Installationsassistent für Spider. Er wird Ihnen in der gewählten Sprache bei der Installation des Setups behilflich sein. Klicken Sie nun auf „Weiter“ um die Installation zu starten.



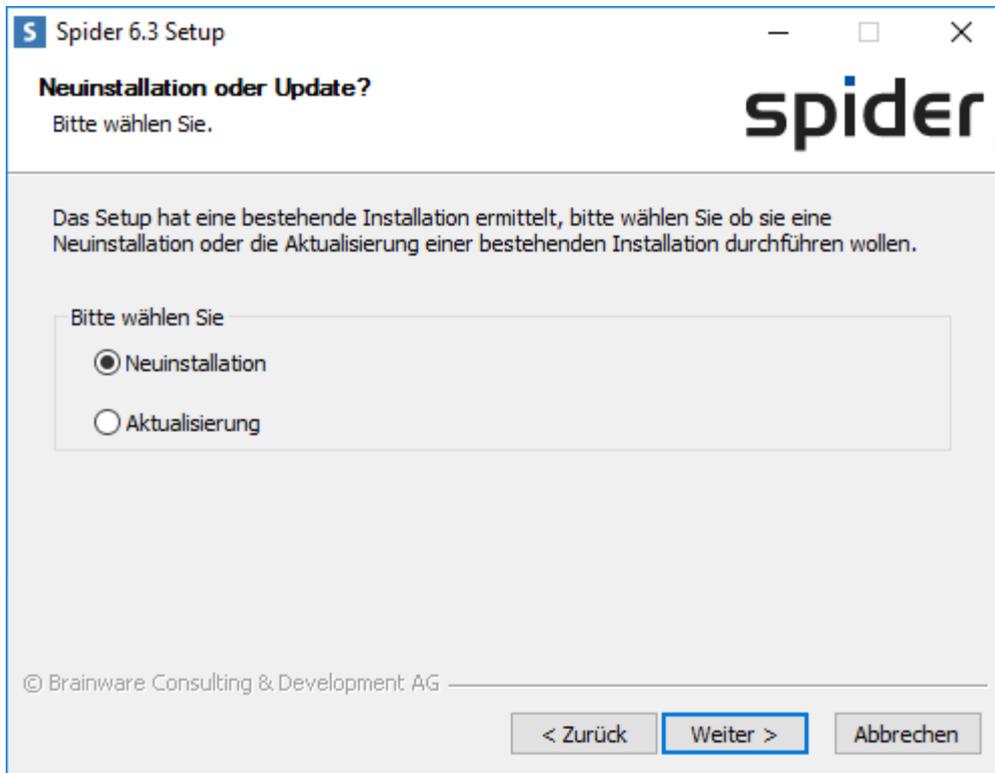
## 3.2.2 Lizenzvereinbarung

Im nächsten Schritt werden Sie gebeten die Lizenzvereinbarung vor Beginn der Installation zu lesen und zu akzeptieren. Das Setup lässt sich nur mit akzeptierter Lizenzvereinbarung weiter ausführen.



### 3.2.3 Neuinstallation oder Aktualisierung

Falls bereits eine Instanz von Spider installiert ist, werden Sie aufgefordert sich für eine Art der Installation zu entscheiden.

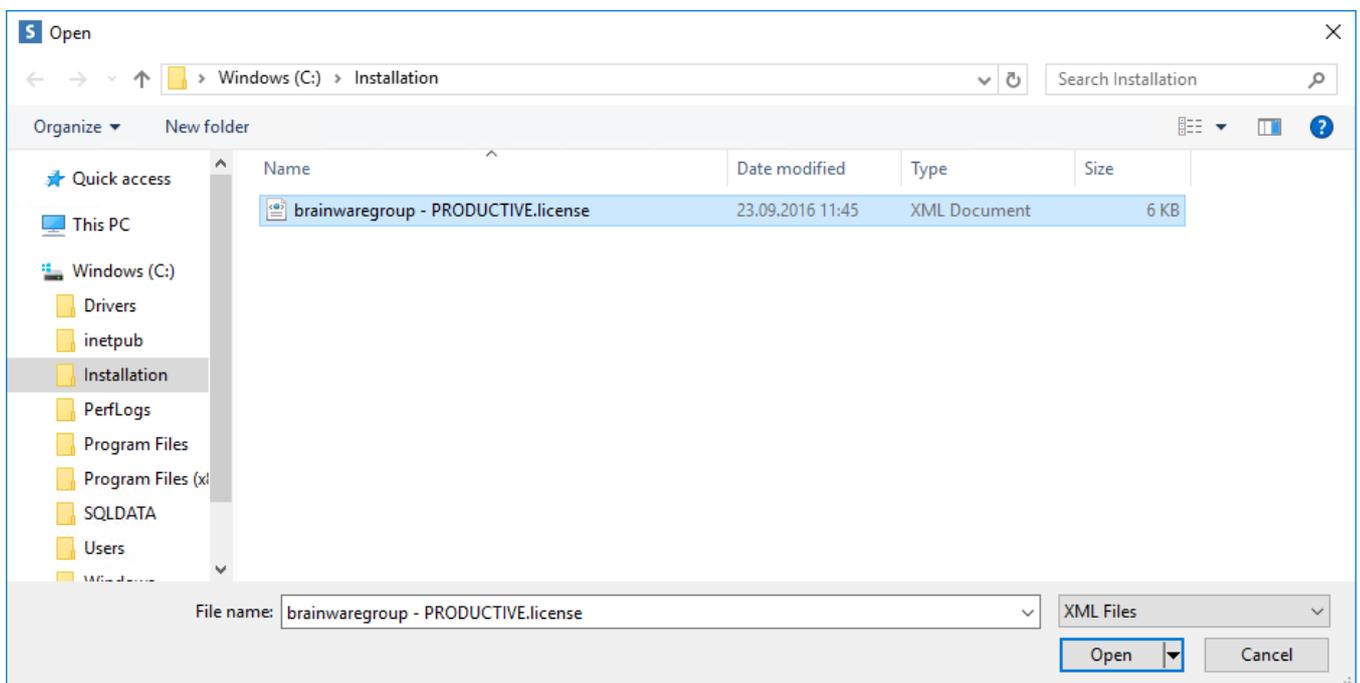


### 3.2.4 Lizenzdatei

Wählen Sie die passende Lizenzdatei aus, um das Setup fortzuführen. Ohne Lizenzdatei lässt sich das Setup nicht weiter fortführen.

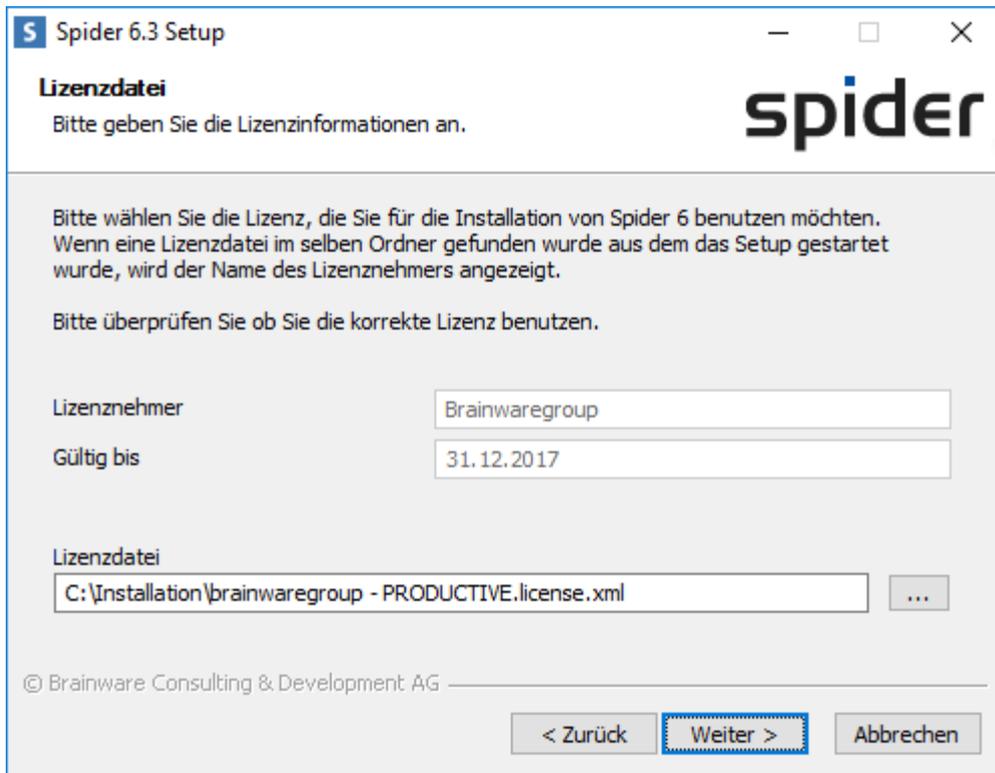


Klicken auf die "... " Schaltfläche:



Auswahl der Lizenz und dann "Open".

Die ausgewählte Lizenz wird geladen und die Details angezeigt.



The screenshot shows a Windows dialog box titled "Spider 6.3 Setup". The window has standard minimize, maximize, and close buttons. The main heading is "Lizenzdatei" (License file). Below the heading, it says "Bitte geben Sie die Lizenzinformationen an." (Please provide the license information). The "spider" logo is visible in the top right corner of the dialog. The main text area contains instructions: "Bitte wählen Sie die Lizenz, die Sie für die Installation von Spider 6 benutzen möchten. Wenn eine Lizenzdatei im selben Ordner gefunden wurde aus dem das Setup gestartet wurde, wird der Name des Lizenznehmers angezeigt." (Please select the license you want to use for the installation of Spider 6. If a license file is found in the same folder from which the setup was started, the name of the licensee will be displayed.) and "Bitte überprüfen Sie ob Sie die korrekte Lizenz benutzen." (Please check if you are using the correct license.). There are three input fields: "Lizenznehmer" (Licensee) with the value "Brainwaregroup", "Gültig bis" (Valid until) with the value "31.12.2017", and "Lizenzdatei" (License file) with the value "C:\Installation\brainwaregroup - PRODUCTIVE.license.xml" and a browse button "...". At the bottom left, there is a copyright notice: "© Brainware Consulting & Development AG". At the bottom right, there are three buttons: "< Zurück" (Back), "Weiter >" (Next), and "Abbrechen" (Cancel). The "Weiter >" button is highlighted with a blue dashed border.

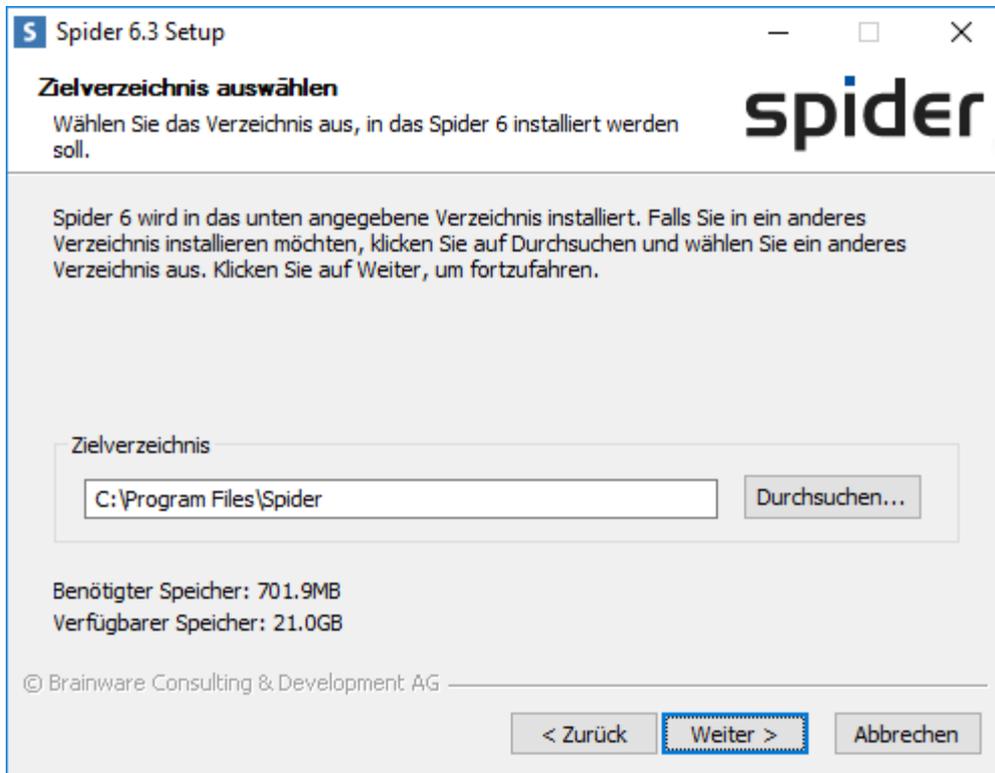
---

**Hinweis** Falls eine Lizenz im gleichen Verzeichnis gefunden wird aus der das Setup gestartet wurde, wird diese (sofern sie gültig ist) automatisch in den Dialog geladen.

---

### 3.2.5 Installationsverzeichnis

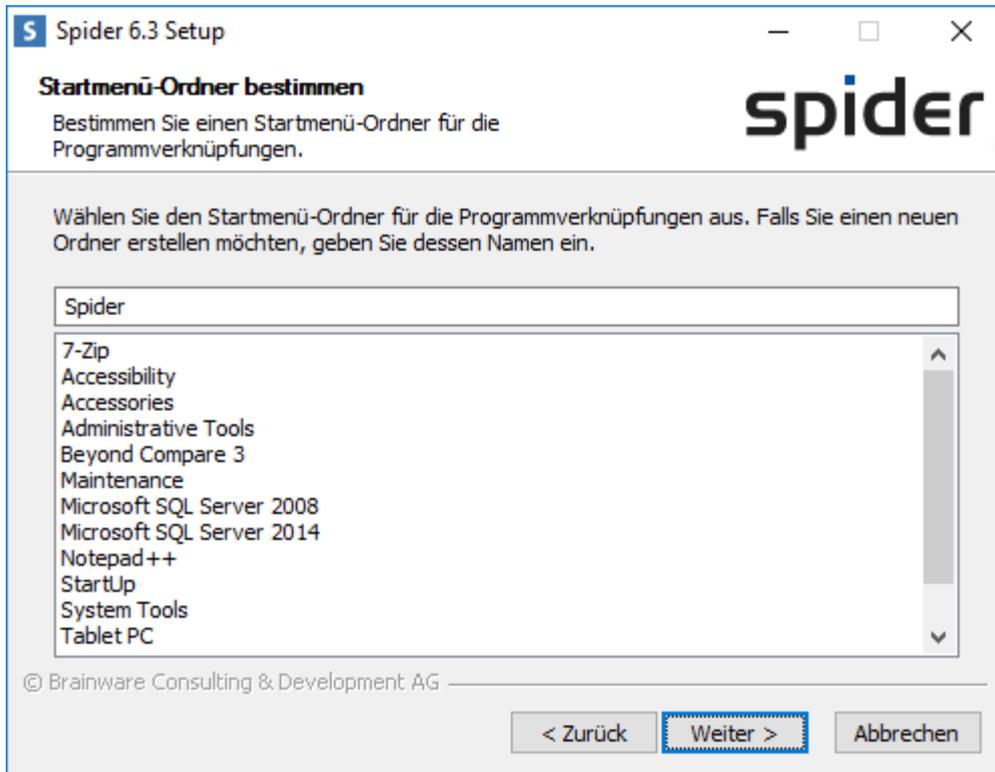
Im nun folgenden Schritt können Sie ein Zielverzeichnis für die Installation auswählen. Den vorgeschlagenen Standardpfad können Sie ändern.



The screenshot shows a Windows installation window titled "Spider 6.3 Setup". The window has a title bar with standard minimize, maximize, and close buttons. The main content area is titled "Zielverzeichnis auswählen" (Select target directory) and includes the "spider" logo. The instructions state: "Wählen Sie das Verzeichnis aus, in das Spider 6 installiert werden soll." (Select the directory where Spider 6 should be installed). Below this, a paragraph explains: "Spider 6 wird in das unten angegebene Verzeichnis installiert. Falls Sie in ein anderes Verzeichnis installieren möchten, klicken Sie auf Durchsuchen und wählen Sie ein anderes Verzeichnis aus. Klicken Sie auf Weiter, um fortzufahren." (Spider 6 will be installed in the directory specified below. If you want to install in a different directory, click Search and select a different directory. Click Next to continue). A text input field labeled "Zielverzeichnis" (Target directory) contains the path "C:\Program Files\Spider". To the right of the input field is a "Durchsuchen..." (Search...) button. Below the input field, the required and available storage space are listed: "Benötigter Speicher: 701.9MB" (Required memory: 701.9MB) and "Verfügbarer Speicher: 21.0GB" (Available memory: 21.0GB). At the bottom left, the copyright notice "© Brainware Consulting & Development AG" is visible. At the bottom right, there are three buttons: "< Zurück" (Back), "Weiter >" (Next), and "Abbrechen" (Cancel). The "Weiter >" button is highlighted with a blue dashed border.

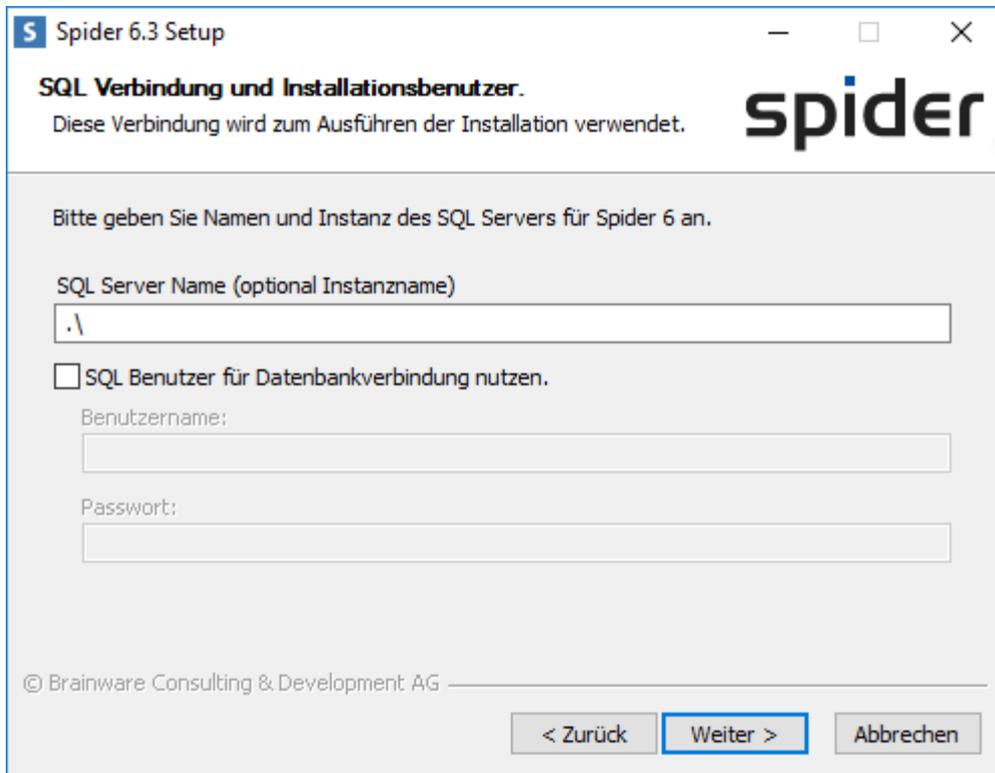
### 3.2.6 Startmenüeintrag

Hier können Sie festlegen, welchen Ordner Spider für das Startmenü verwenden soll.



### 3.2.7 Datenbankserver

Im nun folgenden Schritt bittet Sie der Installationsassistent den Namen und die Instanz eines Microsoft SQL-Servers anzugeben, auf dem die Spider Datenbanken installiert werden sollen.



The screenshot shows a Windows-style window titled "Spider 6.3 Setup". The main heading is "SQL Verbindung und Installationsbenutzer." with a sub-note: "Diese Verbindung wird zum Ausführen der Installation verwendet." The "spider" logo is in the top right. The instruction reads: "Bitte geben Sie Namen und Instanz des SQL Servers für Spider 6 an." There is a text input field for "SQL Server Name (optional Instanzname)" containing ".\". Below it is a checkbox labeled "SQL Benutzer für Datenbankverbindung nutzen." which is unchecked. Underneath are three input fields: "Benutzername:", "Passwort:", and a third empty field. At the bottom left is the copyright notice "© Brainware Consulting & Development AG". At the bottom right are three buttons: "< Zurück", "Weiter >" (highlighted with a blue border), and "Abbrechen".

Der Benutzer, mit dem die Verbindung hergestellt wird, kann entweder der angemeldete (Windows-)Benutzer sein, oder es kann ein SQL Benutzer verwendet werden.

Für die Verwendung eines SQL Benutzers, ist das Kontrollkästchen **SQL Benutzer für Datenbankverbindung nutzen** zu markieren sowie die Felder **Benutzername** und **Passwort** mit den gewünschten Daten auszufüllen.

The screenshot shows the 'Spider 6.3 Setup' window with the title 'SQL Verbindung und Installationsbenutzer'. Below the title, it says 'Diese Verbindung wird zum Ausführen der Installation verwendet.' and the 'spider' logo. The main instruction is 'Bitte geben Sie Namen und Instanz des SQL Servers für Spider 6 an.' There is a text input field for 'SQL Server Name (optional Instanzname)' containing '.\'. Below it is a checked checkbox 'SQL Benutzer für Datenbankverbindung nutzen'. Underneath are two more input fields: 'Benutzername:' containing '<SQL Benutzer >' and 'Passwort:' containing five dots. At the bottom, there are three buttons: '< Zurück', 'Weiter >' (highlighted with a blue border), and 'Abbrechen'. The footer contains '© Brainware Consulting & Development AG'.

**Hinweis** Die Benutzer müssen vorab auf dem Datenbankserver mit den entsprechenden Rechten eingerichtet werden. Abhängig von den Rechten, mit denen der Benutzer ausgestattet ist, stehen in den weiteren Folgedialogen verschiedene Optionen zur Verfügung.

Feld	Beschreibung
SQL Server Name	Name des SQL Servers, ggfs. inklusive Instanznamen und Port z.B. SQLSERVER\SQLINSTANZ,1234
SQL Benutzer für Datenbankverbindung nutzen	Auswahl, ob der angemeldete Domainbenutzer verwendet wird oder ob ein SQL Benutzer verwendet werden soll.
Benutzername	Name des SQL Benutzers.
Passwort	Passwort des SQL Benutzers.

### 3.2.8 Index Proxy Server

Als nächstes kann gewählt werden, ob der Index Proxy während des Setups eingerichtet werden soll.

Der Index Proxy Server ist Teil der Enterprise Services Infrastructure (ESI).

**S** Spider 6.3 Setup

**Index Proxy**  
Details zur Index Proxy Konfiguration

Index Proxy verwenden

Name des Index Proxy Portal Servers  
servername.domain.suffix

Port des Index Proxy Portal Servers  
7080

Benutzername für das Token Service Konto  
<Client ID>

Passwort für das Token Service Konto

© Brainware Consulting & Development AG

< Zurück **Weiter >** Abbrechen

Durch das Markieren des Kontrollkästchens wird der Dialog aktiviert und die benötigten Angaben können eingetragen werden.

**S** Spider 6.3 Setup

**Index Proxy**  
Details zur Index Proxy Konfiguration

Index Proxy verwenden

Name des Index Proxy Portal Servers  
servername.domain.suffix

Port des Index Proxy Portal Servers  
7080

Benutzername für das Token Service Konto  
<Client ID>

Passwort für das Token Service Konto

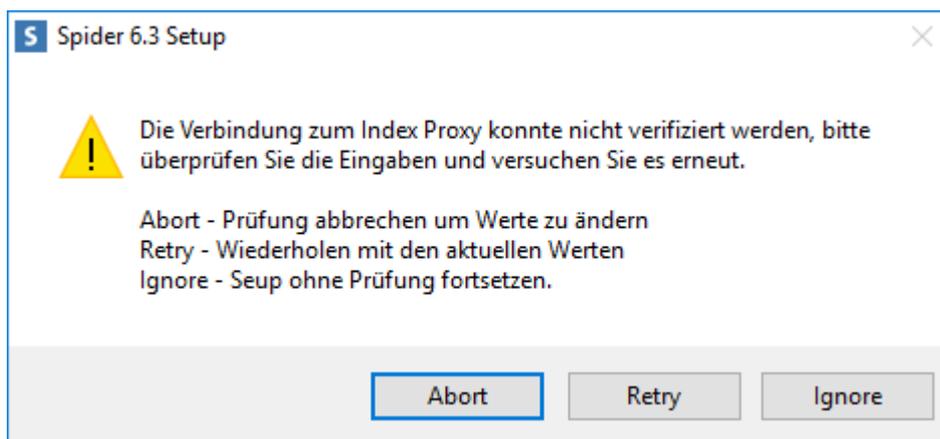
© Brainware Consulting & Development AG

< Zurück **Weiter >** Abbrechen

Feld	Beschreibung
------	--------------

Feld	Beschreibung
Index Proxy verwenden	Aktivierung des Dialogs und Abfrage der Details
Name des Index Proxy Portal Servers	Name (FQDN) des Servers, auf dem der Index Proxy installiert ist,
Port des Index Proxy Portal Servers	Port (Default: 7080) auf dem der Index Proxy auf Anfragen reagiert
Benutzername des Token Service Kontos	Die Client ID des Benutzers der für die Verbindung zum Index Proxy verwendet werden soll, die Client ID wurde während der Installation des Index Proxy vergeben. Bei der Client ID ist auf Groß- und Kleinschreibung zu achten!
Passwort des Token Service Konto	Passwort das zur Client ID gehört.

Nach der Eingabe aller Daten und der Auswahl des **Weiter >** Buttons wird die Verbindung zum Index Proxy geprüft. Sollte es zu einem Fehler kommen, wird die folgende Meldung angezeigt.



Schaltfläche	Beschreibung
Abort	Die Prüfung kann abgebrochen, die Werte korrigiert und dann erneut geprüft werden.
Retry	Erneuter Versuch mit den aktuellen Werten
Ignore	Der Dialog kann ohne positives Prüfergebnis verlassen werden. Die Installation wird mit den im Dialog angegebenen Werten durchgeführt.

## 3.2.9 Web Logins

### Spider

Der Spider Web Login ist ein Datenbankbenutzer, der für den Zugriff der Produktkomponenten auf die Datenbanken benötigt wird. Dieser wird vom Spider Web und den Spider Services verwendet.

Für das Web Login können sowohl neue Benutzer erstellt, oder aber bestehende Benutzer verwendet werden, sofern der SQL-Installationsbenutzer die Rechte zur Anlage eines neuen Logins auf dem Datenbankserver besitzt.

**Bestehenden Benutzer verwenden**

Feld	Beschreibung
Bestehenden Benutzer verwenden	Radio Button, auswählen um die Daten eines bestehenden Benutzers anzugeben
Benutzername	Dropdown aus dem die auf dem SQL Server gefundenen Benutzer ausgewählt werden können.
Passwort	Passwort zu dem gewählten Benutzer

## Neuen Benutzer erstellen

Feld	Beschreibung
Neuen Benutzer erstellen	Radio Button, auswählen um einen neuen Benutzer durch das Setup erstellen zu lassen.
Benutzername	Gewünschter Benutzername
Passwort	Passwort mit dem der Benutzer angelegt werden soll..

**Hinweis** Die Möglichkeit einen Benutzer anzulegen besteht nur, wenn die entsprechenden Rechte vergeben wurden. Die Benutzer müssen vorab auf dem Datenbankserver mit den entsprechenden Rechten eingerichtet werden.

Bitte wählen Sie ein Passwort das den Komplexitätsrichtlinien ihrer Domain und/oder Ihres SQL-Servers entspricht.

## SpiderAdmin

Der Spider Admin Login wird für den Zugriff auf das Admin-Tool zur Administration der Datenbank im Sinne der Businessschicht verwendet.

Sofern der SQL-Installationsbenutzer die Rechte zur Anlage eines neuen Logins auf dem Datenbankserver besitzt, kann an dieser Stelle ein neuer Benutzer angelegt werden. Andernfalls kann ein bestehender Benutzer verwendet werden.

**Bestehenden Benutzer verwenden**

Feld	Beschreibung
Bestehenden Benutzer verwenden	Radio Button, auswählen um die Daten eines bestehenden Benutzers anzugeben
Benutzername	Dropdown aus dem die auf dem SQL Server gefundenen Benutzer ausgewählt werden können.
Passwort	Passwort zu dem gewählten Benutzer

## Neuen Benutzer erstellen

Feld	Beschreibung
Neuen Benutzer erstellen	Radio Button, auswählen um einen neuen Benutzer durch das Setup erstellen zu lassen.
Benutzername	Gewünschter Benutzername
Passwort	Passwort mit dem der Benutzer angelegt werden soll..

**Hinweis** Die Möglichkeit einen Benutzer anzulegen besteht nur, wenn die entsprechenden Rechte vergeben wurden. ie Benutzer müssen vorab auf dem Datenbankserver mit den entsprechenden Rechten eingerichtet werden.

Bitte wählen Sie ein Passwort das den Komplexitätsrichtlinien ihrer Domain und/oder Ihres SQL-Servers entspricht.

### 3.2.10 Datenbanknamen

Die Datenbanken können entweder vorab, oder durch das Setup erstellt werden.

**Hinweis** Die neuen Datenbanken werden mit den Standardeinstellungen des SQL-Servers angelegt. Wenn spezielle Einstellungen gewünscht sind, muss die Datenbank vorab auf dem SQL-Server bereitgestellt werden.

Spider unterstützt nur Datenbanknamen bis zu einer Länge von 50 Zeichen.

**Vorab erstellte Datenbanken auswählen**

**S** Spider 6.3 Setup

**Datenbanken**  
Bitte wählen Sie die Datenbanken

Bitte wählen Sie eine existierende Datenbank oder geben sie die Namen zur Erstellung neuer Datenbanken an.

Core: SpiderCore

Asset: SpiderAsset

Contract: SpiderContract

Licence: SpiderLicence

Integration: SpiderIntegration

Prefix:

Suffix:

© Brainware Consulting & Development AG

< Zurück **Weiter >** Abbrechen

Aus dem jeweiligen Drop Down werden die auf dem SQL Server ermittelten Datenbanken für das jeweilige Produkt ausgewählt.

**Hinweis** Die Integration Datenbank steht nur bei bestimmten Lizenzausprägungen zur Verfügung.

**Folgende Datenbanken werden für die Installation benötigt:**

- Core
- Core\_TempDB
- Core\_SessionState
- Asset
- Licence
- Contract
- (Integration)

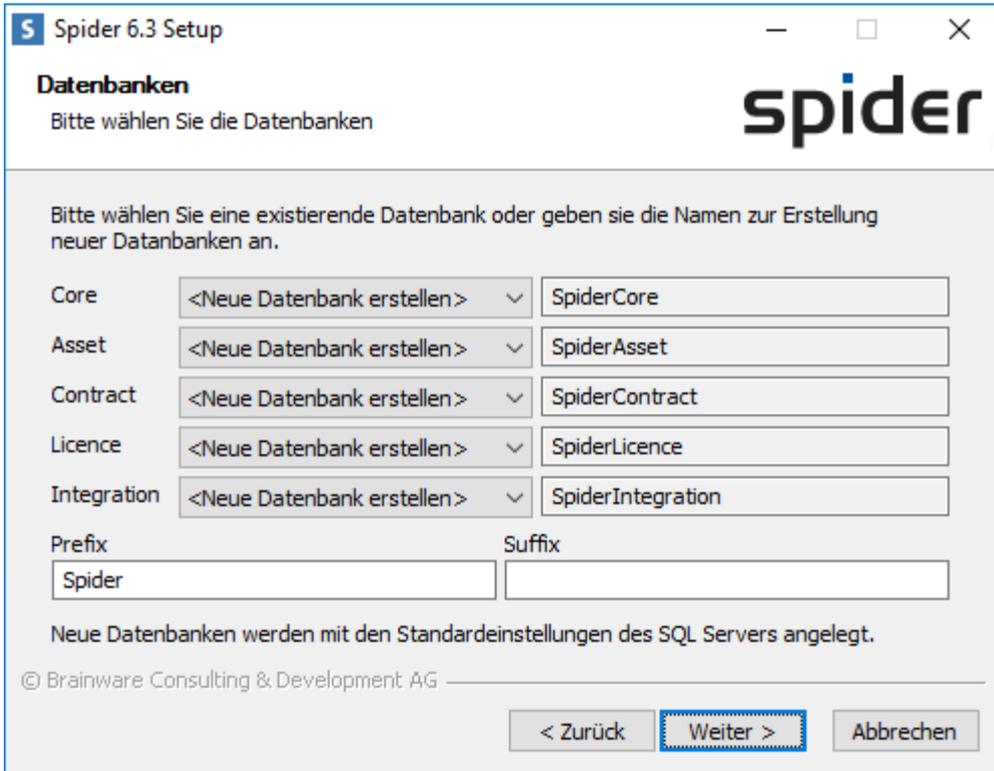
**Achtung** Die Namen der Datenbanken können frei gewählt werden, wenn die Datenbanken vorab angelegt werden, ist darauf zu achten, dass die TempDB Datenbank und Sessionstate Datenbank nach dem Muster

<Name der Core Datenbank>\_TempDB  
<Name der Core Datenbank>\_SessionState

angelegt werden.

Die länge des Datenbanknamens darf 50 Zeichen nicht übersteigen.

**Datenbanken durch das Setup erstellen lassen.**



Feld	Beschreibung
Core, Asset, Contract, Licence, Integration	Namen der jeweiligen Produktdatenbank die auszuwählen sind.
Prefix	Präfix der allen(!) Datenbanken voran gestellt wird.
Suffix	Suffix der allen(!) Datenbanken hintenan gestellt wird.

Die Namen der Produktdatenbanken sind mit

- Core
- Asset
- Contract
- Licence
- Integration

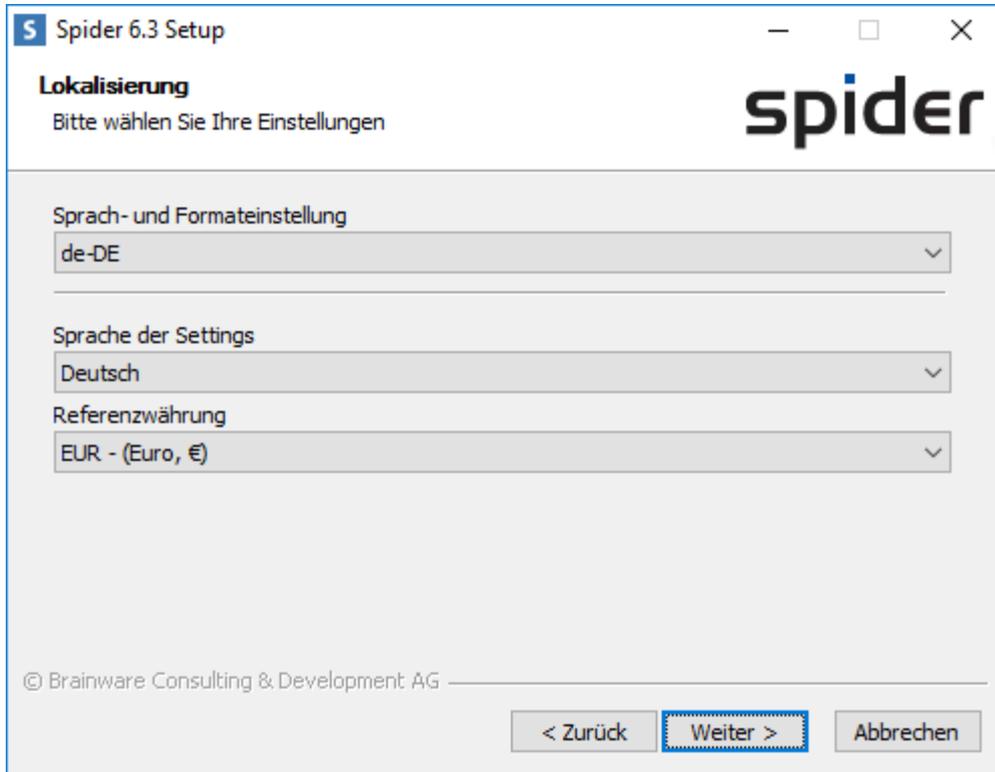
vorbelegt, Änderungen können über die Anpassung des Präfix oder Suffix erreicht werden.

**Achtung** Theoretisch ist auch eine Mischung zwischen Neuanzulegenden und vorhandenen Datenbanken möglich. Es wird allerdings davon abgeraten eine gemischte Erstellung von Datenbanken zu verwenden.

### 3.2.11 Sprach- und Formateinstellungen

Die Spracheinstellungen für die Settings, bestimmt mit welcher Grundsprache die Settings installiert werden. Diese Einstellung kann anschließend nicht mehr geändert werden.

Die Standard-Culture für Spider kann auch im Anschluss noch geändert werden.



The screenshot shows the 'Spider 6.3 Setup' window with the 'Lokalisierung' (Localization) section. The window title is 'Spider 6.3 Setup' and the 'spider' logo is visible in the top right. The instructions say 'Bitte wählen Sie Ihre Einstellungen' (Please select your settings). The settings are as follows:

Setting Name	Selected Value
Sprach- und Formateinstellung	de-DE
Sprache der Settings	Deutsch
Referenzwährung	EUR - (Euro, €)

At the bottom, there are three buttons: '< Zurück' (Back), 'Weiter >' (Next), and 'Abbrechen' (Cancel). The 'Weiter >' button is highlighted with a blue dashed border. The copyright notice '© Brainware Consulting & Development AG' is visible at the bottom left.

### 3.2.12 SMTP Einstellungen

Über den SMTP-Server werden die unterschiedlichen E-Mail-Erinnerungen aus Spider verschickt. Es wird ein Benutzer für den Zugriff auf den SMTP-Server benötigt.

Feld	Beschreibung
SMTP Server	FQDN des SMTP Servers
SMTP Benutzer	Benutzer ID die verwendet werden soll um sich gegenüber dem SMTP Server zu authentifizieren
Passwort	Das Passwort des o.g. Benutzers
Absender	Absende Adresse unter der die Emails von Spider versendet werden.
SSL verwenden	Falls ausgewählt, wird die Kommunikation mit dem SMTP Server per SSL versucht (sofern der SMTP Server dies unterstützt)

**Hinweis** Diese Angaben werden in jedem Produkt eingetragen und von den Spider-Diensten verwendet um E-Mails zu verschicken. Eine spätere Einrichtung oder Änderung der SMTP-Einstellungen muss daher in jedem Produkt erfolgen.

### 3.2.13 Administrationskonto

Das Administrator Konto, ist das Konto, mit der Spider verwaltet werden kann. Dieses Konto bekommt administrative Rechte in Teilen von Spider und steht nach der Installation für den Zugriff des <Spider>-Webs zur Verfügung.

Feld	Beschreibung
Login	Login des Benutzers mit dem sich am Spider-Web angemeldet werden kann.
Benutzername	Name des Benutzers
Passwort	Das Passwort mit dem der Login angelegt wird.

### 3.2.14 IIS Konfiguration

Sie können den IIS Applikationsnamen für das Spider-Web festlegen, im Standard lautet er „Spider“. Das Spider Web ist dann unter <http://servername/Spider> erreichbar.

Alle anderen Einstellungen ergeben sich aus dem gewählten Applikationsnamen und können nicht eigenständig angepasst werden. Nur die Verschlüsselung der Konfigurationsdateien kann in diesem Schritt gesetzt werden.

Feld	Beschreibung
IIS Applikationsname	Name den die IIS Applikation im Rahmen der Installation erhält.
Konfigurationsdateien verschlüsseln.	Mit Auswahl des Kontrollkästchens werden die Details die zur Verbindung des <Spider>-Webs zur Datenbank abgelegt werden verschlüsselt.

**Hinweis** Details zur Ver- und Entschlüsselung der IIS Konfiguration kann im Kapitel [Verschlüsselung der Konfiguration](#) (auf Seite 88) nachgelesen werden.

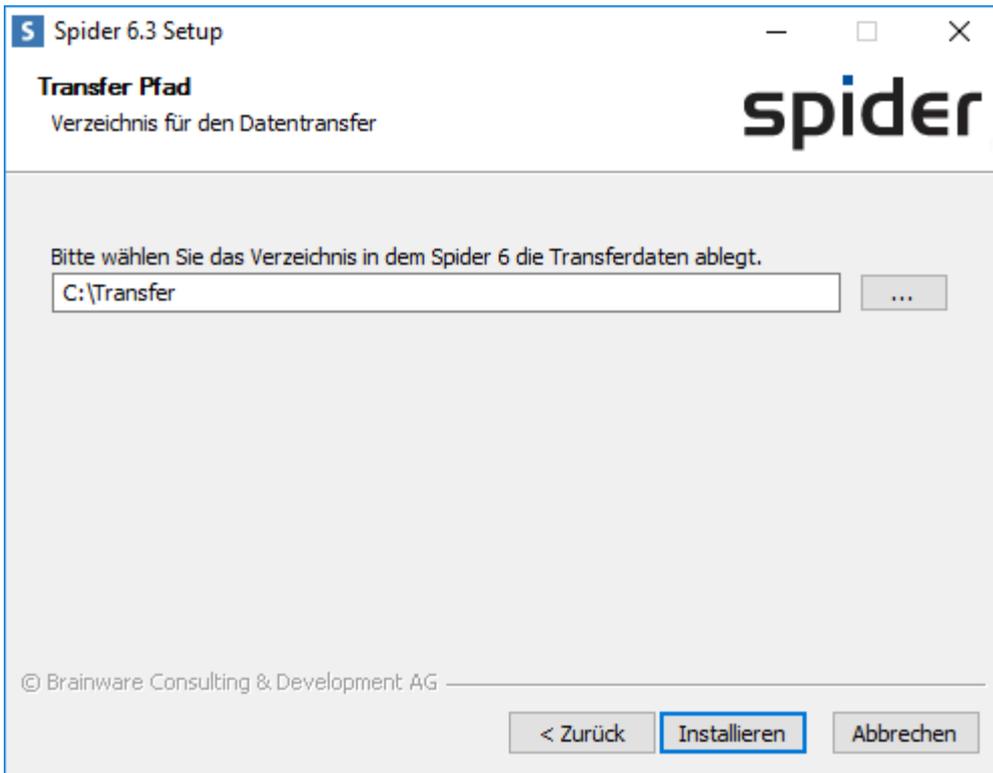
Bei großen Umgebungen oder wiederholten timeout der Verbindung, kann es sinnvoll sein die Parameter **Max Pool Size=50000;Pooling=True;** in die Webconfig aufzunehmen.

### 3.2.15 Transferverzeichnis

Im letzten Schritt der Installation werden Sie gebeten, ein Verzeichnis für den Datentransfer anzugeben.

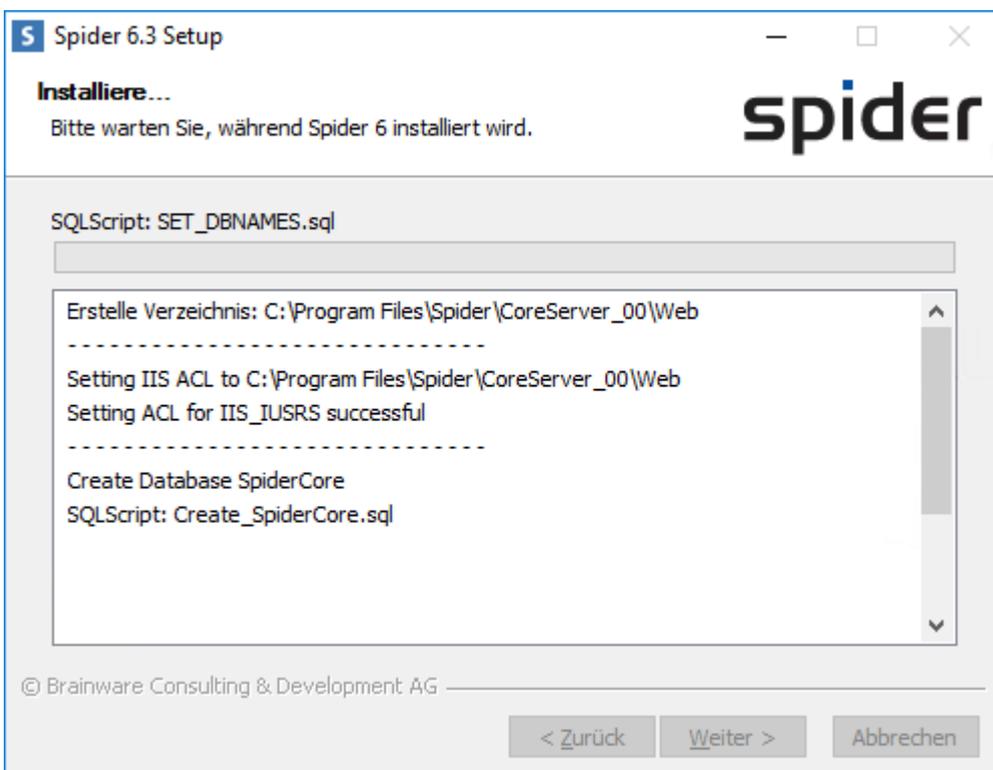
Das Transferverzeichnis stellt den Arbeitspfad für die Imports und Exports der verschiedenen Dienste von Spider zur Verfügung.

Klicken Sie anschließend auf „Installieren“ um mit der Installation zu beginnen.

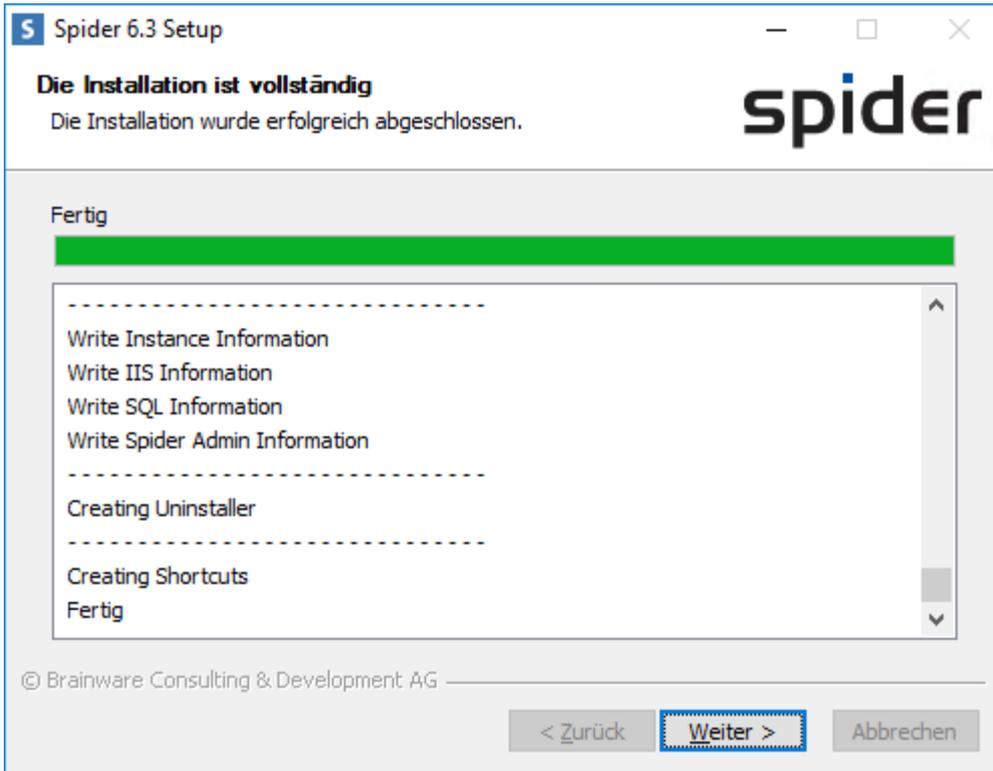


### 3.2.16 Installation

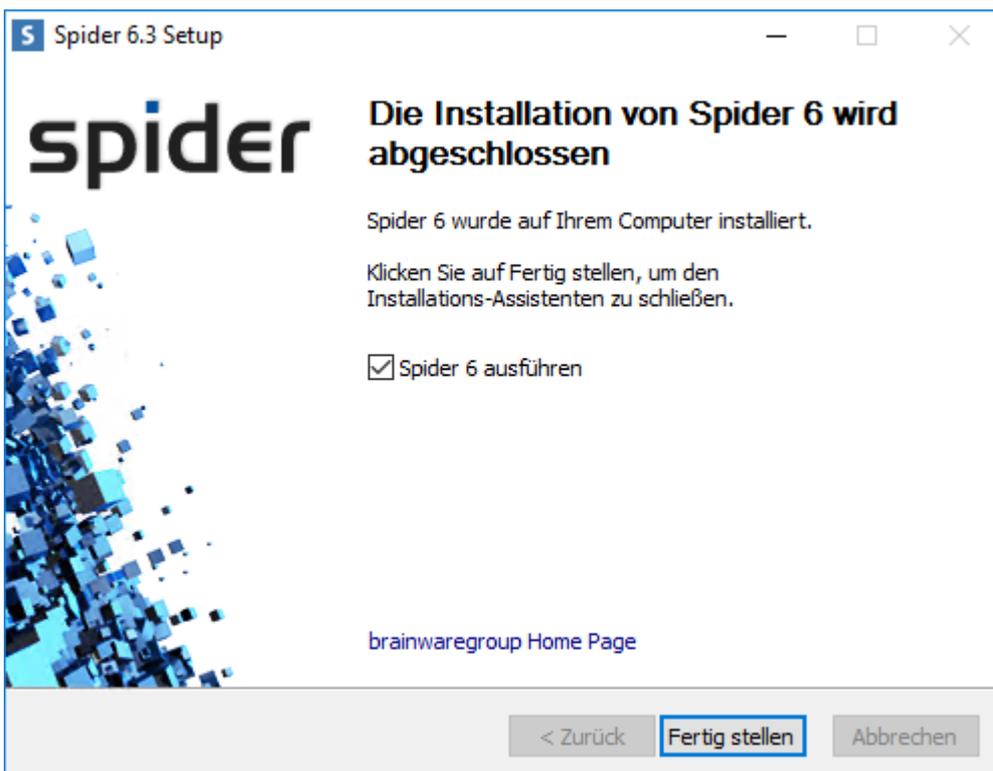
Der Installationsassistent installiert nun Spider. Dies kann einige Zeit in Anspruch nehmen. Bitte warten Sie, bis die Installation beendet ist.



Nach erfolgreichem Abschluss der Installation wird dies im Fortschrittsdialog angezeigt.



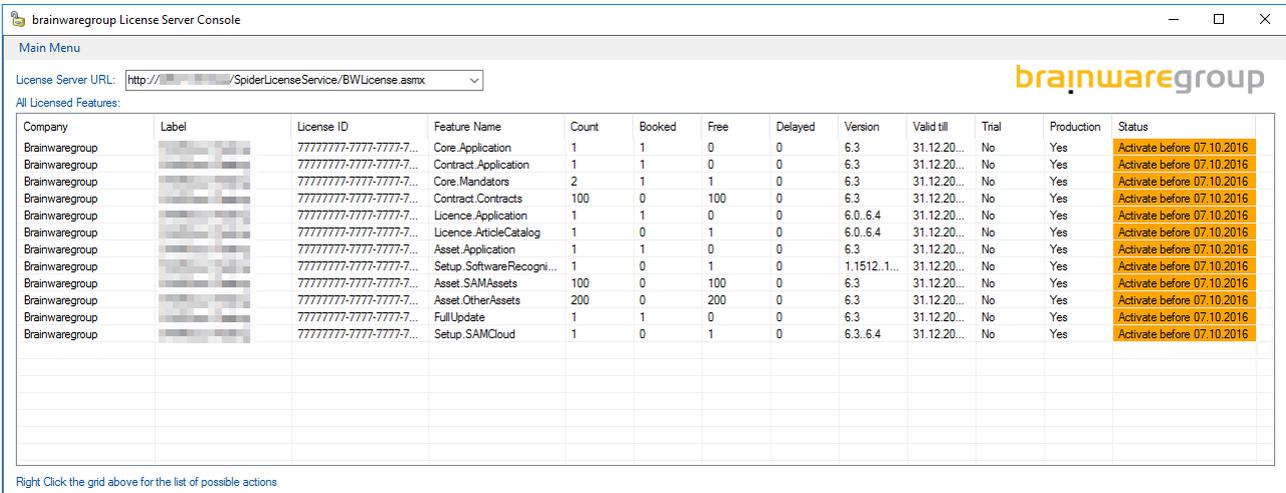
Das Setup ist erfolgreich abgeschlossen und kann beendet werden.



Nach Abschluss der Installation wird ein Logfile geschrieben. Details zu den Logfiles, sind im Kapitel [Logfiles](#) (siehe Seite 210) beschrieben.

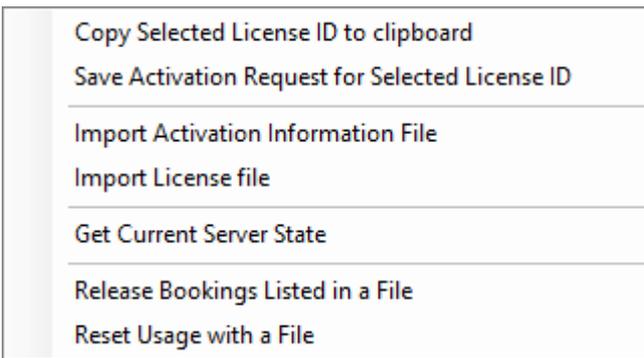
### 3.2.17 Aktivieren der Lizenz

Die Installation von Spider erfordert eine gültige Lizenz. Um Spider nach der Installation verwenden zu können, ist es notwendig, diese Lizenz zu aktivieren. Die Aktivierung der Lizenz erfolgt über die Lizenzkonsole (via Startmenü -> Spider -> License Console).



Über das Kontextmenü muss eine Datei zur Anforderung der Lizenzaktivierung generiert werden. Dazu wird der Punkt **Save Activation Information File** gewählt, die resultierende Datei gespeichert und an Flexera geschickt

Flexera stellt dann eine Aktivierungsdatei aus, die über den Punkt **Import Activation Information File** des Kontextmenüs importiert wird.



brainwaregroup License Server Console

Main Menu

License Server URL:  brainwaregroup

All Licensed Features:

Company	Label	License ID	Feature Name	Count	Booked	Free	Delayed	Version	Valid till	Trial	Production	Status
Brainwaregroup	[redacted]	????????-????-????-?	Core.Application	1	1	0	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Contract.Application	1	1	0	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Core.Mandators	2	1	1	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Contract.Contracts	100	0	100	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	License.Application	1	1	0	0	6.0.6.4	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	License.ArticleCatalog	1	0	1	0	6.0.6.4	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Asset.Application	1	1	0	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Setup.SoftwareRecogni...	1	0	1	0	1.1512.1...	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Asset.SAMAssets	100	0	100	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Asset.OtherAssets	200	0	200	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	FullUpdate	1	1	0	0	6.3	31.12.20...	No	Yes	Ok
Brainwaregroup	[redacted]	????????-????-????-?	Setup.SAMCloud	1	0	1	0	6.3.6.4	31.12.20...	No	Yes	Ok

Right Click the grid above for the list of possible actions

Spider ist nun betriebsbereit. Ein erster Login kann auf einem Client mit den im Setup verwendeten Benutzerdaten erfolgen.

## 3.3 Erweiterte Konfiguration

In diesem Kapitel werden erweiterte Konfigurationen für Spider beschrieben.

### 3.3.1 Betrieb mit Domänenkonto

Alternativ zum Betrieb mit SQL Benutzern kann Spider auch mit einem Domainbenutzer betrieben werden.

Dazu werden im Folgenden die nachstehenden Schritte beschrieben

- Vorbereitung des SQL Servers
- Berechtigung des IIS
- IIS Applikationspools anpassen
- Umstellung der Dienstkontos
- Umstellung der Connection Strings

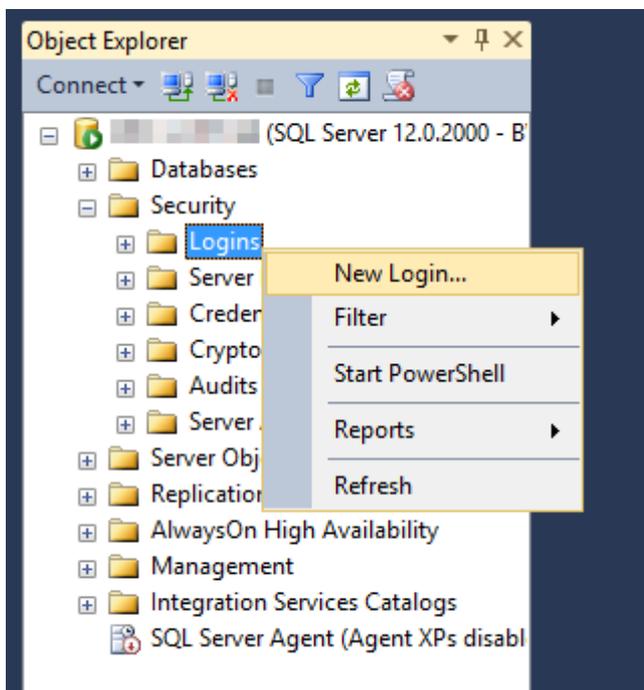
Voraussetzung:

- Ein Domainaccount mit Benutzerrechten in der Domain

#### SQL Server vorbereiten

Das gewünschte Benutzerkonto muss im SQL Server für die Verwendung berechtigt werden.

- **Microsoft SQL Server Management Studio** starten und mit der gewünschten Datenbank [\] verbinden.
- **Rechtsklick auf Security > Logins > New Login**



➤ Eintragen des **Login name:**

**Login - New**

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: domain\svc-techaccount Search...

Windows authentication

SQL Server authentication

Password: [ ]

Confirm password: [ ]

Specify old password

Old password: [ ]

Enforce password policy

Enforce password expiration

User must change password at next login

Mapped to certificate [ ]

Mapped to asymmetric key [ ]

Map to Credential [ ] Add

Mapped Credentials

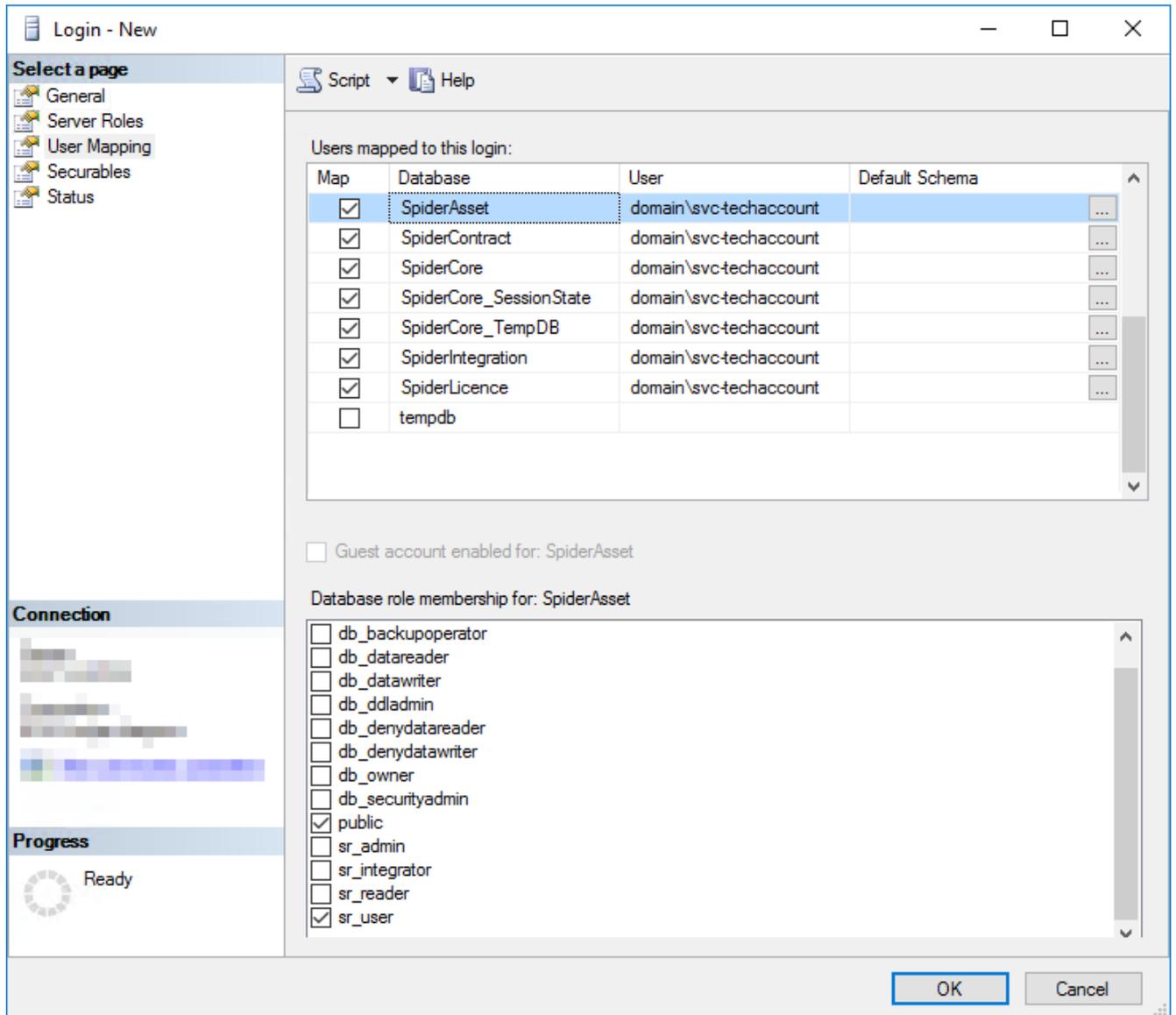
Credential	Provider
------------	----------

Remove

Default database: master

Default language: <default>

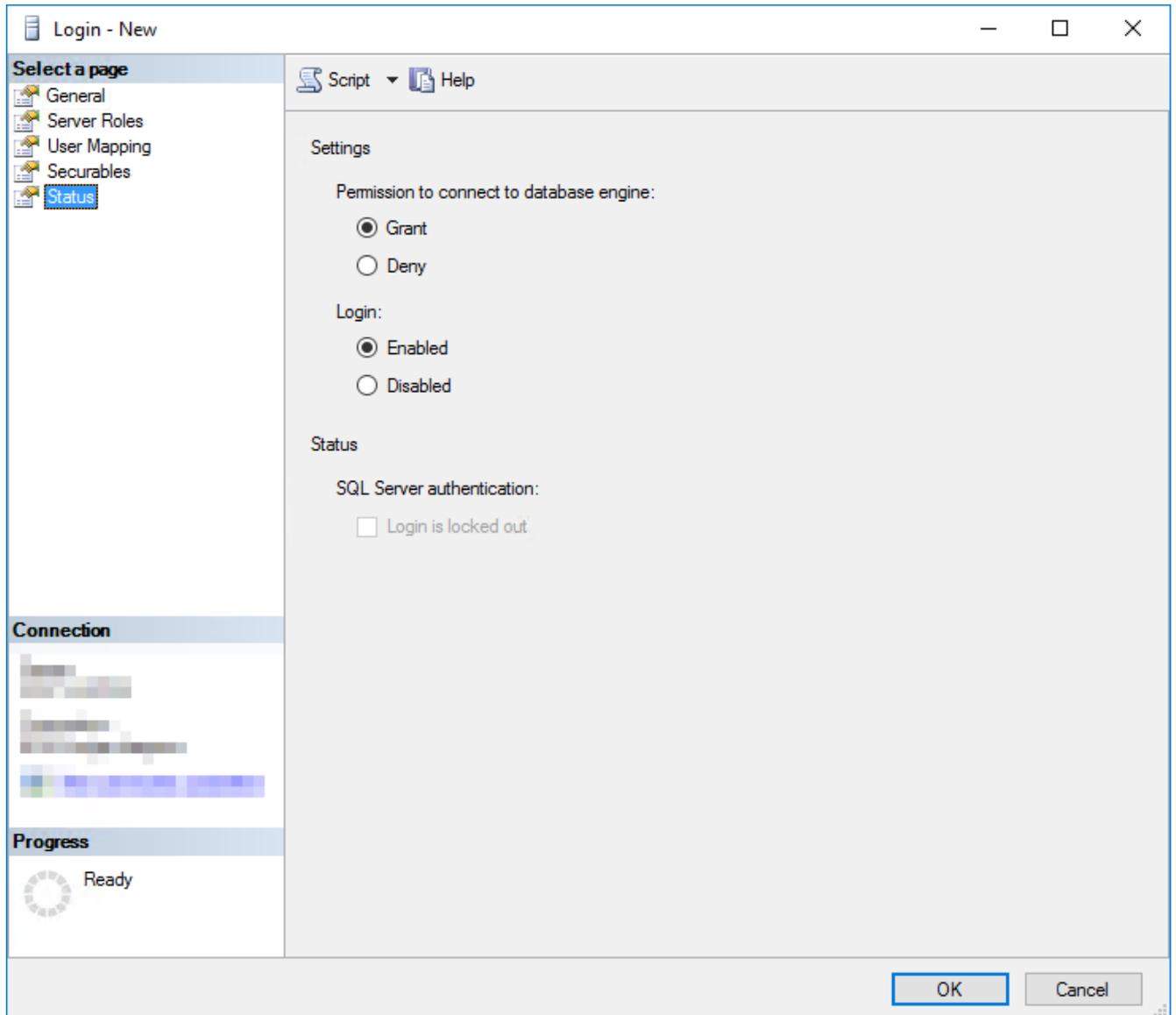
OK Cancel

➤ Wechsel auf den Menüpunkt **User Mapping**➤ Eintragen der **Rollen Mitgliedschaften** für die Spider Datenbanken durch Setzen des Hakens bei **Map** und Auswählen der Mitgliedschaft im unteren Teil des Dialogs.

Datenbank	Database Role Membership
Asset	public, sr_user
Contract	public, sr_user
Core	public, sr_user
Core_SessionState	public
Core_TempDB	public, sr_user
Integration	public, sr_user
Licence	public, sr_user

**Hinweis** Die Integration Datenbank ist nicht in jeder Installation enthalten bzw. kann einen anderen Namen haben.

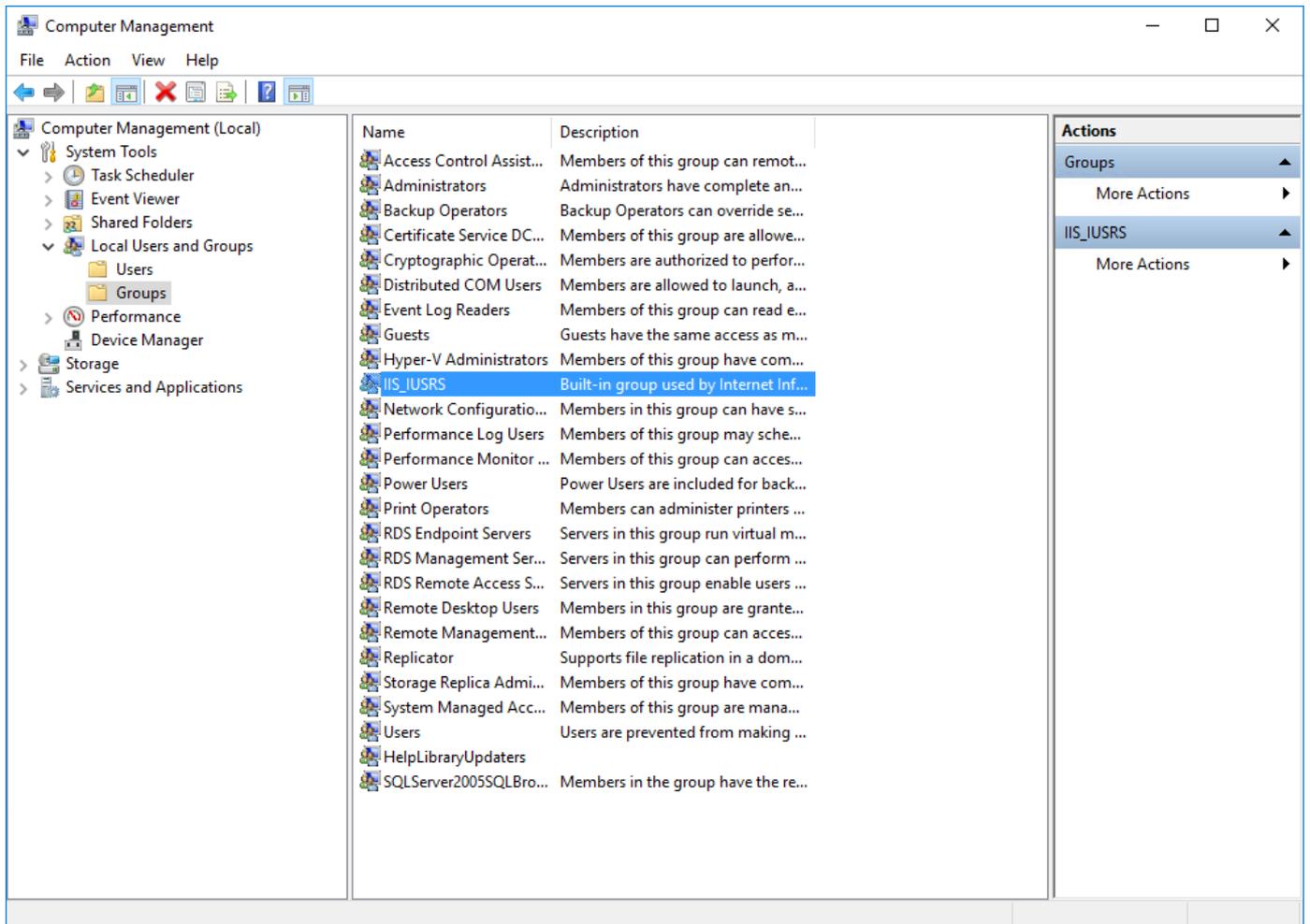
- Wechsel auf den Menüpunkt **Status** und Kontrolle, ob die Berechtigung zum Verbinden erteilt und der Login aktiviert wurde.



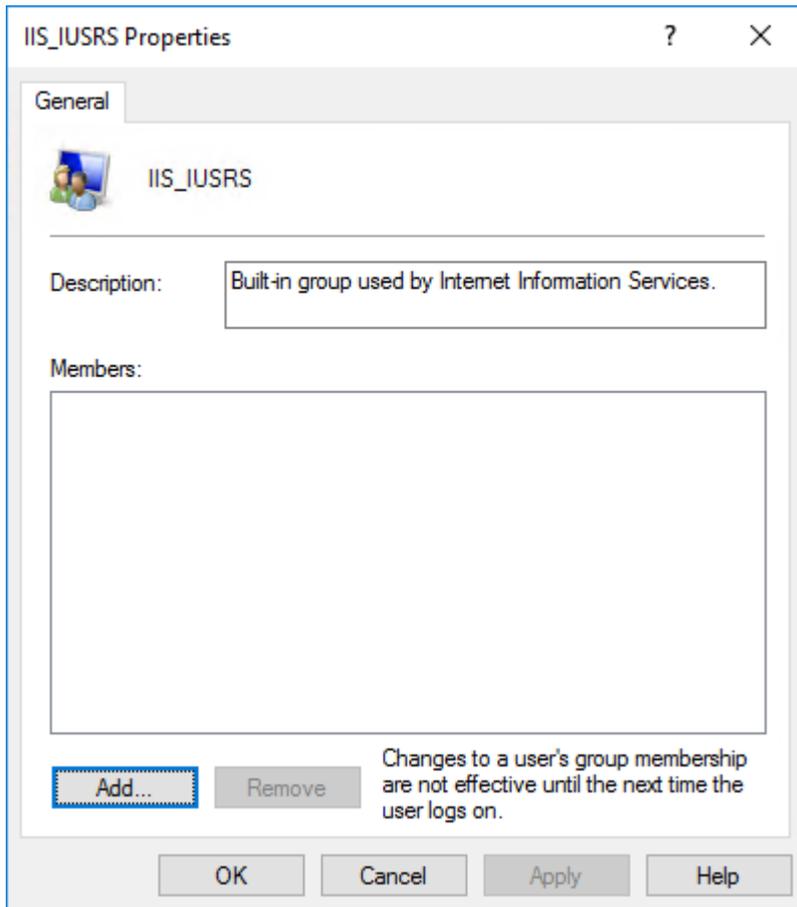
- Verlassen des Dialogs mit **OK**.

## Domänenkonto für den IIS berechtigen

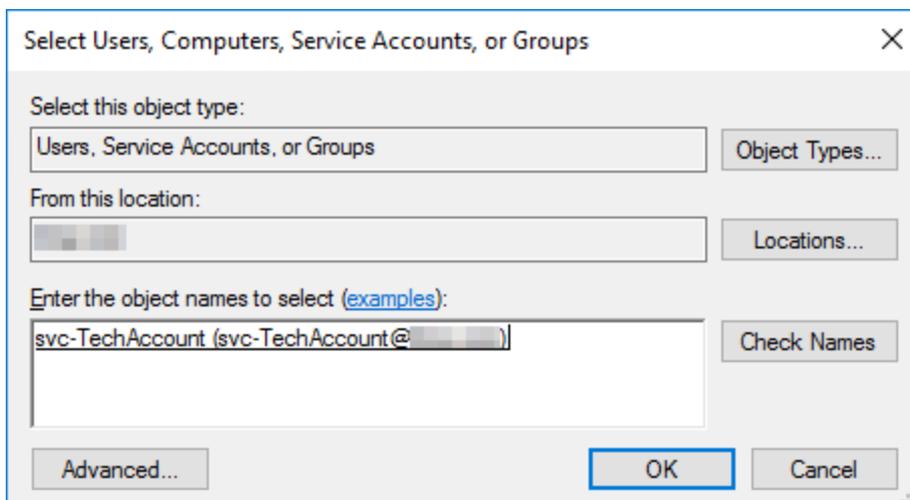
- Starten des Computer Management Control Panels
- **Computer Management (local) > System Tool > Local Users and Groups > Groups**



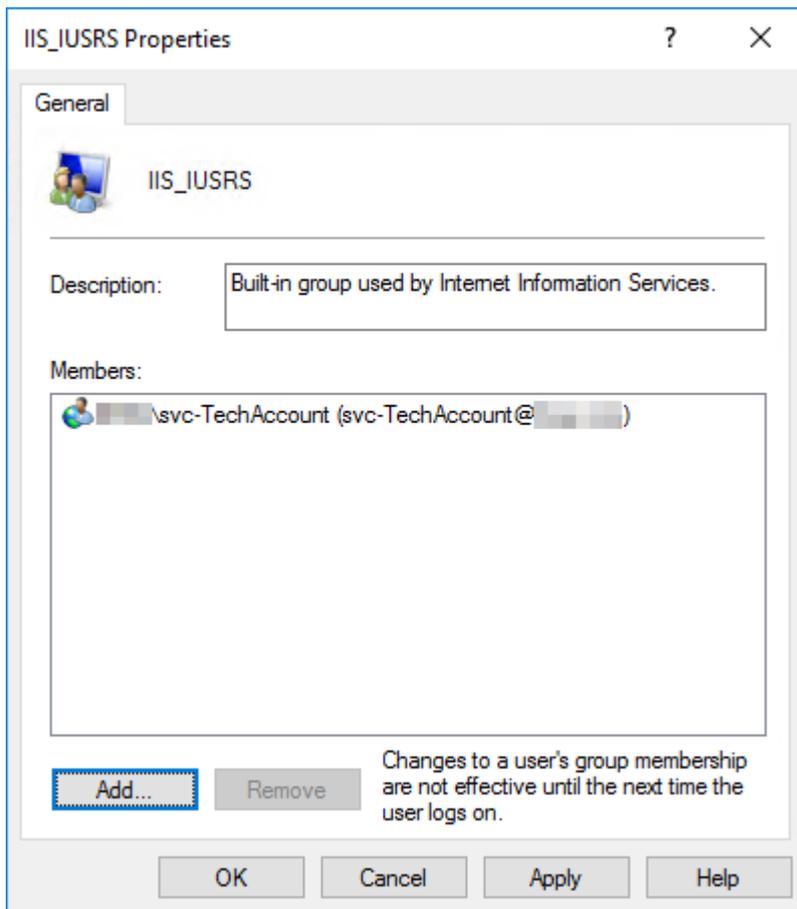
- Doppelklick auf die Gruppe **IIS\_IUSRS**



- **Add...** Schaltfläche klicken



- Den Domainbenutzernamen eintragen und danach mit **Check Names** > **OK** den Dialog verlassen



- Den Dialog mit **OK** verlassen und **Computer Management** schließen

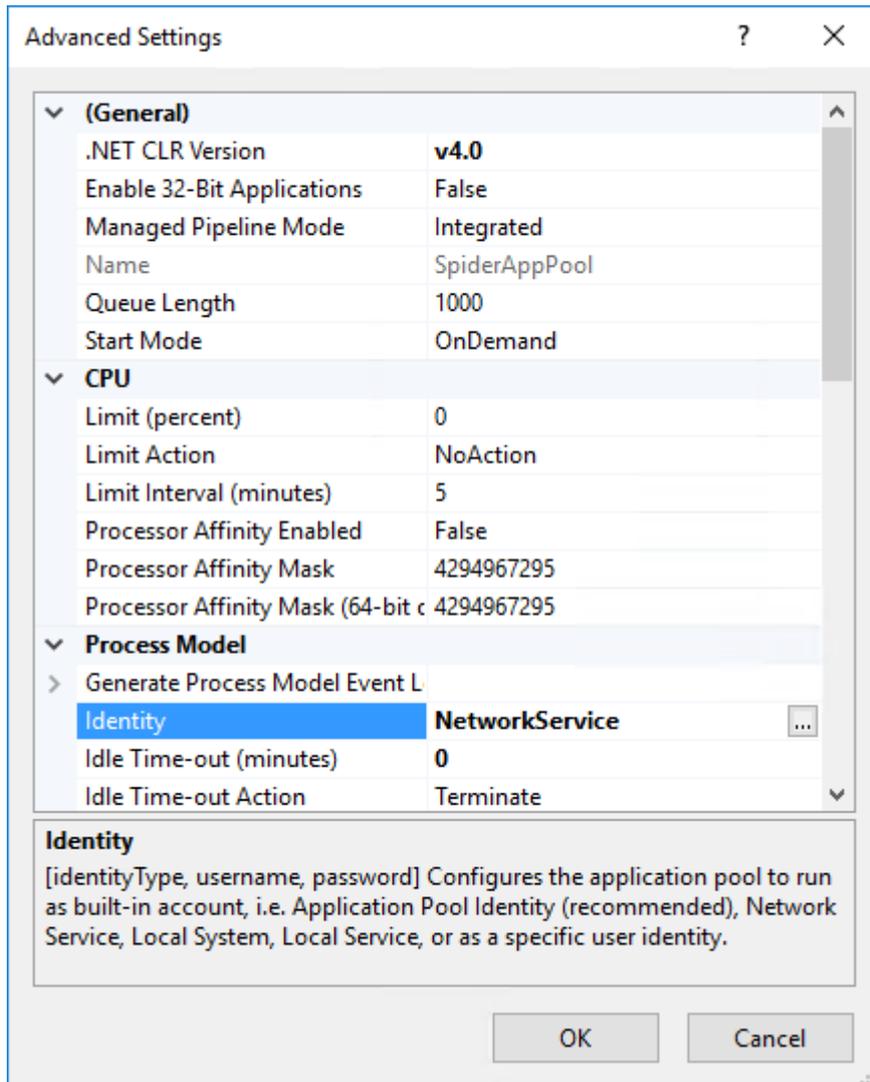
## Identität der Applikationspools anpassen

- Starten des **Internet Information Services (IIS) Managers**
- **Start Page > Application Pools**
- Auswahl des Applikationspools
- Rechtsklick auf den Applikationspool und aus dem Kontextmenü **Advanced Settings...** auswählen (Der Name des Applikationspools wurde während der Installation unter [IIS Konfiguration](#) (auf Seite 38) festgelegt)

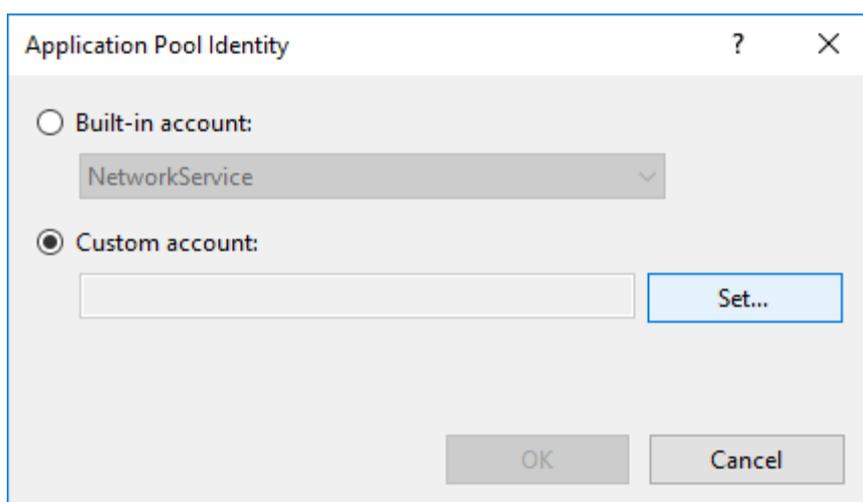
The screenshot shows the IIS Manager interface. On the left, the 'Connections' pane shows the tree structure: Start Page > Application Pools > Sites > Default Web Site > SpiderAppPool. The main pane displays the 'Application Pools' page with a table of application pools. The 'SpiderAppPool' is selected, and a context menu is open over it, with 'Advanced Settings...' highlighted.

Name	Status	.NET CLR V...	Managed Pipel...	Identity	Applications
.NET v2.0	Started	v2.0	Integrated	ApplicationPoolId...	0
.NET v2.0 Classic	Started	v2.0	Classic	ApplicationPoolId...	0
.NET v4.5	Started	v4.0	Integrated	ApplicationPoolId...	0
.NET v4.5 Classic	Started	v4.0	Classic	ApplicationPoolId...	0
Classic .NET AppPool	Started	v2.0	Classic	ApplicationPoolId...	0
DefaultAppPool	Started	v4.0	Integrated	ApplicationPoolId...	1
SpiderAppPool	Started	v4.0	Integrated	NetworkService	1
SpiderIndexAppPool	Started	v4.0	Integrated	NetworkService	1
SpiderLicenseService	Started	v4.0	Integrated	NetworkService	1

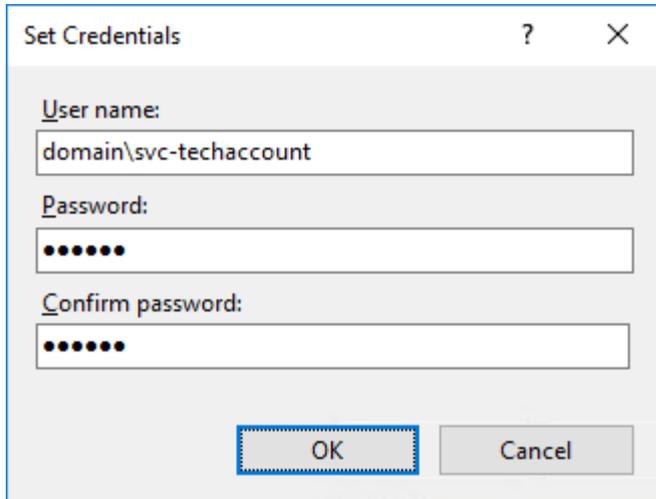
- Unter **Process Model > Identity** die ... Schaltfläche wählen



- Bei Application Pool Identity den Radio Button **Custom Account:** auswählen und die **Set...** Schaltfläche klicken.



- Benutzername und Passwort des Domainbenutzers eingeben und **OK** klicken.



Set Credentials

User name:  
domain\svc-techaccount

Password:  
●●●●●●

Confirm password:  
●●●●●●

OK Cancel

- Alle Dialoge mit **OK** verlassen.
- Alle vorangegangenen Schritte für den Applikationspool mit dem Suffix IndexAppPool wiederholen

SpiderAppPool	Started	v4.0	Integrated	domain\svc-techaccount	1
SpiderIndexAppPool	Started	v4.0	Integrated	domain\svc-techaccount	1

- Zum Abschluss den IIS mit dem Befehl **iisreset** neu starten.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.
C:\Users\>iisreset

Attempting stop...
Internet services successfully stopped
Attempting start...
Internet services successfully restarted

C:\Users\>
```

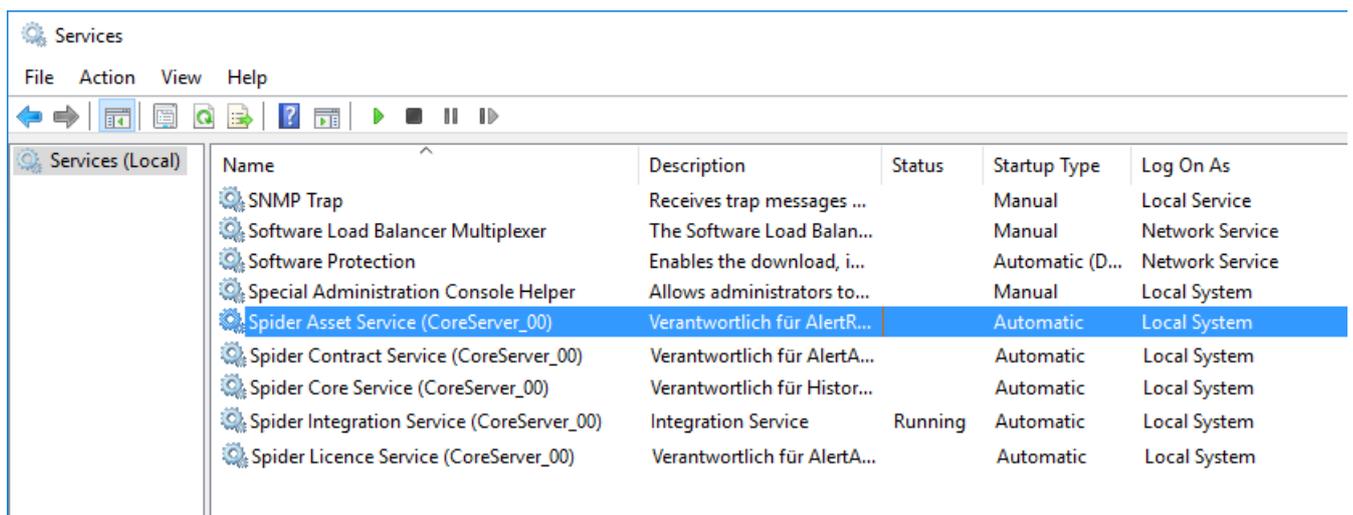
## Umstellen der Zugangsdaten

Damit der Domainbenutzer auch für die Verbindung verwendet wird, muss den unterschiedlichen Komponenten dies mitgeteilt werden.

### Dienstknoten anpassen

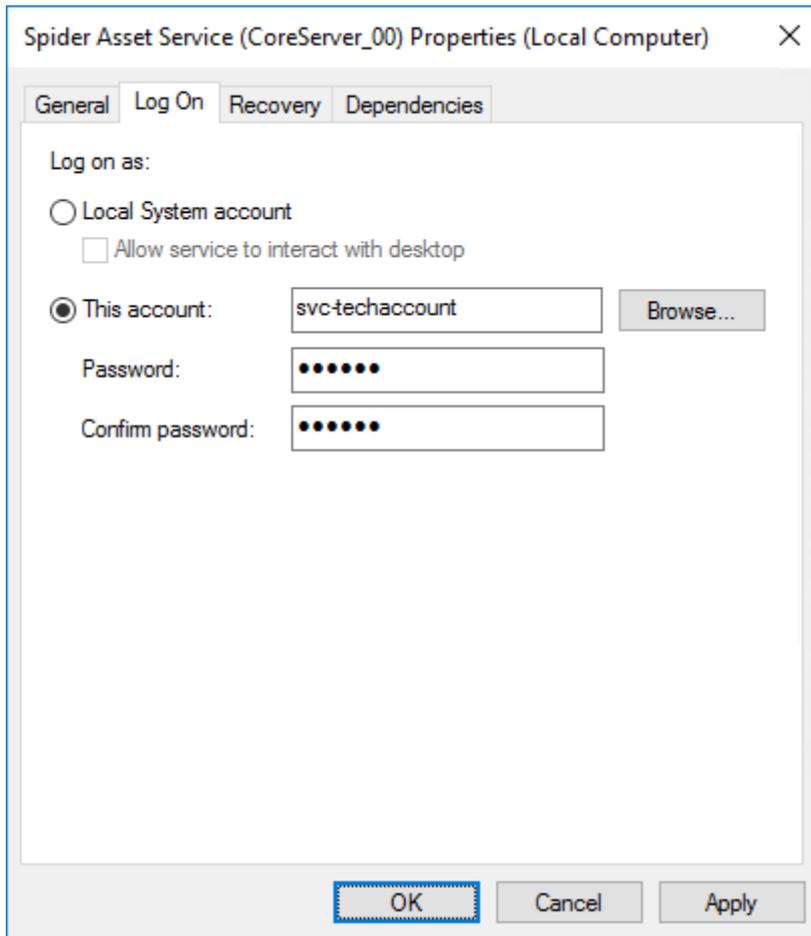
Die Spider Dienste müssen für die Verwendung mit dem Domänenbenutzer angepasst werden.

- Starten des Computer **Management Control Panels**



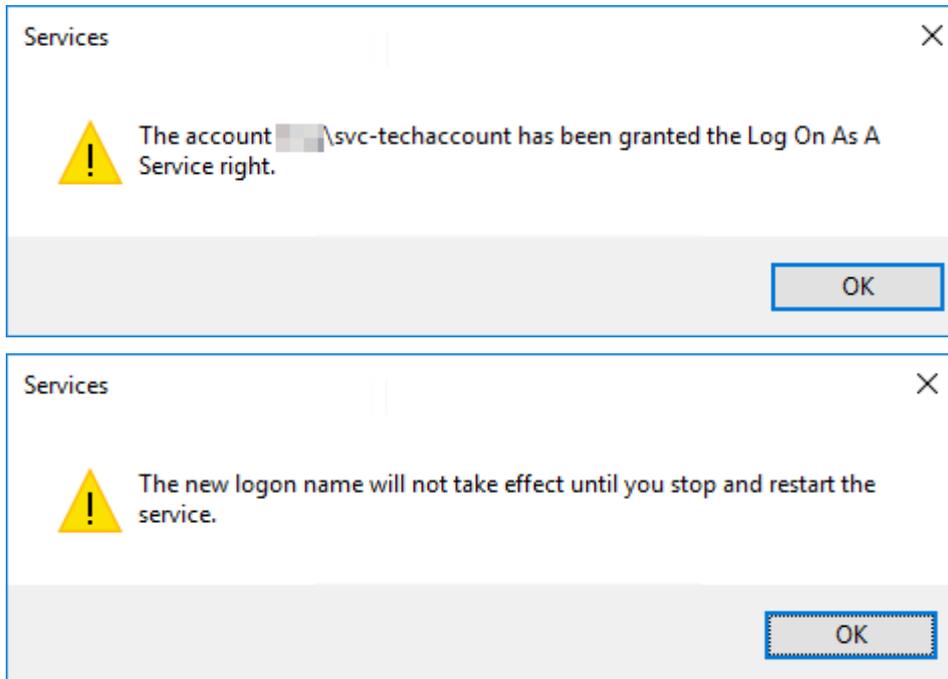
- Spider Service auswählen und mit Doppelklick öffnen

- Auf den Reiter **Log On** wechseln, **This account** auswählen und die Daten des Domänenbenutzers anpassen.



- **OK** klicken und den Dialog verlassen.

**Hinweis** Gegebenenfalls werden Dialoge angezeigt, die darüber informieren, dass dem Benutzer das Log On As A Service Recht gewährt wurde bzw. dass der Dienst neu gestartet werden muss, bevor die Änderung aktiv wird.



➤ Das Vorgehen für alle Spider Dienste wiederholen, die zu dieser Instanz (CoreServer\_xx) gehören.

- Asset
- Contract
- Core
- Licence
- Integration (nicht immer installiert)

**Achtung** Die Dienste müssen jetzt noch nicht gestartet werden, sondern erst nachdem die Connection Strings angepasst wurden.

## Connections Strings anpassen

Für den Zugriff der Applikation und der Services ist die Datei ConnectionStrings.config zuständig. Diese muss für die verschiedenen Komponenten einzeln geändert werden.

Ablageorte der Connection Strings

Komponente	Ablageort
Spider-Web	C:\Program Files\Spider\CoreServer_XX\Web\_Settings\connectionStrings.config
SessionState	C:\Program Files\Spider\CoreServer_XX\Web\_Settings\sessionState.config

Komponente	Ablageort
IndexProxyApi	C:\Program Files\Spider\CoreServer_XX\IndexProxy\_Settings\connectionStrings.config
Dienste	C:\Program Files\Spider\CoreServer_00\Services\ <servicename>\Spider.&lt;ServiceName&gt;.Service.Exe.config</servicename>

Die Einträge uid=... und pwd=... müssen ersetzt werden durch **Integrated Security=True**.

### Spider-Web und IndexProxyApi ConnectionStrings.config

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <add name="CoreConnectionString" connectionString="server=.;
    database=SpiderCore;
    uid=Spider;
    pwd=abcde;
    application name=Spider;
    persist security info=True;
    packet size=4096"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <add name="CoreConnectionString" connectionString="server=.;
    database=SpiderCore;
    Integrated Security=True;
    application name=Spider;
    persist security info=True;
    packet size=4096"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

### Session State Datenbank

```
<?xml version="1.0" encoding="utf-8" ?>
<sessionState mode="SqlServer" sqlConnectionString="Data Source=.;
  Database=SpiderCore_SessionState;
  User Id=Spider;
  Password=abcde"
  cookieless="UseCookies" timeout="30" allowCustomSqlDatabase="true" />
```

```
<?xml version="1.0" encoding="utf-8" ?>
<sessionState mode="SqlServer" sqlConnectionString="Data Source=.;
  Database=SpiderCore_SessionState;
  Integrated Security=True"
  cookieless="UseCookies" timeout="30" allowCustomSqlDatabase="true" />
```

## Dienste

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <appSettings>
    <add key="ServiceName" value="Spider Asset Service (CoreServer_00)" />
    <add key="ConnectionString" value="data source=.;
      initial catalog=SpiderAsset;
      user id=Spider;
      password=abcde;
      persist security info=True;
      Application name=SPAssetSvc_CoreServer_00;
      packet size=4096" />
    <add key="InitializeTimeoutSeconds" value="300" />
  </appSettings>
</configuration>
```

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <appSettings>
    <add key="ServiceName" value="Spider Asset Service (CoreServer_00)" />
    <add key="ConnectionString" value="data source=.;
      initial catalog=SpiderAsset;
      Integrated Security = True;
      persist security info=True;
      Application name=SPAssetSvc_CoreServer_00;
      packet size=4096" />
    <add key="InitializeTimeoutSeconds" value="300" />
  </appSettings>
</configuration>
```

**Achtung** Es kann sein, dass der Inhalt einzelner .config Dateien verschlüsselt ist. Der Prozess zum Entschlüsseln ist im Kapitel [Verschlüsselung der Zugangsdaten](#) (siehe "" auf Seite 88) beschrieben.

### 3.3.2 Nachträgliche Einrichtung Index Proxy

Die nachträgliche Einrichtung der Index Proxy Installation kann über ein reguläres Update erledigt werden. Dort wird bei jeder Aktualisierung zum Einrichten des [Index Proxy](#) (siehe "" auf Seite 27) aufgefordert, solange dies noch nicht geschehen ist.

Sollte dies nicht funktionieren, sind folgende Schritte auszuführen:

#### Konfigurationseinträge

##### Einträge in der [dbo].[config] Tabelle in der Core Datenbank

Area: Application.Indexing

Key	Value
Enabled	True
IndexProxyEndpoint	http://<ESIServer>:<Port>/indexproxy/api/2/ z.B. http://servername.domain:7080/indexproxy/api/2/
PortalClientId	Benutzername der beim ESI Setup vergeben wurde
PortalClientSecret	Passwort zu o.g. Benutzername
PortalEndpoint	http://<ESIServer>:<Port>/portal/api/2/ z.B. http://servername.domain:7080/portal/api/2/

#### IIS Applikation connectionStrings.config

Die IIS Applikation wird parallel zur Instanz von Spider installiert (Pfad ..\CoreServer\_XX\IndexProxy\\_Settings). Damit die Applikation sich korrekt mit der Datenbank verbinden kann, ist eine sog. connectionStrings.config Datei im Unterverzeichnis \_Settings zu erstellen.

**Hinweis** Bei einer manuellen Nachinstallation muss zuerst der Ordner "\_Settings" erstellt werden.

Die benötigten Werte können aus der connectionStrings.config Datei der Spider IIS Applikation entnommen werden (Pfad: Pfad ..\CoreServer\_XX\Web\\_Settings).

#### Login mit Integrated Security

```
<?xml version="1.0" encoding="utf-8" ?>
<connectionStrings>
  <add name="CoreConnectionString"
        providerName="System.Data.SqlClient"
        connectionString="server=<SQL Server>;
                        database=<Core Datenbank>;
                        Integrated Security=True;
                        application name=SpiderIndex;
```

```

        persist security info=True;
        packet size=4096" />
</connectionStrings>

```

### Login mit SQL User

```

<?xml version="1.0" encoding="utf-8" ?>
<connectionStrings>
  <add name="CoreConnectionString"
        providerName="System.Data.SqlClient"
        connectionString="server=<SQL Server>;
                        database=<Core Datenbank>;
                        uid=<SQL User>;
                        pwd=<SQL User Password>
                        Integrated Security=True;
                        application name=SpiderIndex;
                        persist security info=True;
                        packet size=4096" />
</connectionStrings>

```

---

**Hinweis** Bei Verwendung von SQL Benutzern kann es wünschenswert sein, die Angaben lokal zu verschlüsseln. Eine Anleitung dazu ist in [Verschlüsselung der Konfiguration](#) (auf Seite 88) zu finden.

---

## Einstellungen in der Registry

---

Damit das Setup beim nächsten Update nicht nochmal nach der Eingabe der Details fragt, sind in der Registry Einträge zu hinterlegen.

Key: HKEY\_LOCAL\_MACHINE\SOFTWARE\WOW6432Node\brainwaregroup\Spider\CoreServer\_XX\IndexProxy

Name	Type	Data
IndexProxyClientID	REG_SZ	<Benutzername zur Verbindung mit dem ESI> das Passwort wird absichtlich nicht hinterlegt.
IndexProxyServerName	REG_SZ	FQDN des ESI Servers
IndexProxyServerPort	REG_SZ	Port des ESI Servers (Standard: 7080)
UseIndexProxy	REG_SZ	1

### 3.3.3 Umstellung auf SSL Betrieb

---

Zur Absicherung der Kommunikation kann auf Spider per SSL zugegriffen werden.

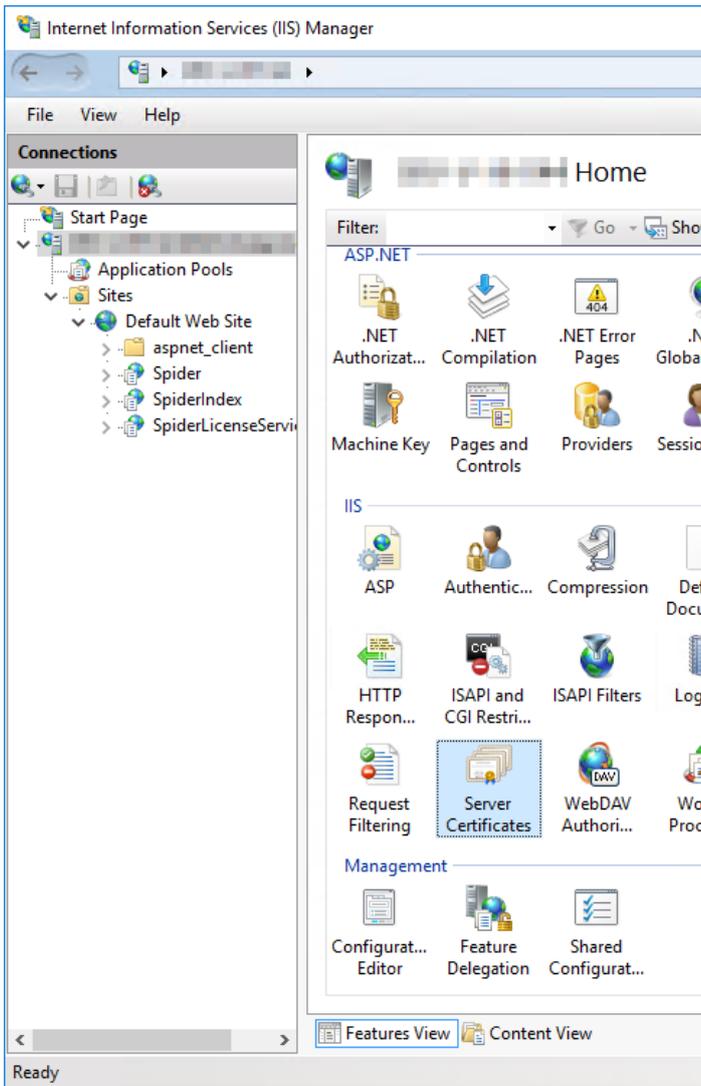
**Achtung** Die IIS Applikation des Licence Service ist nicht SSL fähig. Wie dies als Ausnahme konfiguriert wird, ist im weiteren Verlauf beschrieben.

**Wichtig** Voraussetzung für den Betrieb ist ein gültiges SSL Zertifikat, das auf den Spider Applikationsserver ausgestellt ist

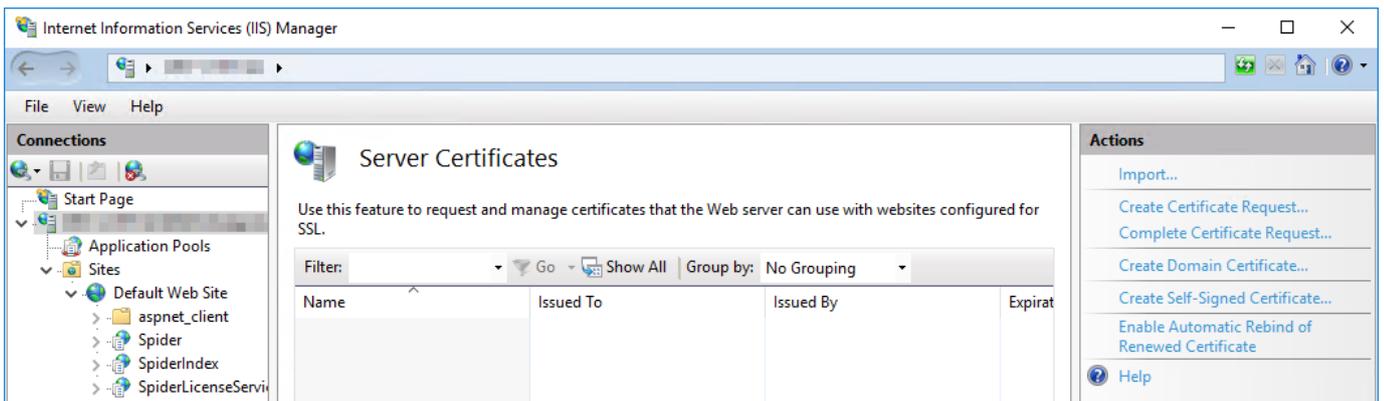
#### IIS Default Web Site konfigurieren

---

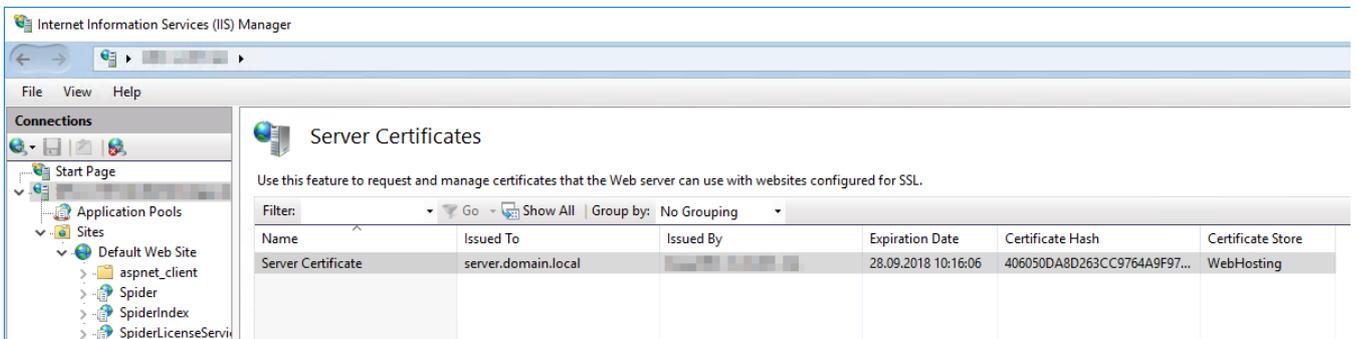
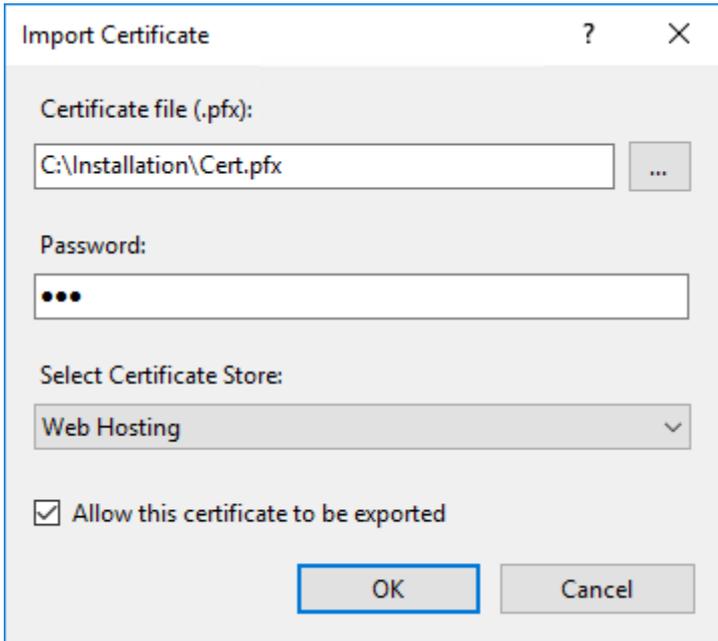
- Starten des Computer **Internet Information Services (IIS) Manager**
- **Server auswählen > Server Certificates**



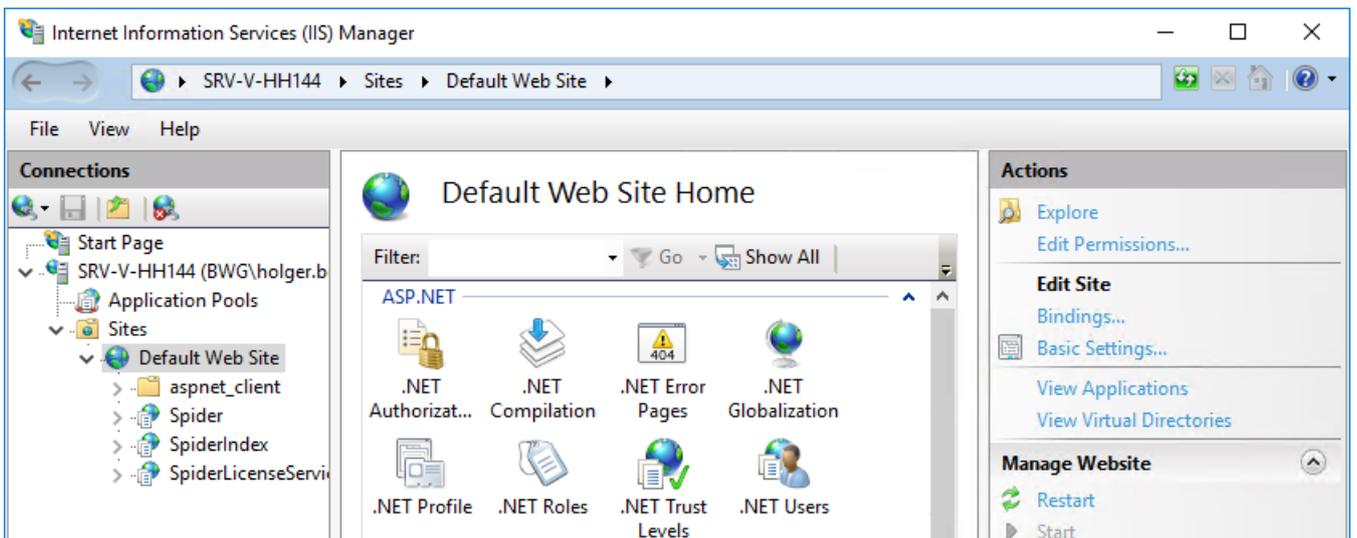
➤ **Actions > Import**

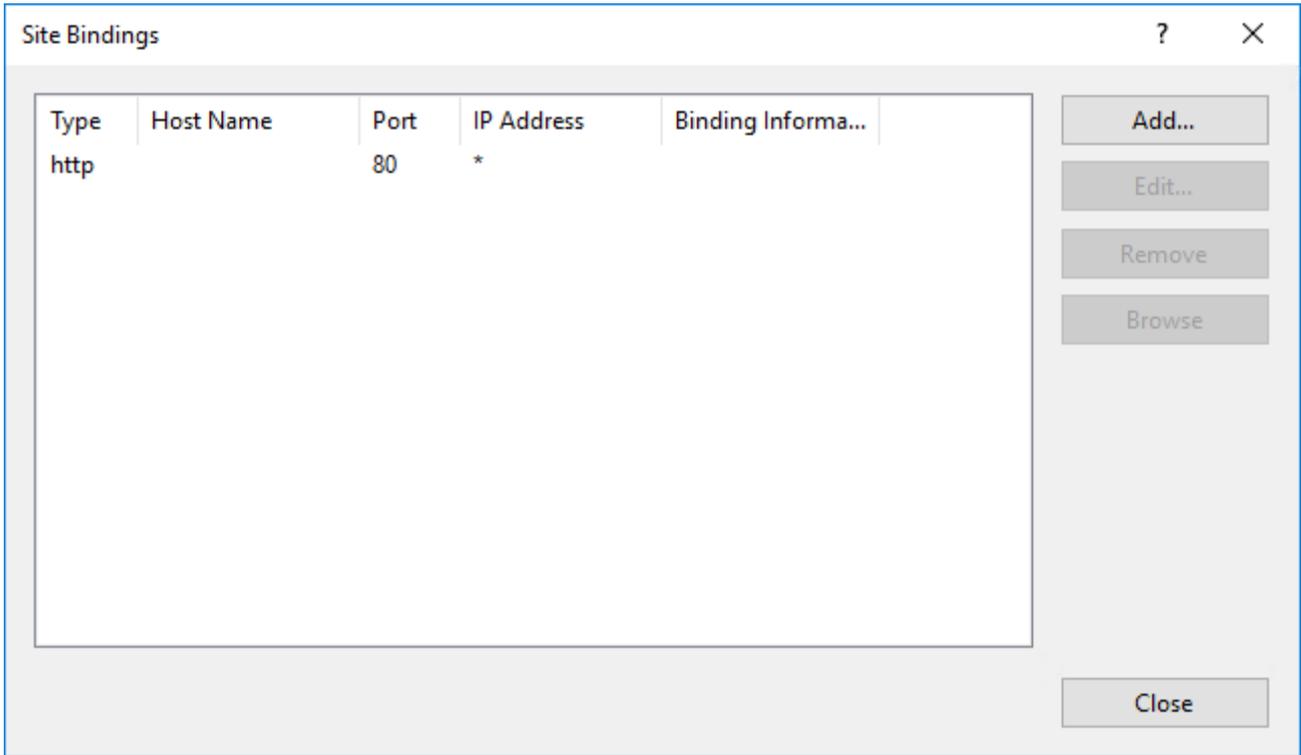


➤ **Auswahl des Zertifikates > Passwort angeben > Select Certificate Store: **Web Hosting** > OK**

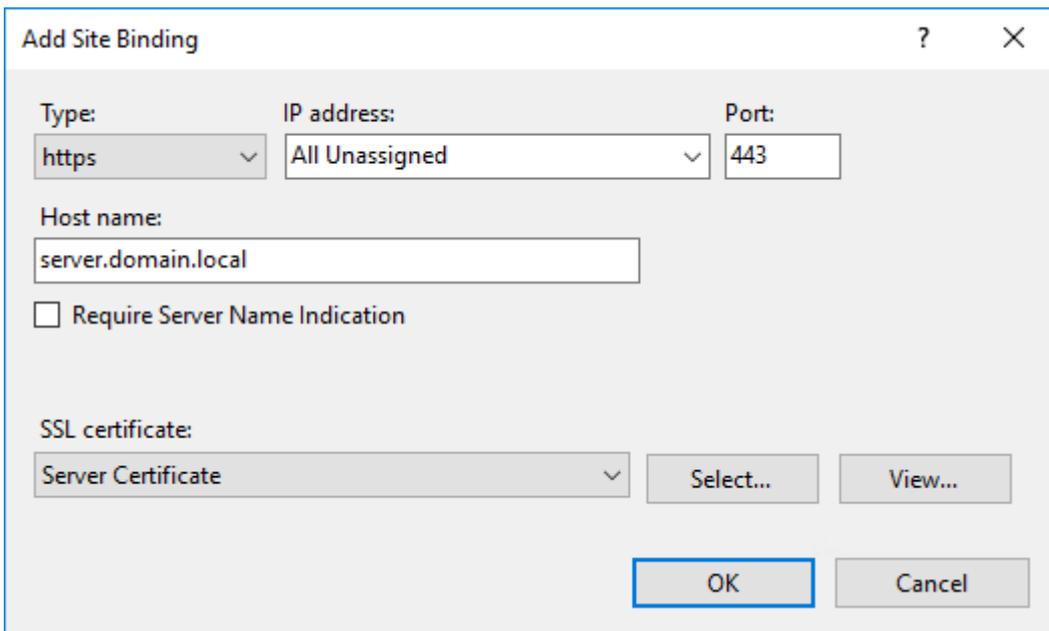


➤ **Default Web Site > Edit Site > Bindings**





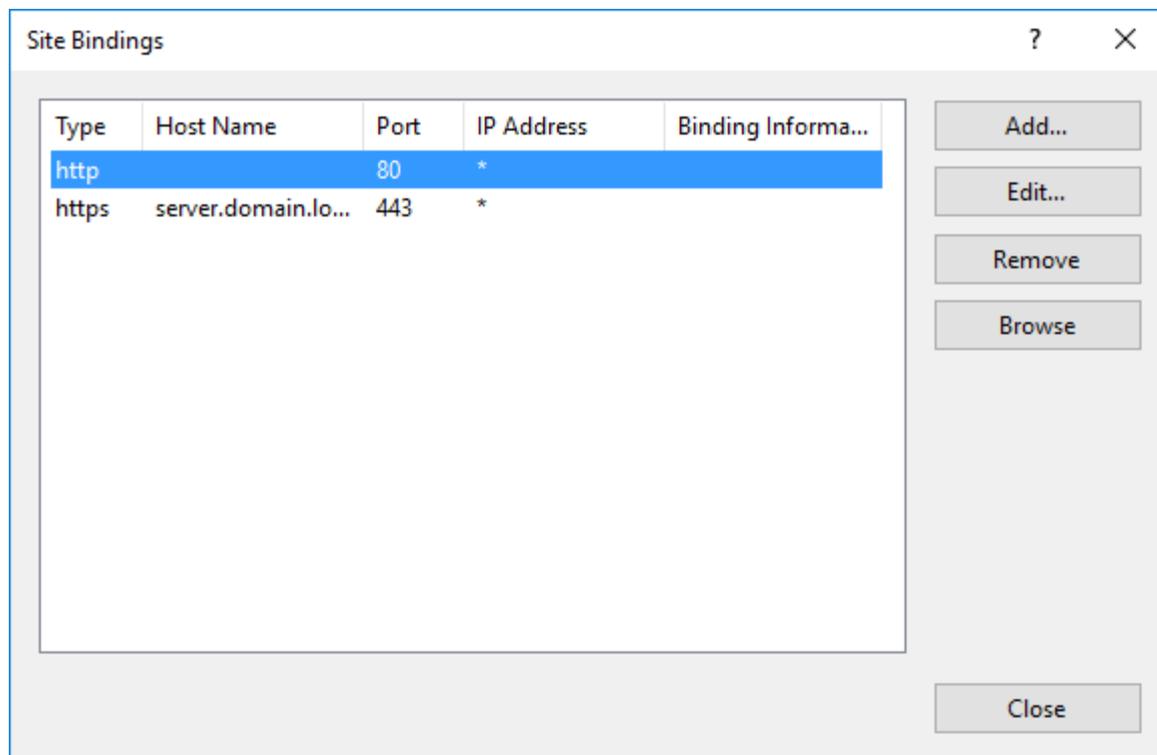
➤ **Add...**



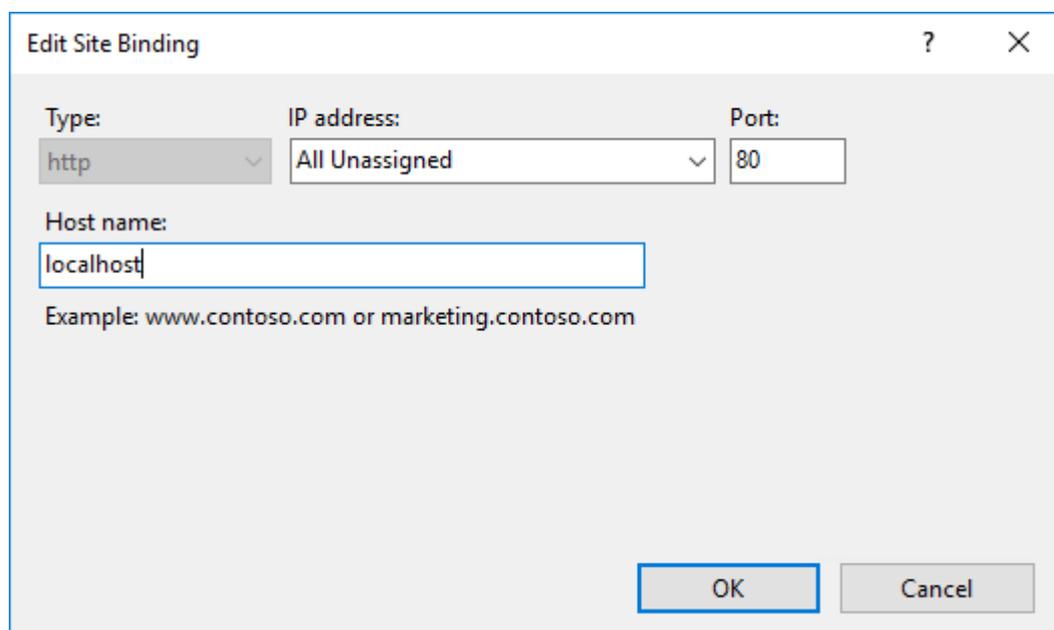
Feld	Inhalt
Type	https
IP address:	All Unassigned, oder aber wenn der Server nur auf einer (von Mehreren konfigurierten) IP Adresse anfragen entgegen nehmen soll.
Host name:	Hostname der im Zertifikat hinterlegt ist

Feld	Inhalt
SSL certificate	Entsprechendes Zertifikat auswählen.

➤ **OK**

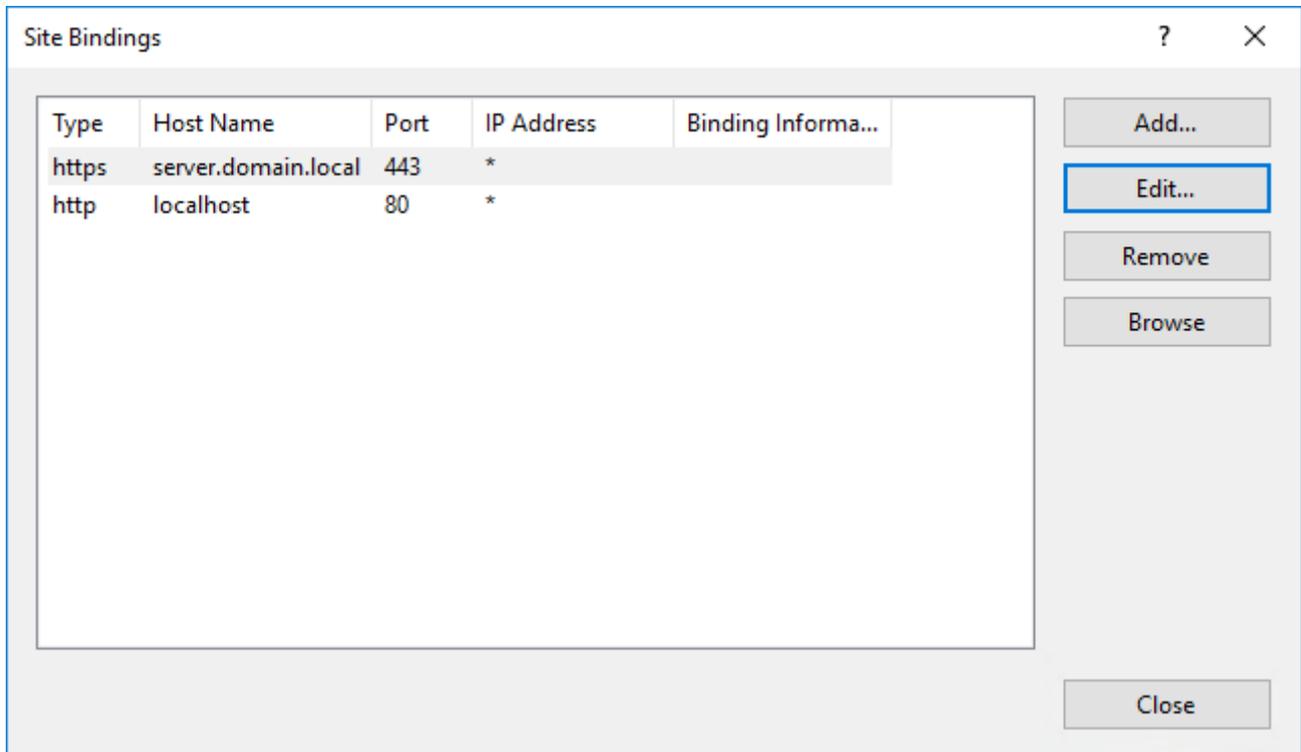


➤ Markieren des **http** Bindings > **Edit...**

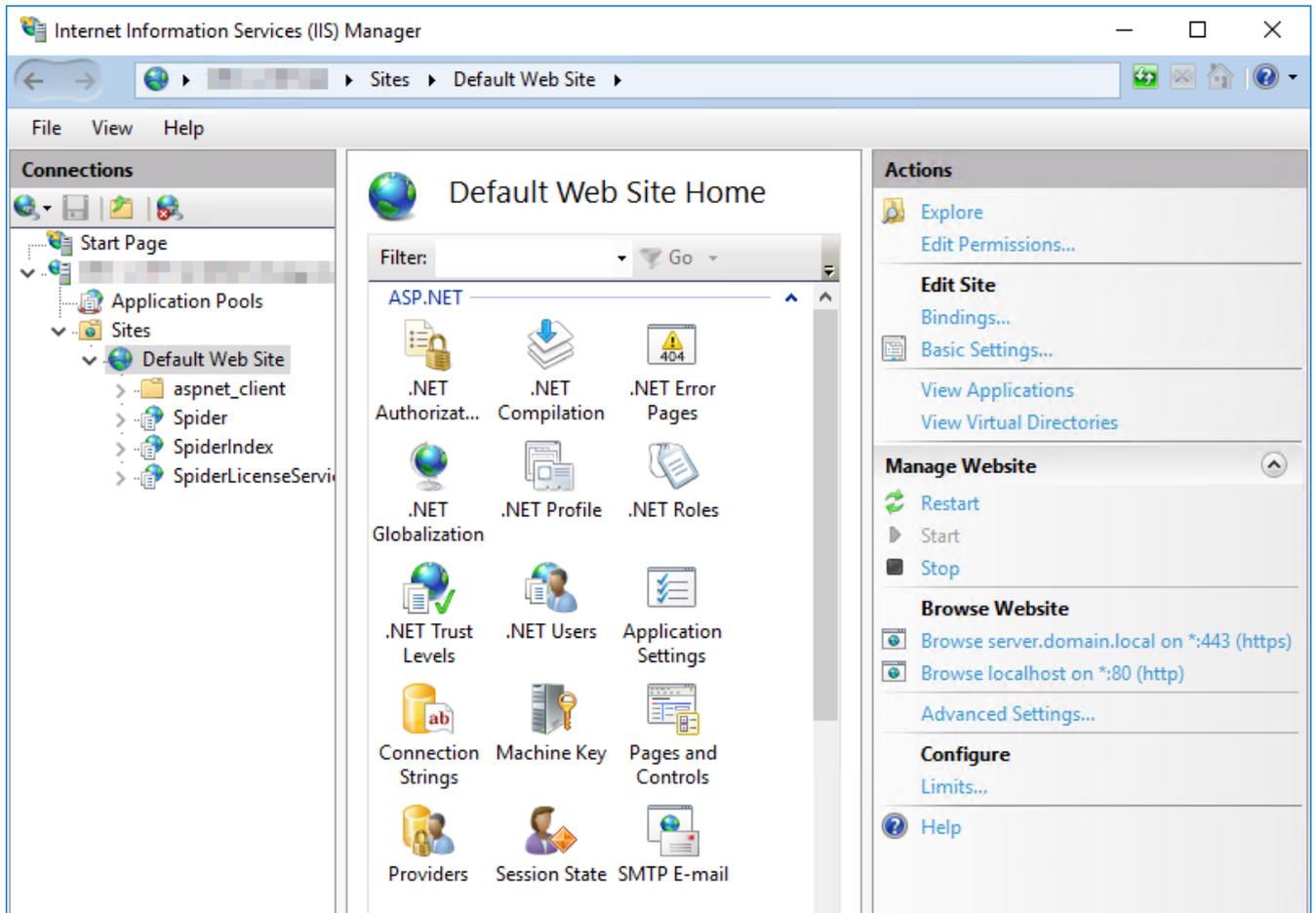


Bei Host name: localhost eintragen, damit ist sichergestellt, dass Applikationen wie der License Service bzw. Index-ProxyApi weiterhin mit der lokalen Maschine kommunizieren können.

➤ **OK**



➤ **Close**



Jetzt ist sichergestellt, dass der Server nach außen hin per HTTPS kommuniziert, während für die lokale Kommunikation HTTP verwendet werden kann.

**Achtung** Die Kommunikation mit dem Lizenzserver per HTTP ist Voraussetzung dafür, dass das Setup das System aktualisieren kann.

## Konfigurationseinträge

### Einträge in der [dbo].[config] Tabelle in der Core Datenbank

Area: Application

Key	Value
Server	https://server.domain.local  server.domain.local muss durch den im Zertifikat angegebenen FQDN ersetzt werden.

## IIS neu starten

Damit sämtliche Änderungen korrekt übernommen werden, ist es nötig den IIS mit dem Befehl **iisreset** neu zu starten.

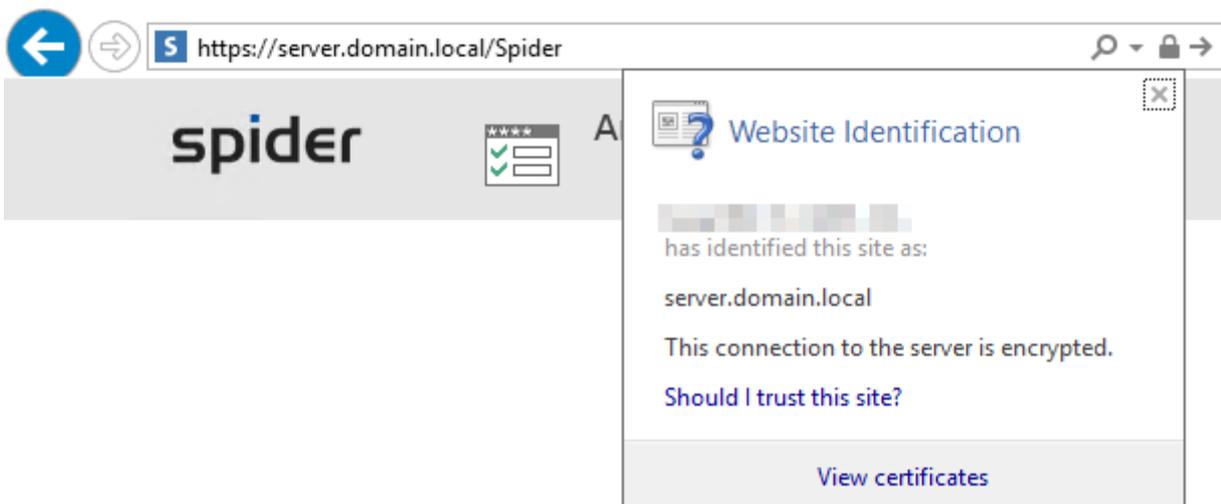
```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\>iisreset

Attempting stop...
Internet services successfully stopped
Attempting start...
Internet services successfully restarted

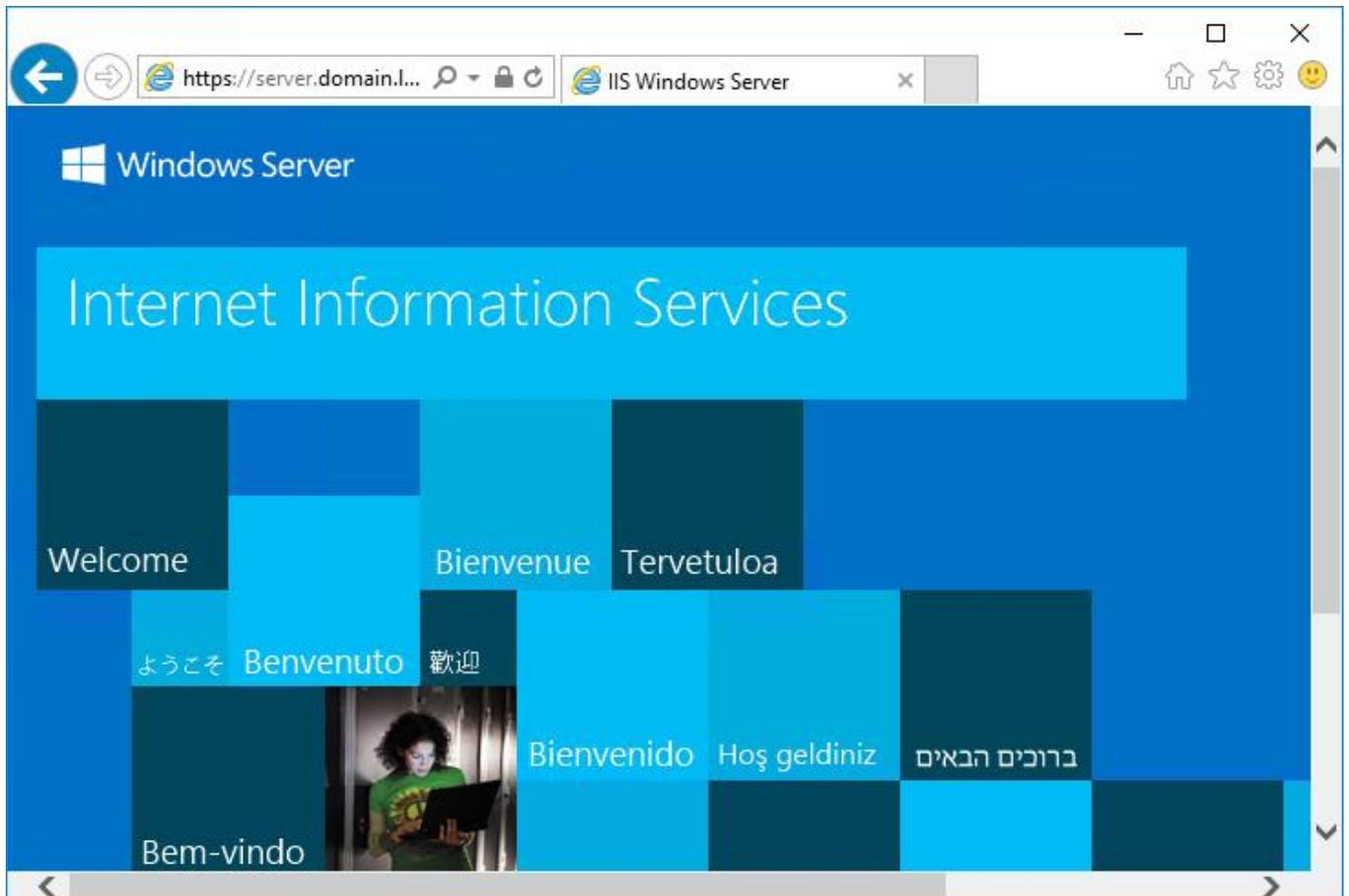
C:\Users\>
```

Jetzt ist die Kommunikation mit der Spider Applikation per SSL möglich.



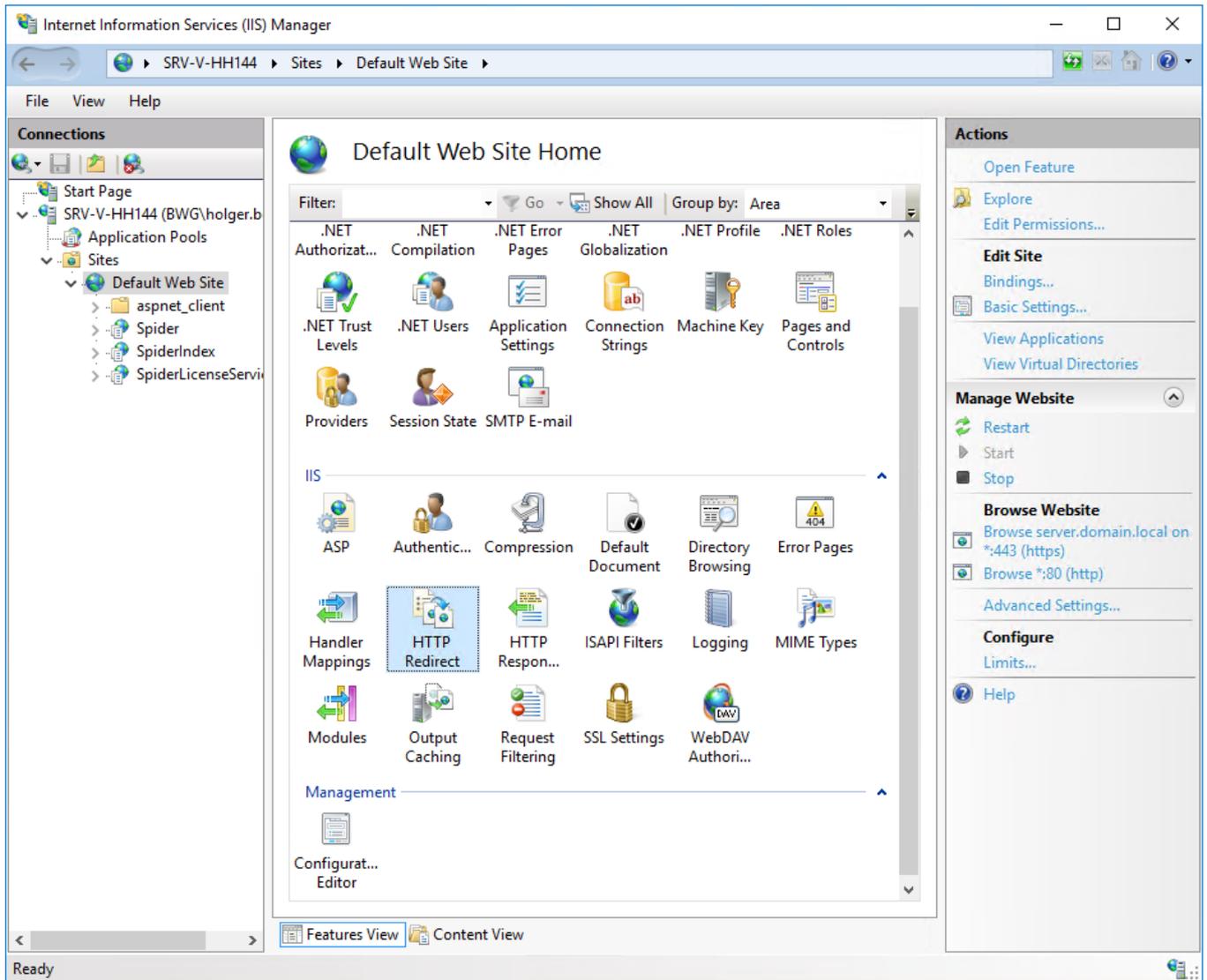
### 3.3.4 HTTP(S) Umleitung

Wird die Adresse des IIS Servers direkt aufgerufen (Beispiel `https://server.domain.local`), wird in der Regel die IIS Standardwebsite angezeigt.

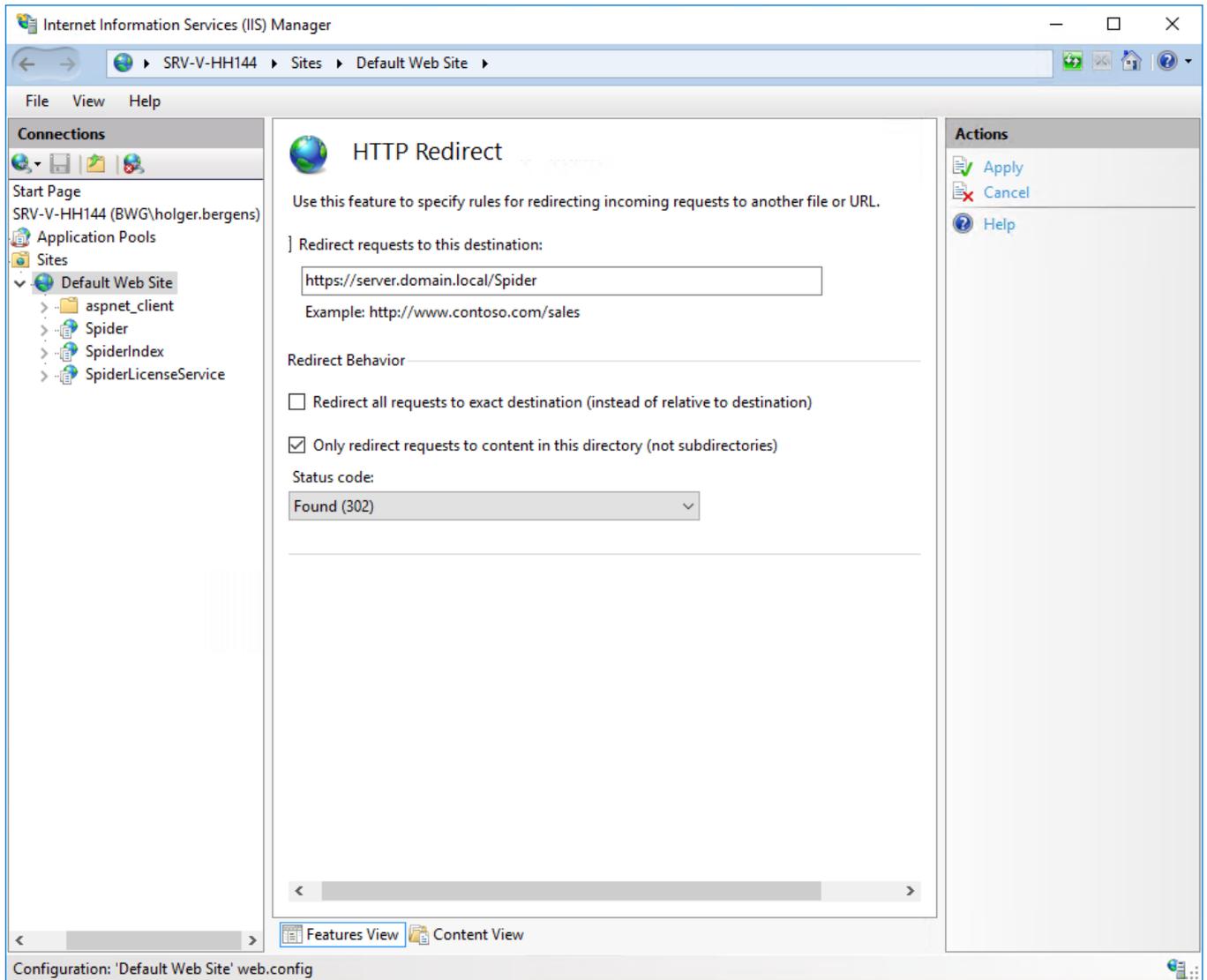


Damit die Adresse des IIS Servers auch die Spider Applikation zeigt, kann eine Umleitung eingerichtet werden.

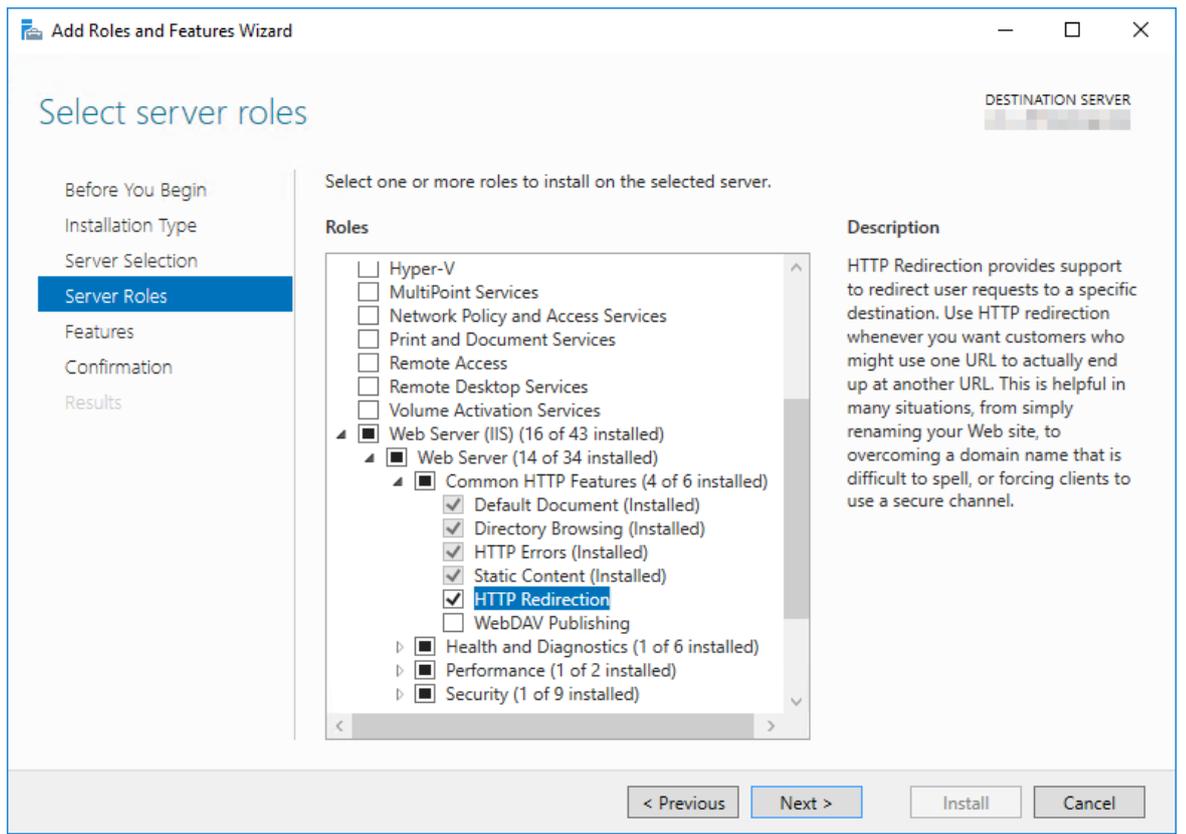
- Starten des Computer **Internet Information Services (IIS) Manager**
- **Default Web Site > IIS > HTTP Redirect**



- Eintragen der Zielwebsite (auf korrektes HTTP/S achten) und **Apply** klicken.



**Hinweis** Sollte der Menüpunkt **HTTP Redirect** fehlen, muss das Rollen Feature ggfs. nachinstalliert werden:



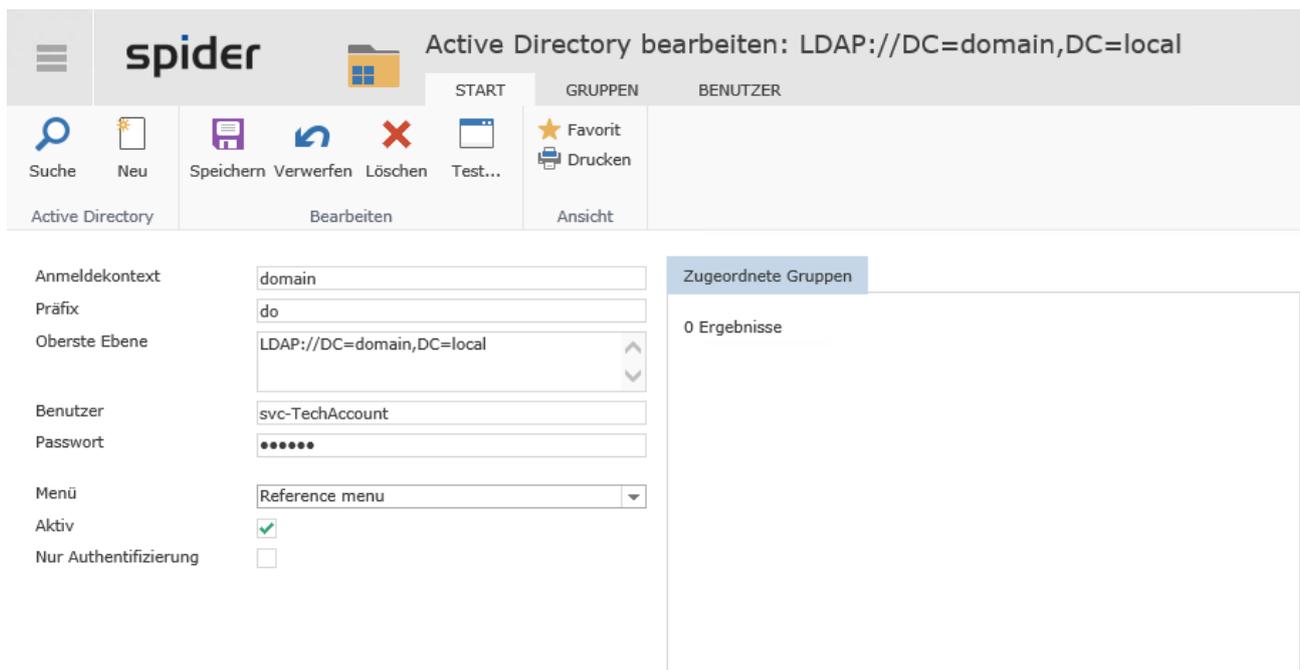
### 3.3.5 Einrichtung Active Directory Login

Um Domänenbenutzern den Zugriff auf das Spider System zu ermöglichen kann das Active Directory Login verwendet werden.

Für die Einrichtung benötigt man Gruppen im Active Directory, die einer oder mehreren Spider-Rollen zugewiesen werden.

#### Einstellungen in Spider

- Aufruf der Spider Website (als Administrator)
- **System > Active Directories > Neu**



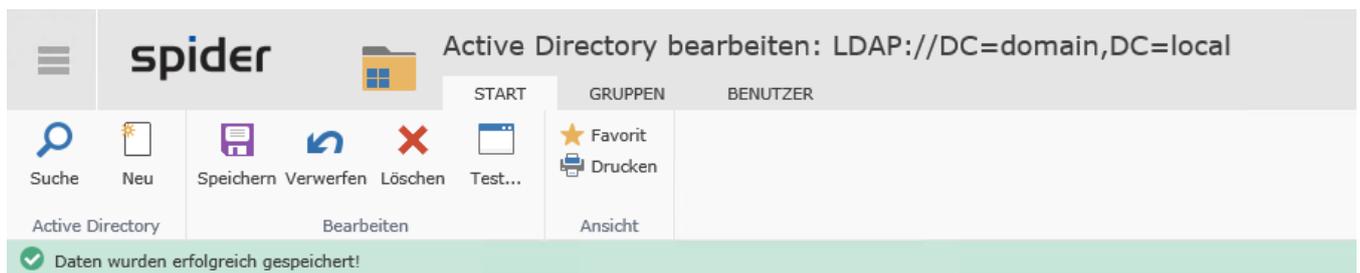
Feld	Beschreibung
Anmeldekontext	NetBios Name der Domain
Präfix	Gewählter Präfix zur Identifizierung
Oberste Ebene	LDAP Eintrag der obersten Ebene der zu verbindenden Domain.
Benutzer	AD Benutzer mit dem die Verbindung zum AD hergestellt wird.
Passwort	Passwort des o.g. AD Benutzers
Menü	Standardmenü für dieses AD
Aktiv	Active Directory Eintrag ist aktiv und kann verwendet werden.
Nur Authentifizierung	Das AD wird nur für die Authentifizierung der Benutzer verwendet, Rechte und Rollen werden weiterhin in Spider verwaltet.

**Hinweis:** Die Verwendung einer signierten/sicheren LDAP-Verbindung ist durch die Angabe des Ports im Feld "Oberste Ebene" möglich. Im Falle einer Standardkonfiguration müsste dem Eintrag demnach ein ":636" angehängt werden.

➤ **Test...**



➤ **Speichern**



➤ **Reiter Gruppen > Gruppen zuweisen**

➤ Domänen-Benutzer eintragen von dem die Gruppen ermittelt werden sollen

➤ **Suchen**



Domänen-Benutzer



Active Directory Gruppen

➤ Gewünschte Gruppen markieren > **Übertragen**

The screenshot shows the Spider web interface for managing Active Directory groups. The top navigation bar includes 'START', 'GRUPPEN', and 'BENUTZER'. The main area is titled 'Active Directory: LDAP://DC=domain,DC=local'. A search bar for 'Domänen-Benutzer' contains 'ADUser'. Below the search bar, a list of Active Directory groups is displayed with checkboxes for selection. Three groups are selected: 'CN=ADGroup-Admin, CN=Users, DC=bwg, DC=corp', 'CN=ADGroup-Reader, CN=Users, DC=bwg, DC=corp', and 'CN=ADGroup-User, CN=Users, DC=bwg, DC=corp'. A green notification bar at the bottom states 'Die ausgewählten Gruppen wurden portiert.' (The selected groups were ported).

spider Active Directory: LDAP://DC=domain,DC=local

START GRUPPEN BENUTZER

Gruppen zuweisen Übertragen

Liste Gruppen

Gruppen verwalten Gruppen

Domänen-Benutzer ADUser Suchen

### Active Directory Gruppen

7 Ergebnisse

<input type="checkbox"/>	ADGroup
<input type="checkbox"/>	CN=Users,CN=Builtin,DC=bwg,DC=corp
<input type="checkbox"/>	CN=Domain Users,CN=Users,DC=bwg,DC=corp
<input checked="" type="checkbox"/>	CN=ADGroup-Admin,CN=Users,DC=bwg,DC=corp
<input checked="" type="checkbox"/>	CN=ADGroup-Reader,CN=Users,DC=bwg,DC=corp
<input checked="" type="checkbox"/>	CN=ADGroup-User,CN=Users,DC=bwg,DC=corp

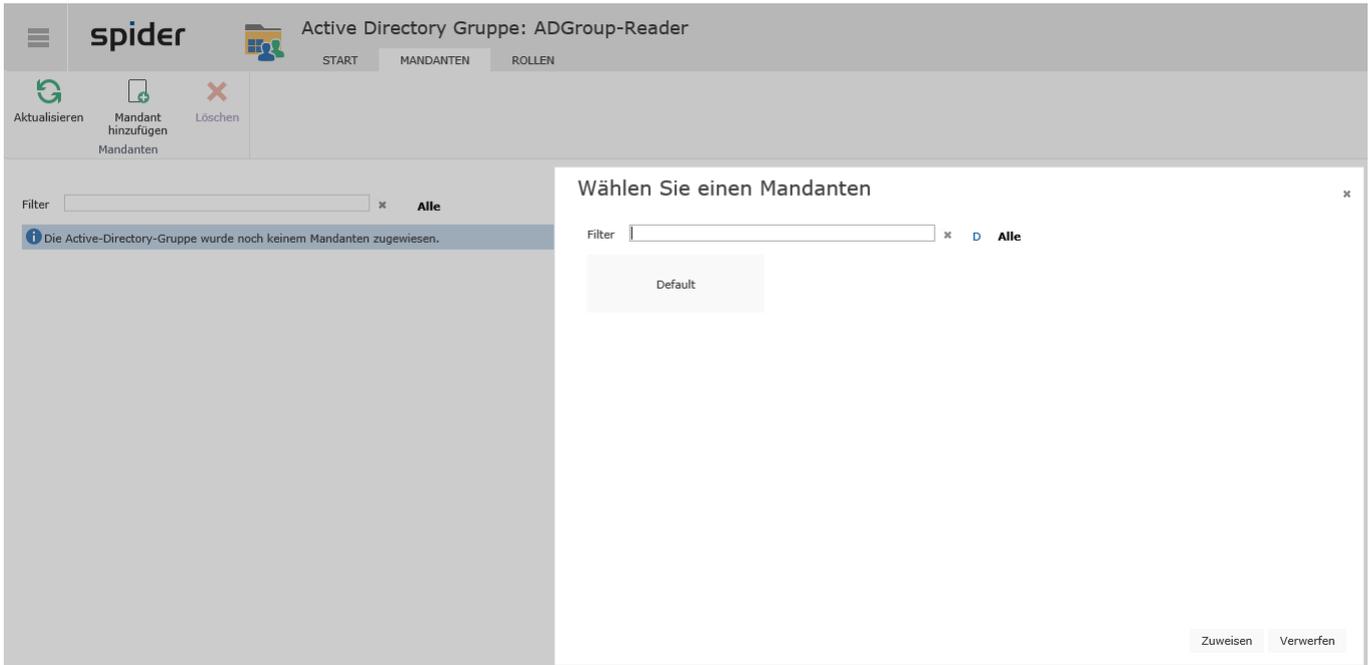
Gruppen verwalten Gruppen

Die ausgewählten Gruppen wurden portiert.

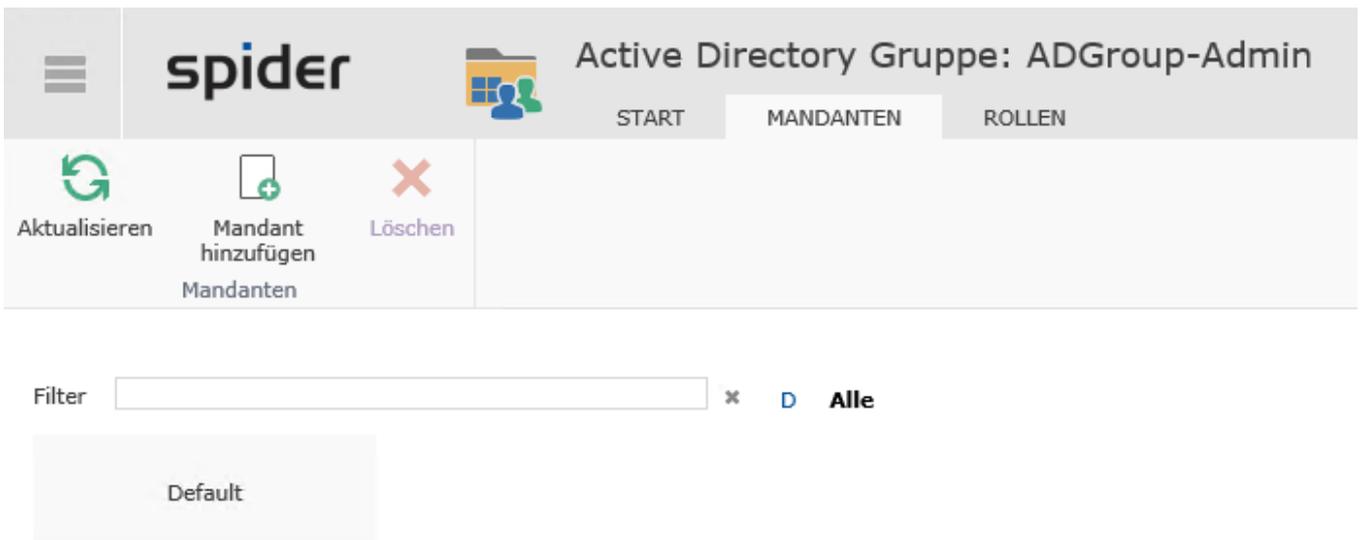
➤ **Start**

➤ **Auswahl einer Gruppe aus Zugeordnete Gruppen**

➤ Reiter **Mandanten**



➤ **Mandant hinzufügen** > Mandant auswählen > **Zuweisen**



➤ Reiter **Rollen**

➤ Gewünschte Rollen markieren > **Rollen hinzufügen**

Nicht zugeordnete Rollen

Rolle

Applikation

Suchen

17 Ergebnisse

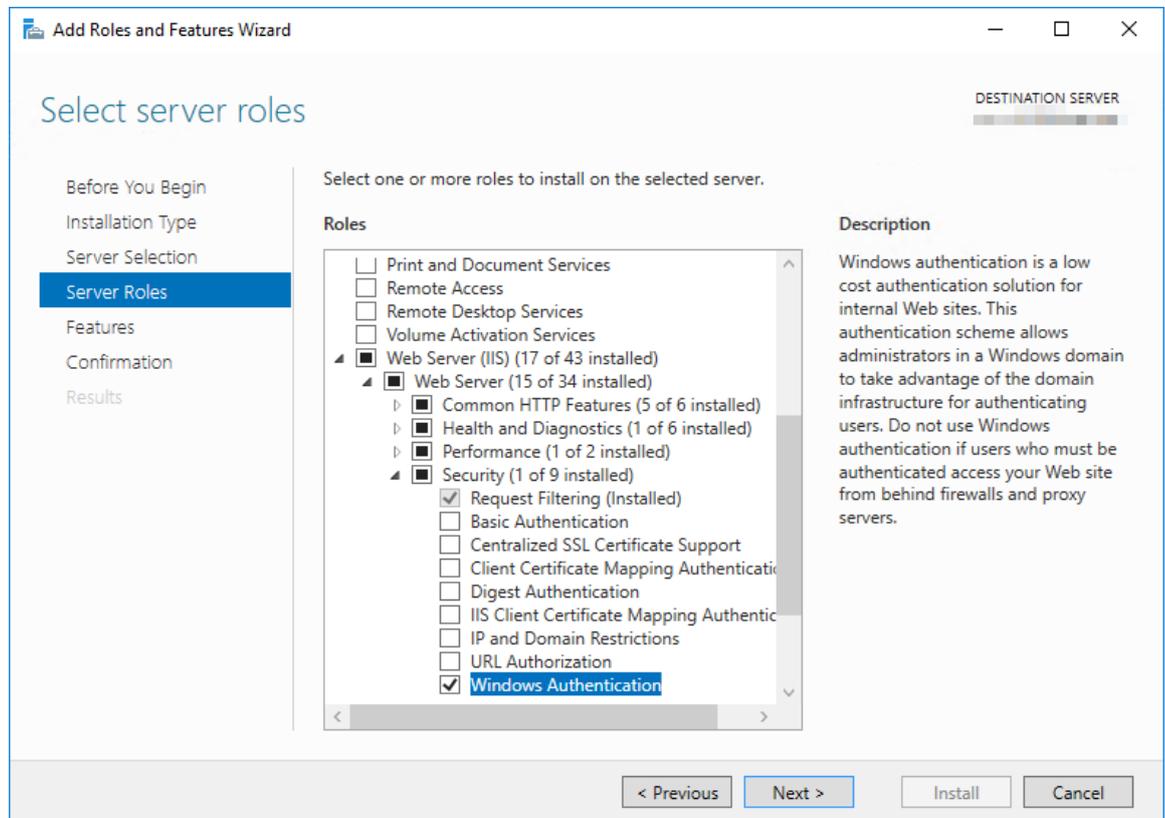
<input type="checkbox"/>	Applikation	Rolle
<input checked="" type="checkbox"/>	Spider Asset	AssetAdmin
<input checked="" type="checkbox"/>	Spider Asset	AssetManager
<input checked="" type="checkbox"/>	Spider Asset	AssetMasterData
<input type="checkbox"/>	Spider Asset	AssetReader
<input checked="" type="checkbox"/>	Spider Contract	ContractAdmin
<input checked="" type="checkbox"/>	Spider Contract	ContractManager
<input checked="" type="checkbox"/>	Spider Contract	ContractReader
<input checked="" type="checkbox"/>	Spider Core	Admin
<input type="checkbox"/>	Spider Core	CoreBasic
<input type="checkbox"/>	Spider Core	CoreDashboard
<input type="checkbox"/>	Spider Core	CoreReader
<input checked="" type="checkbox"/>	Spider Core	MasterDataEditor
<input checked="" type="checkbox"/>	Spider Core	UserManager
<input checked="" type="checkbox"/>	Spider Licence	LicenceAdmin
<input checked="" type="checkbox"/>	Spider Licence	LicenceManager
<input checked="" type="checkbox"/>	Spider Licence	LicenceMasterData
<input checked="" type="checkbox"/>	Spider Licence	LicenceReader

Hiermit ist die Zuweisung von Mandanten und Rollen zu einer Active Directory Gruppe abgeschlossen. Dies muss bei Bedarf für andere Gruppen, Mandanten oder Rollen wiederholt werden.

**Hinweis** In der Area "Application.Security.ActiveDirectory" der Spider Core Config gibt es den boolschen Key "DebugInformation". Ist dieser wahr, wird bei fehlgeschlagenem AD-Login die Loginseite mit zusätzlichen AD-Informationen angezeigt.

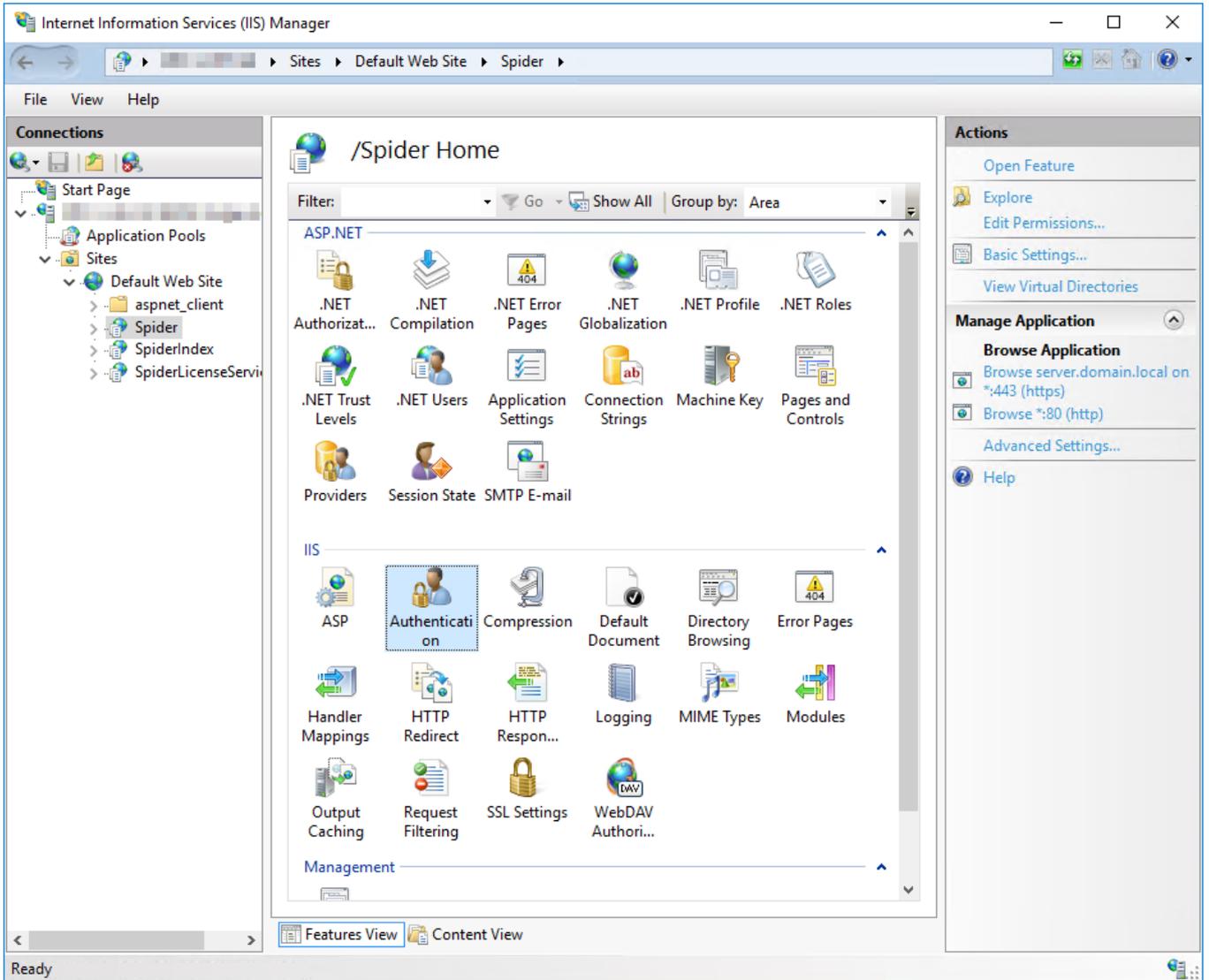
## IIS Einstellungen - Authentication

**Hinweis** Falls die Einstellung Windows Authentication nicht vorhanden ist, muss das Rollen Feature ggfs. nachinstalliert werden:

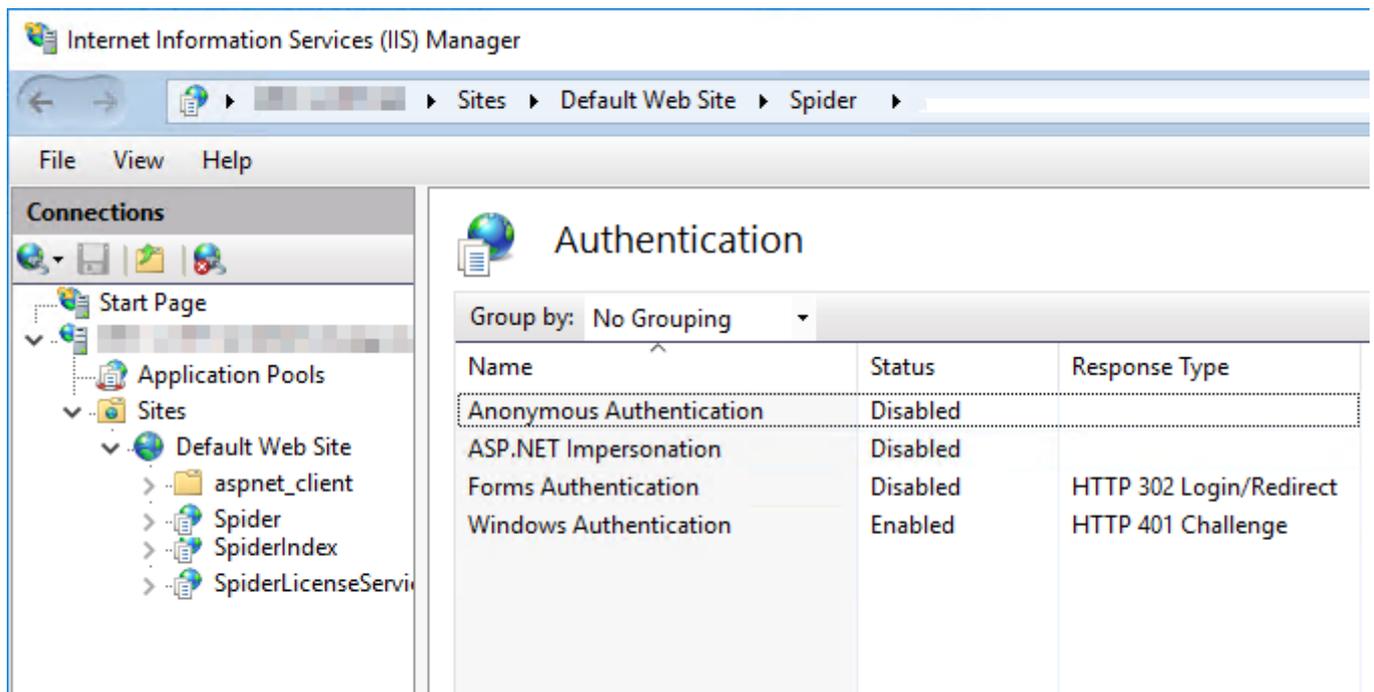


➤ Starten des Computer **Internet Information Services (IIS) Manager**

➤ **Server auswählen > Sites > Default Web Site > Spider Web Site > Authentication**



- Alle Authentication Typen außer **Windows Authentication** auf **Disabled** setzen, **Windows Authentication** auf **Enabled** setzen.



Damit sämtliche Änderungen korrekt übernommen werden, ist es nötig den IIS mit dem Befehl **iisreset** neu zu starten.

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.

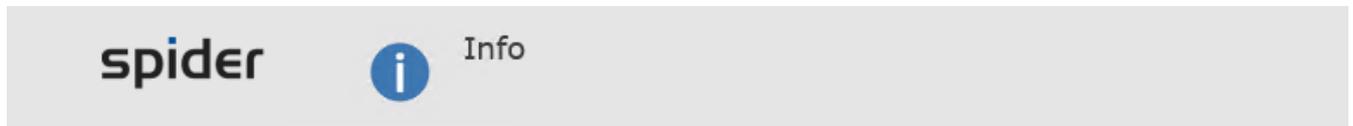
C:\Users\>iisreset

Attempting stop...
Internet services successfully stopped
Attempting start...
Internet services successfully restarted

C:\Users\>
```

### 3.3.6 Info Seite - support.config

Im Auslieferungszustand sind auf der Info Seite, die über einen Klick auf den Info Button im Menu oder im Anmeldebildschirm erreicht werden kann, keine Angaben zum Support hinterlegt.



#### Support

Bevor Sie sich mit einer Supportanfrage an unseren Helpdesk wenden, bitten wir Sie, zuerst die Produktdokumentation und unsere Knowledgebase zu nutzen.

*Es sind keine Supportinformationen im System hinterlegt. Bitte kontaktieren Sie Ihren Administrator.*

#### Copyright

© 2001-2016 Brainware Consulting & Development AG  
Ein Unternehmen der brainwaregroup. Alle Rechte vorbehalten.

<http://www.brainwaregroup.com>

Um dort eigene Informationen zu hinterlegen, muss die Datei **support.config** im Pfad **<InstallDir>\Spider\CoreServer\_XX\Web\\_settings**, editiert werden.

Die Datei enthält eine XML Struktur, in der Sprachabhängig Informationen hinterlegt werden können.

```
<Intro>
  <LanguageCode>DE</LanguageCode>
  <Text>Bevor Sie sich mit einer Supportanfrage an unseren Helpdesk wenden, bitten wir Sie, zuerst die Produktdokumentation und unsere Knowledgebase zu nutzen.</Text>
</Intro>
```

Das Intro ermöglicht es einen allgemeinen Text auf der Info Seite einzublenden. Zusätzlich gibt es die Contact Einträge, mit denen ein oder mehrere Kontaktarten konfiguriert werden können.

```
<Contact>
  <LanguageCode>DE</LanguageCode>
  <Name>Knowledgebase</Name>
  <Icon>SupportKnowledgebase</Icon>
  <Text>Knowledgebase</Text>
  <Url></Url>
</Contact>
```

Feld	Beschreibung
LanguageCode	Das anzugebende Sprachkürzel kann der Spalte "639-1" auf der Seite <a href="https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes">https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes</a> entnommen werden.

Feld	Beschreibung
Name	Eintrag, der den Typ der anzuzeigenden Ressource definiert. Page (eine Webseite) Form (Ein Formular auf einer Webseite) Email (eine Email Adresse) Phone (eine Telefonnummer)
Icon	Das anzuzeigende Icon, siehe nachstehende Tabelle.
Text	Der anzuzeigende Text
URL	Falls bei Name eine Eintrag des Types Page, Form oder Email gemacht wird, kann hier die Ziel-URL eingetragen werden. Achtung, bei einer Emailadresse muss der Adresse ein <b>mailto:</b> vorangestellt werden!

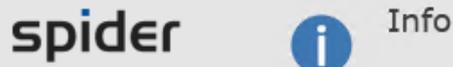
Icon-Code	Icon
SupportKnowledgebase	
SupportPage	
SupportForm	
SupportEmail	
SupportPhone	

**Wichtig** Pro LanguageCode kann nur ein "Name" verwendet werden! Die Kombination aus LanguageCode und Name muss also eindeutig sein.

```
<Contact>
  <LanguageCode>DE</LanguageCode>
  <Name>Page</Name>
  <Icon>SupportPage</Icon>
  <Text>Supportseite</Text>
  <Url>https://support.brainwaregroup.com/</Url>
</Contact>
<Contact>
  <LanguageCode>DE</LanguageCode>
  <Name>Form</Name>
  <Icon>SupportForm</Icon>
  <Text>Support Formular</Text>
  <Url>http://support.brainwaregroup.com/form/</Url>
</Contact>
<Contact>
  <LanguageCode>DE</LanguageCode>
  <Name>Email</Name>
  <Icon>SupportEmail</Icon>
  <Text>E-Mail</Text>
  <Url>mailto:support@brainwaregroup.com</Url>
</Contact>
<Contact>
  <LanguageCode>DE</LanguageCode>
  <Name>Phone</Name>
```

```
<Icon>SupportPhone</Icon>
<Text>Telefon: +41 41 748 22 11</Text>
<Url />
</Contact>
```

resultiert in:



## Support

Bevor Sie sich mit einer Supportanfrage an unseren Helpdesk wenden, bitten wir Sie, zuerst die Produktdokumentation und unsere Knowledgebase zu nutzen.

-  [Supportseite](#)
-  [Support Formular](#)
-  [E-Mail](#)
-  [Telefon: +41 41 748 22 11](#)

## Copyright

© 2001-2016 Brainware Consulting & Development AG  
 Ein Unternehmen der brainwaregroup. Alle Rechte vorbehalten.  
<http://www.brainwaregroup.com>

### 3.3.7 Verfügbare Sprachen - culture.config

Die im Web zur Verfügung stehenden Sprachen können über die Datei culture.config (Pfad **<InstallDir>\Spider\CoreServer\_XX\Web\_settings**) konfiguriert werden.

```
<?xml version="1.0" ?>
<culture>
  <availableCultures>
    <clear />
    <add cultureCode="en-GB" />
    <add cultureCode="en-US" />
    <add cultureCode="en-ZA" />
    <add cultureCode="de-DE" />
    <add cultureCode="de-CH" />
    <add cultureCode="de-AT" />
    <add cultureCode="ja-JP" />
  </availableCultures>
  <defaultCulture cultureCode="de-DE" />
</culture>
```

Eine Liste der verfügbaren Culture Codes können Sie der Tabelle auf <https://msdn.microsoft.com/library/ee825488.aspx>, Spalte Language Culture Name entnehmen.



Durch Hinzufügen oder Entfernen von Einträgen kann die Liste angepasst werden.

Der Eintrag `<defaultCulture cultureCode="" />` bestimmt, welche Sprache angezeigt wird, wenn ein Benutzer die Seite aufruft.

**Wichtig** Das Hinzufügen einer neuen Sprache in der `culture.config` bedeutet nicht, dass für diese Sprache auch Übersetzungen vorhanden sind. Sind keine Übersetzungen für den Culture Code vorhanden, werden automatisch Begriffe der Default Culture angezeigt.

### 3.3.8 Einstellungen in der Config Tabelle

Mit den Einstellungen in der Config Tabelle kann das Verhalten von Spider angepasst werden. Nicht alle Einträge sind über den Spider Admin oder das WebFrontend editierbar.

Area	Key	Value	Beschreibung
Authentication	BrowserEnableAutocomplete	True   False	<p>Mit diesem Eintrag kann bei sensiblen Feldern wie dem Passwort Feld auf der Login Seite, das Standardverhalten für das automatische ausfüllen gesetzt werden.</p> <p><b>Achtung:</b> Nicht jeder Browser unterstützt, bzw. reagiert auf diese Einstellung, die Sicherstellung der Funktionalität dieser Einstellung kann nur mit einem Browser gewährleistet werden der diese Einstellung auswertet.</p>

### 3.3.9 Anzeige erweiterter Fehlermeldungen

Die in Spider implementierte Ausnahmebehandlung sieht die Anzeige von Fehlermeldungen auf einer spezifischen Fehlerseite vor. Aufgrund sicherheitsspezifischer Aspekte zeigt die Seite nicht alle Informationen einer Ausnahme an. Um detaillierte Fehlermeldungen des IIS anzuzeigen, ist Spider mit den folgenden Einstellungen zu konfigurieren.

Dazu ist es nötig einige Dateien mit einem Editor zu verändern:

### customErrors.config (..\Spider\CoreServer\_XX\Web\_Settings)

```
<?xml version="1.0" encoding="utf-8"?>
<!--
  For DEBUG switch customErrors mode to "Off" (with upper-case "O") and
  httpErrors errorMode to "Detailed" and then restart the ApplicationPool
  in IIS. Without a restart the changes will not be applied!

  (An ApplicationPool restart can also be achieved by saving the
  Web.config file without any changes. The IIS recognizes the file has
  been saved and then restarts the application pool.)

  If you encounter an HTTP 500 error message, ensure that httpErrors
  existingResponse is set to "Auto"!
-->

<customErrors
  mode="On"
  defaultRedirect="~/Info/Error.aspx">

  <!-- NEVER redirect a 401 error! This will cause various problems in the application! -->

  <error statusCode="403" redirect="~/Default.aspx" />
  <error statusCode="404" redirect="~/Default.aspx" />
  <error statusCode="405" redirect="~/Info/Error.aspx?Category=8" />
  <error statusCode="406" redirect="~/Info/Error.aspx?Category=8" />
  <error statusCode="412" redirect="~/Info/Error.aspx?Category=8" />
  <error statusCode="500" redirect="~/Info/Error.htm" />
  <error statusCode="501" redirect="~/Info/Error.htm" />
  <error statusCode="502" redirect="~/Info/Error.htm" />

</customErrors>
```

```
<?xml version="1.0" encoding="utf-8"?>
<!--
  For DEBUG switch customErrors mode to "Off" (with upper-case "O") and
  httpErrors errorMode to "Detailed" and then restart the ApplicationPool
  in IIS. Without a restart the changes will not be applied!

  (An ApplicationPool restart can also be achieved by saving the
  Web.config file without any changes. The IIS recognizes the file has
  been saved and then restarts the application pool.)

  If you encounter an HTTP 500 error message, ensure that httpErrors
  existingResponse is set to "Auto"!
-->

<customErrors
  mode="Off"
  defaultRedirect="~/Info/Error.aspx">
```

```

<!-- NEVER redirect a 401 error! This will cause various problems in the application! -->

<error statusCode="403" redirect="~/Default.aspx" />
<error statusCode="404" redirect="~/Default.aspx" />
<error statusCode="405" redirect="~/Info/Error.aspx?Category=8" />
<error statusCode="406" redirect="~/Info/Error.aspx?Category=8" />
<error statusCode="412" redirect="~/Info/Error.aspx?Category=8" />
<error statusCode="500" redirect="~/Info/Error.htm" />
<error statusCode="501" redirect="~/Info/Error.htm" />
<error statusCode="502" redirect="~/Info/Error.htm" />

</customErrors>.

```

### httpErrors.config (..\Spider\CoreServer\_XX\Web\\_Settings)

```

<?xml version="1.0" ?>
<!--
  For DEBUG switch customErrors mode to "Off" (with upper-case "O") and
  httpErrors errorMode to "Detailed" and then restart the ApplicationPool
  in IIS. Without a restart the changes will not be applied!

  (An ApplicationPool restart can also be achieved by saving the
  Web.config file without any changes. The IIS recognizes the file has
  been saved and then restarts the application pool.)

  If you encounter an HTTP 500 error message, ensure that httpErrors
  existingResponse is set to "Auto"!
-->
-->
<httpErrors errorMode="Custom" defaultResponseMode="Redirect" existingResponse="Auto">
  <!-- NEVER redirect a 401 error! This will cause various problems in the application! -->
  <remove statusCode="403" subStatusCode="-1" />
  <remove statusCode="404" subStatusCode="-1" />
  <remove statusCode="405" subStatusCode="-1" />
  <remove statusCode="406" subStatusCode="-1" />
  <remove statusCode="412" subStatusCode="-1" />
  <remove statusCode="500" subStatusCode="-1" />
  <remove statusCode="501" subStatusCode="-1" />
  <remove statusCode="502" subStatusCode="-1" />
  <!-- Replace "/Spider/" in path attribute with the path of the application. "~" is not allowed here! -->
  <error statusCode="403" prefixLanguageFilePath="" path="/Spider/Default.aspx" responseMode="Redirect" />
  <error statusCode="404" prefixLanguageFilePath="" path="/Spider/Default.aspx" responseMode="Redirect" />
  <error statusCode="405" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="406" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="412" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="500" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
  <error statusCode="501" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
  <error statusCode="502" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
</httpErrors>

```

```

<?xml version="1.0" ?>
<!--
  For DEBUG switch customErrors mode to "Off" (with upper-case "O") and
  httpErrors errorMode to "Detailed" and then restart the ApplicationPool
  in IIS. Without a restart the changes will not be applied!

  (An ApplicationPool restart can also be achieved by saving the
  Web.config file without any changes. The IIS recognizes the file has
  been saved and then restarts the application pool.)

  If you encounter an HTTP 500 error message, ensure that httpErrors
  existingResponse is set to "Auto"!
-->
<httpErrors errorMode="'Detailed' defaultResponseMode="Redirect" existingResponse="Auto">
  <!-- NEVER redirect a 401 error! This will cause various problems in the application! -->
  <remove statusCode="403" subStatusCode="-1" />
  <remove statusCode="404" subStatusCode="-1" />
  <remove statusCode="405" subStatusCode="-1" />
  <remove statusCode="406" subStatusCode="-1" />
  <remove statusCode="412" subStatusCode="-1" />
  <remove statusCode="500" subStatusCode="-1" />
  <remove statusCode="501" subStatusCode="-1" />
  <remove statusCode="502" subStatusCode="-1" />
  <!-- Replace "/Spider/" in path attribute with the path of the application. "~" is not allowed here! -->
  <error statusCode="403" prefixLanguageFilePath="" path="/Spider/Default.aspx" responseMode="Redirect" />
  <error statusCode="404" prefixLanguageFilePath="" path="/Spider/Default.aspx" responseMode="Redirect" />
  <error statusCode="405" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="406" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="412" prefixLanguageFilePath="" path="/Spider/Info/Error.aspx?Category=8" responseMode="ExecuteURL" />
  <error statusCode="500" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
  <error statusCode="501" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
  <error statusCode="502" prefixLanguageFilePath="" path="/Spider/Info/Error.htm" responseMode="Redirect" />
</httpErrors>

```

**Achtung** Die Einstellungen in den Dateien müssen unter Beachtung von Groß- und Kleinschreibung erfolgen!

### 3.3.10 Verschlüsselung der Konfiguration

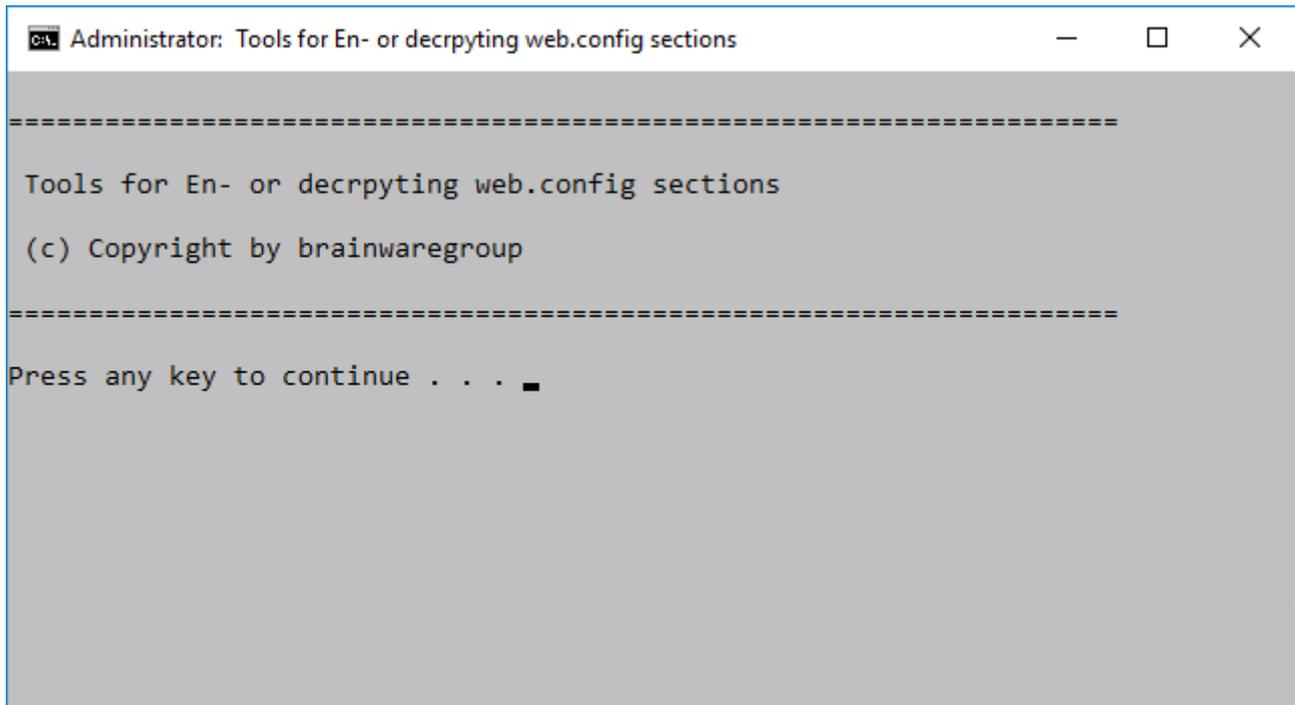
Falls in der Konfiguration die Klartextinformationen des SQL Zugriffs abgelegt werden, können die Connections Strings verschlüsselt werden.

Im Utilities Ordner (..\Spider\CoreServer\_00\Utilities\Spider web.config Crypter.cmd) der Spider Instanz liegt eine Batchdatei, mit der die Ver- und Entschlüsselung durchgeführt werden kann.

#### Verschlüsselung

- Ausführen der **Spider web.config Crypter.cmd**

## ➤ Taste zum Fortfahren drücken



```
Administrator: Tools for En- or decrpyting web.config sections

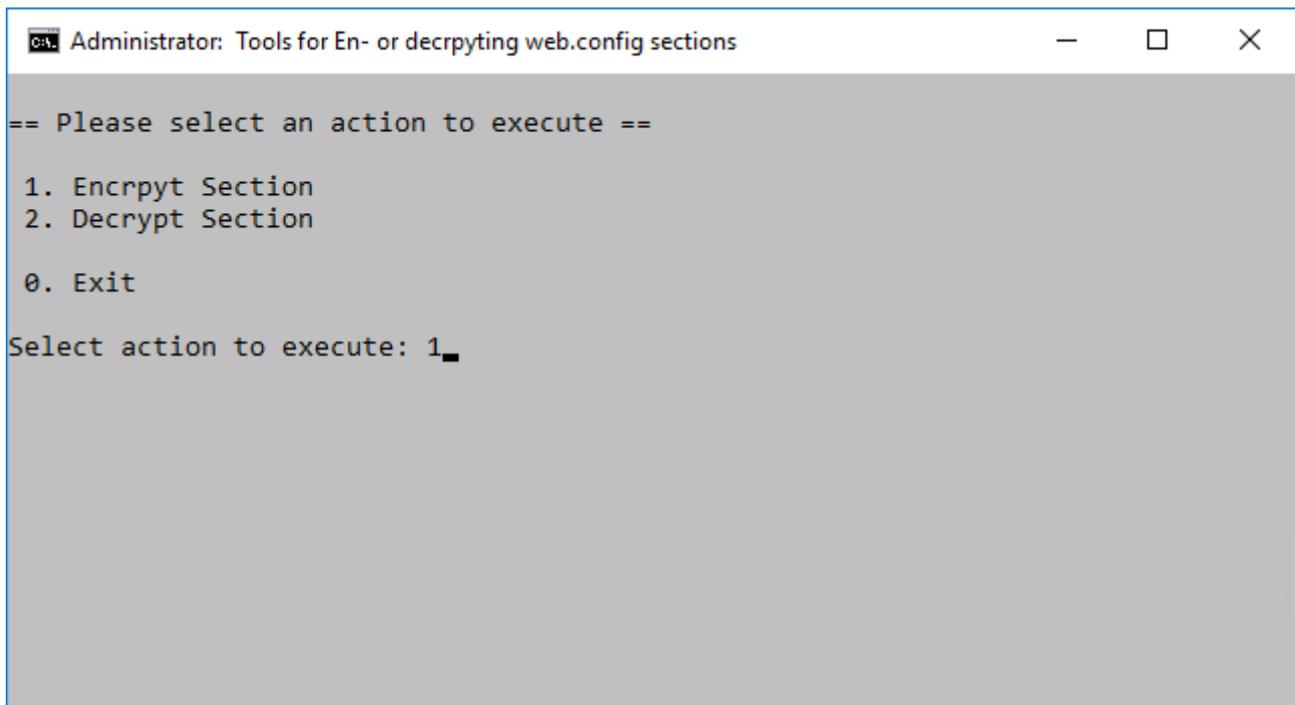
-----

Tools for En- or decrpyting web.config sections
(c) Copyright by brainwaregroup

-----

Press any key to continue . . . █
```

## ➤ Auswahl ob Ent- oder Verschlüsselt werden soll



```
Administrator: Tools for En- or decrpyting web.config sections

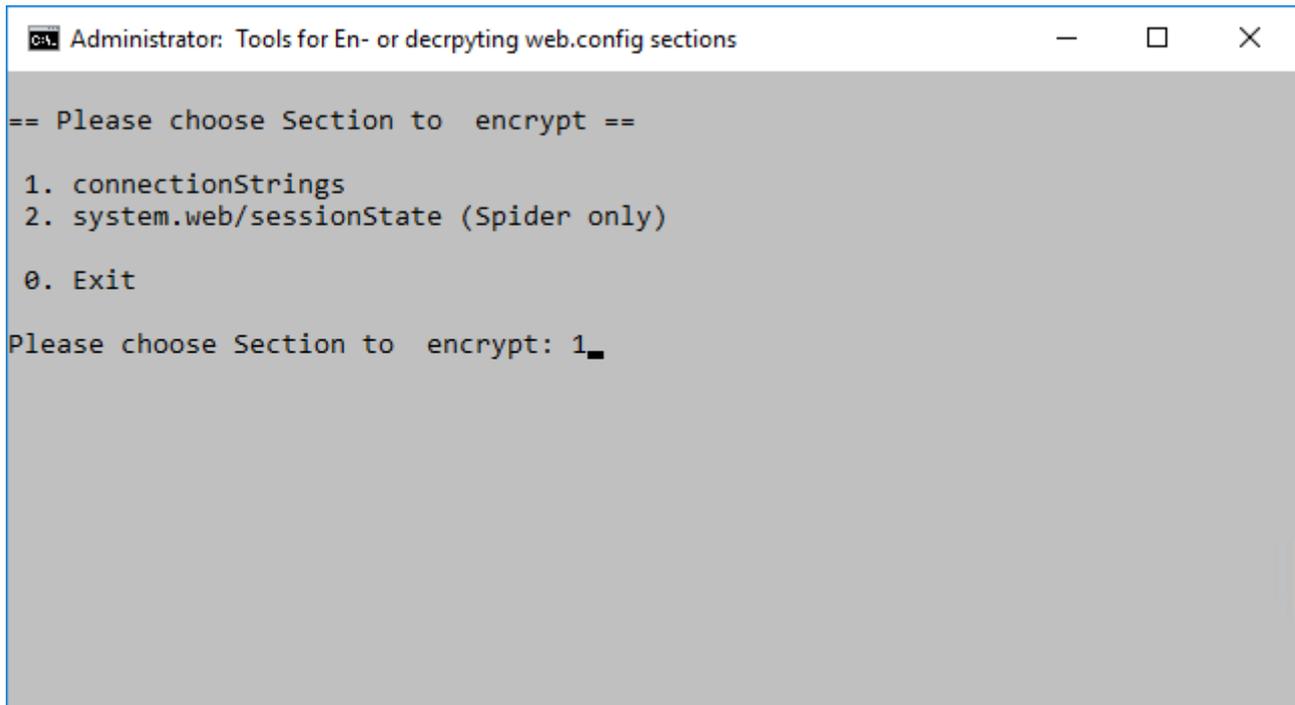
== Please select an action to execute ==

1. Encrpyt Section
2. Decrypt Section

0. Exit

Select action to execute: 1_█
```

## ➤ Auswahl der zu verschlüsselnden Sektion



```
Administrator: Tools for En- or decrypting web.config sections

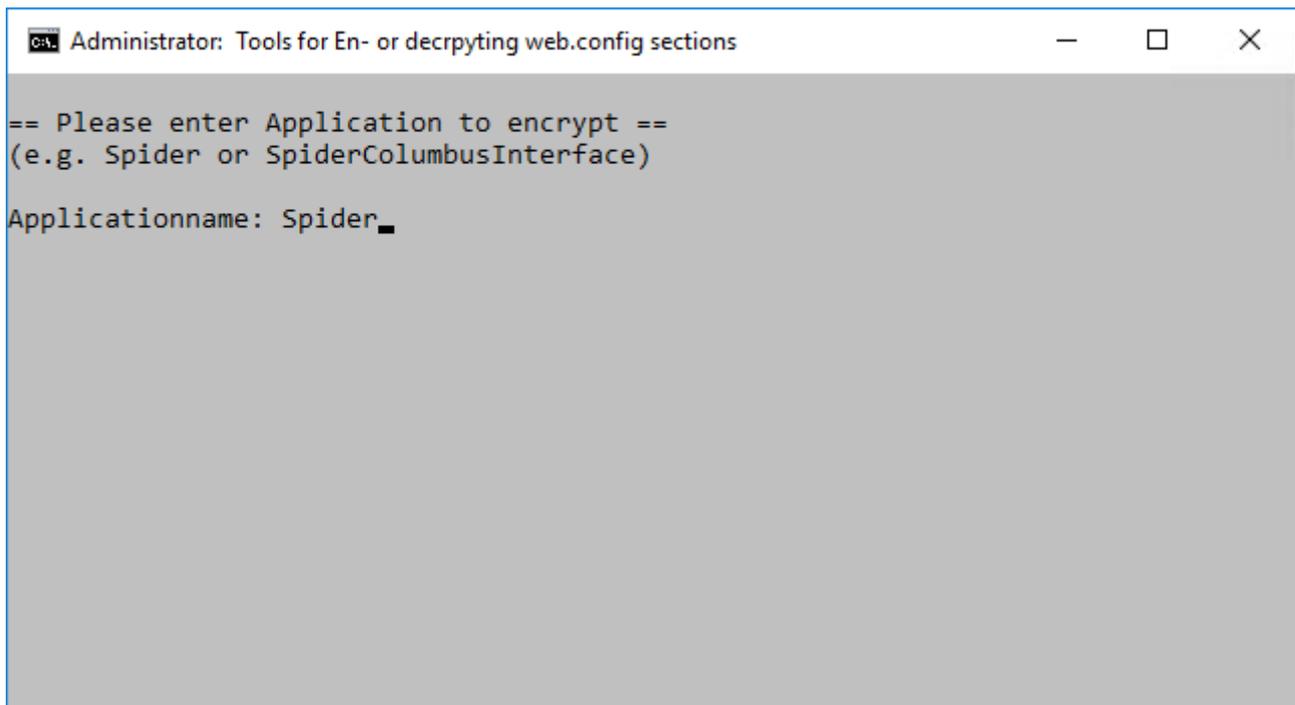
== Please choose Section to  encrypt ==

1. connectionStrings
2. system.web/sessionState (Spider only)

0. Exit

Please choose Section to  encrypt: 1_
```

## ➤ Name der IIS Applikation eingeben



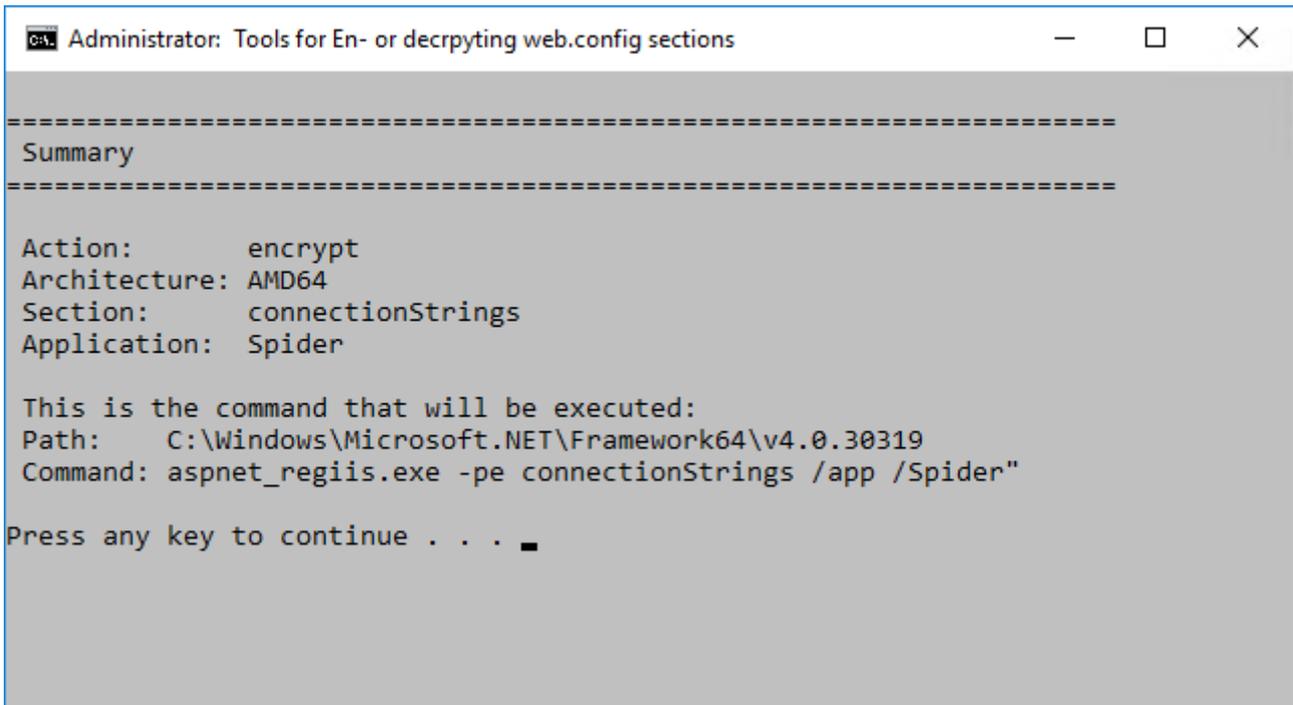
```
Administrator: Tools for En- or decrypting web.config sections

== Please enter Application to encrypt ==
(e.g. Spider or SpiderColumbusInterface)

Applicationname: Spider_
```

Es können die connectionStrings der Spider-Web Applikation und/oder der SpiderIndexProxyApi verarbeitet werden.

- Zusammenfassung mit Betätigen einer Taste bestätigen



```
Administrator: Tools for En- or decrypting web.config sections

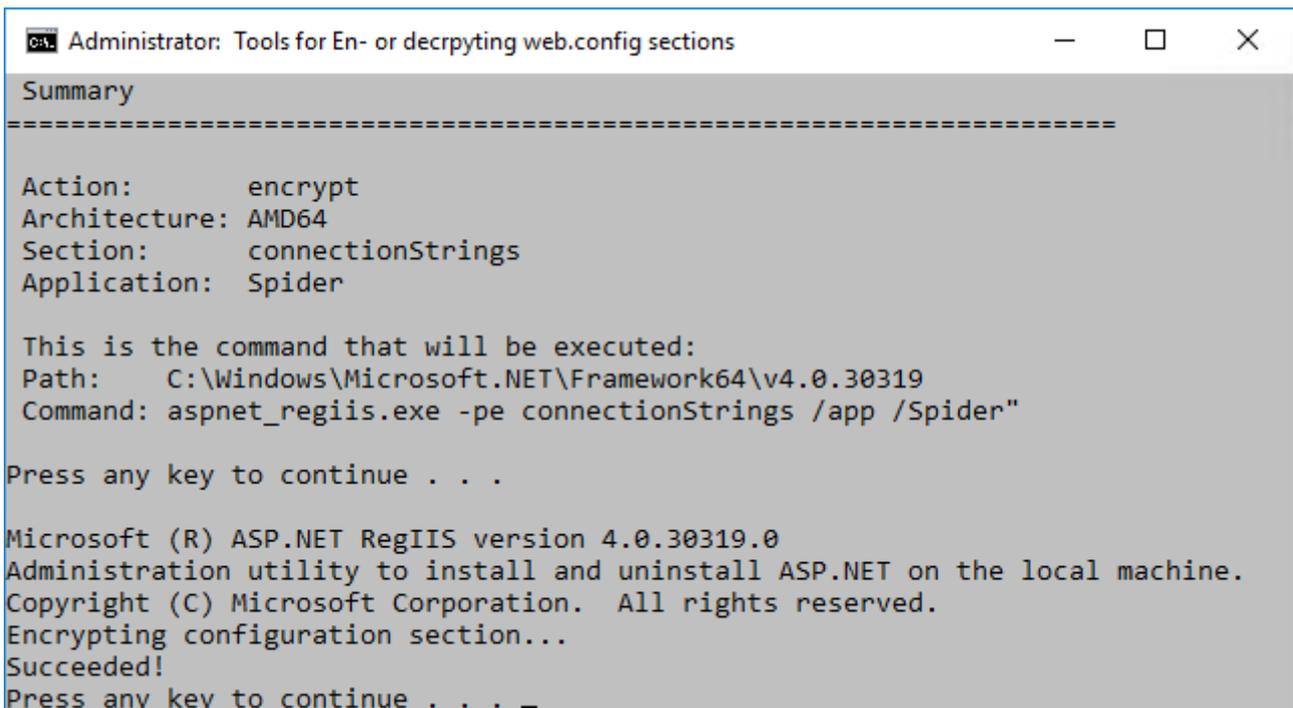
=====
Summary
=====

Action:      encrypt
Architecture: AMD64
Section:     connectionStrings
Application: Spider

This is the command that will be executed:
Path:       C:\Windows\Microsoft.NET\Framework64\v4.0.30319
Command:    aspnet_regiis.exe -pe connectionStrings /app /Spider"

Press any key to continue . . . █
```

- Taste zum Fortfahren drücken



```
Administrator: Tools for En- or decrypting web.config sections

Summary
=====

Action:      encrypt
Architecture: AMD64
Section:     connectionStrings
Application: Spider

This is the command that will be executed:
Path:       C:\Windows\Microsoft.NET\Framework64\v4.0.30319
Command:    aspnet_regiis.exe -pe connectionStrings /app /Spider"

Press any key to continue . . . █

Microsoft (R) ASP.NET RegIIS version 4.0.30319.0
Administration utility to install and uninstall ASP.NET on the local machine.
Copyright (C) Microsoft Corporation. All rights reserved.
Encrypting configuration section...
Succeeded!
Press any key to continue . . . █
```

Auf dem gleichen Weg lassen sich die entsprechenden Dateien auch wieder entschlüsseln.

### Beispiel zur Verschlüsselung ohne die Batchdatei

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis.exe -pe connectionStrings /app /<IIS Applika-
tionsname>
```

### Beispiel zur Entschlüsselung ohne die Batchdatei

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis.exe -pd connectionStrings /app /<IIS Applika-
tionsname>
```

### Beispiel zur Verschlüsselung der SessionState Konfiguration ohne die Batchdatei

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis.exe -pe system.web/sessionState /app /<IIS Ap-
plikationsname>
```

### Beispiel zur Entschlüsselung der SessionState Konfiguration ohne die Batchdatei

```
%WINDIR%\Microsoft.NET\Framework64\v4.0.30319\aspnet_regiis.exe -pd system.web/sessionState /app /<IIS Ap-
plikationsname>
```

### Beispiel unverschlüsselte Datei

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <add name="CoreConnectionString" connectionString="server=.;
    database=SpiderCore;
    Integrated Security=True;
    application name=Spider;
    persist security info=True;
    packet size=4096"
    providerName="System.Data.SqlClient" />
</connectionStrings>
```

### Beispiel verschlüsselte Datei

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings configProtectionProvider="RsaProtectedConfigurationProvider">
  <EncryptedData Type="http://www.w3.org/2001/04/xmldsig#Element"
    xmlns="http://www.w3.org/2001/04/xmldsig#">
    <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmldsig#tripleDES-cbc" />
    <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
      <EncryptedKey xmlns="http://www.w3.org/2001/04/xmldsig#">
        <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmldsig#rsa-1_5" />
        <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
          <KeyName>Rsa Key</KeyName>
        </KeyInfo>
      </EncryptedKey>
      <CipherData>
        <CipherValue>BwPqFNjS0A9dhKo0UxNK+me2pTQ/p4GixYvuGNJ6DRd3zTFJJQxJ14uZrW585u1Xt1Ix1oEs8MMUDCmoyuCs0hcw0+7pKTt0m4EODghL+Hfo72x
FGikQfkXFKh6enNaRWP32BsxuobAQ8gB8+uN10eJJTB18gaJDjQ306K4Q72gm1Kmm0F600K+r/ceyOyvabjvnX2rceFF4R0W0StXa3jCF3mcP5TJThIY3BmWII9
Pyoab+42o0P+R1K5R2j0Ln6Qj1Hhm2tHb+FCGF/b4cUrLUApKf9S3iMuwrvBpAw0pGJfgMqr1st5yihbsMb+FMAVQQmg7tGTqt9WICtKfQ==</CipherValue>
      </CipherData>
    </EncryptedData>
  </KeyInfo>
```

```
<CipherData>
<CipherValue>90D12S80i6ouIZ/1KZRjxIx3HvRzEqWzS1TfvXpDZXxDBuNxRMFzEG92RLme1+f6fvim9pZqzW2e126FIHNhx/XYQQ5UhI7x0khVLL+cXvxfQX5
1Hr1cLHJ2GwNFUsPucWaAX2LXDxwA9XcAjQeOZqR2sw8TyfADsDzLJYib+Iv7F30H4hXv5D5+TuwCkOq60T+oeFsLiVojJFyyB1hy0GwgPzkwahD06fkPN14UQrG
ik3bV3k+iSx3kBa1DL8130gtgJPLoEn9ZdpXwc9kRXdvXompUyNBOW6p617jRYaE6xdC5y7I39hqyy035EhBnkjT9p1TTonRPKe39Z4WJPFcOGMAy23IhZ+YTW+J
vgcLd7zB4ZfnU+GTBHC2sfMbdy1E155181qKBcHaE7m5BDGXTnnUH1A26R3UtKhWgmq1H9MnR1KJDEQ==</CipherValue>
</CipherData>
</EncryptedData>
</connectionStrings>
```

### 3.3.11 Fehlerbehandlungssystem

---

Generell gilt: Spider hat ein einheitliches Fehlerbehandlungssystem. Jedoch ist es mit der zugrundeliegenden Technologie (ASP.NET) nur möglich bestimmte Fehlerklassen abzufangen. Einige Ausnahmen behandelt der Webserver (IIS) selbst und es ist im Rahmen unserer Applikation nicht möglich die Darstellung dieser Server Responses zu beeinflussen. Microsoft schreibt hierzu: „You cannot customize the following HTTP error messages: 400, 403.9, 411, 414, 500, 500.11, 500.14, 500.15, 501, 503, and 505.“

Details: [Configuring HTTP Error Responses in IIS 7](#)

## 3.4 Aktualisierung

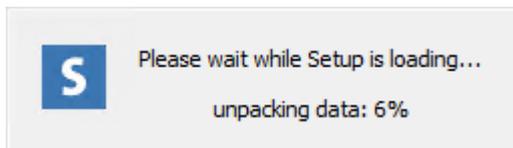
---

Dieses Kapitel beschreibt die Aktualisierung einer Spider Installation.

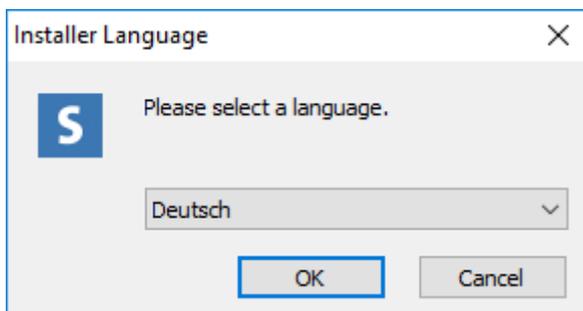
### 3.4.1 Start des Updates

---

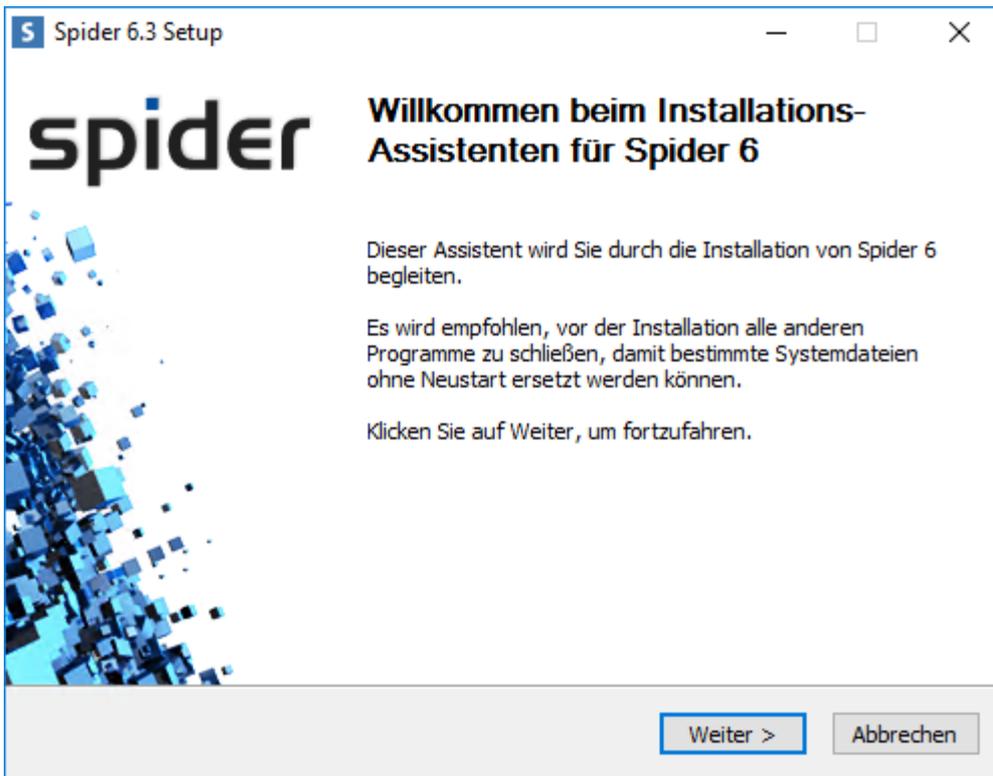
Starten Sie die Installation, indem Sie die Spider 6.x Setup.exe ausführen.



Zunächst werden Sie dazu aufgefordert, die Sprache für den Installationsassistenten auszuwählen. Klicken Sie hierfür auf die gewünschte Sprache in der Auswahlbox und bestätigen Sie Ihre Eingabe mit „OK“.

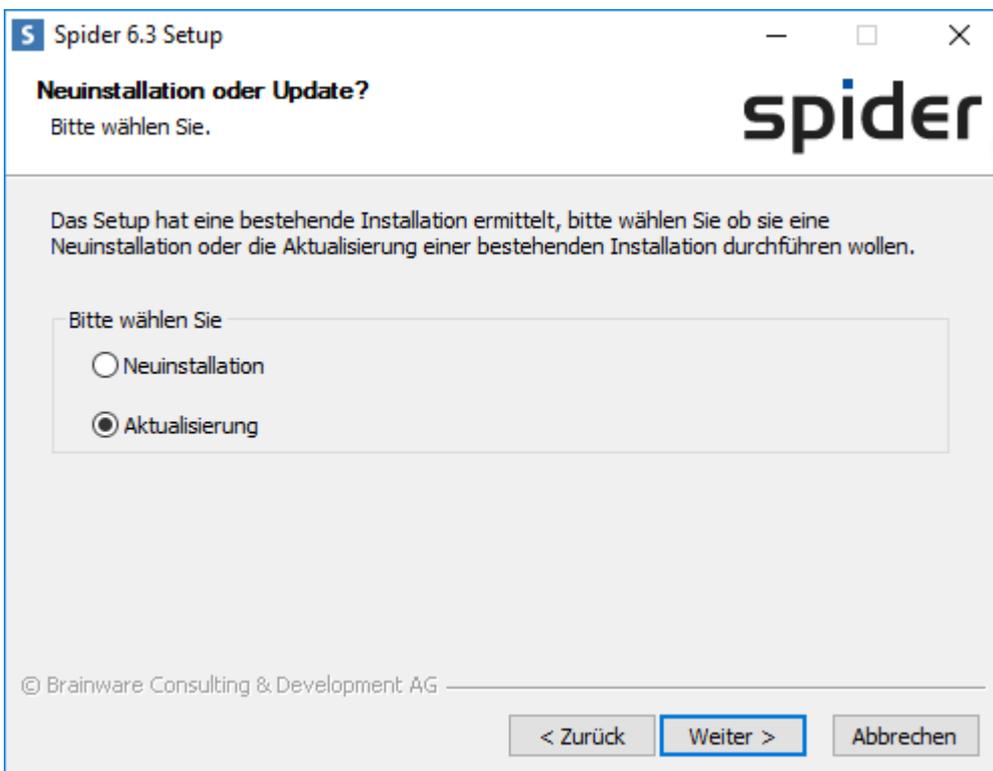


Anschließend öffnet sich der Installationsassistent für Spider. Er wird Ihnen in der gewählten Sprache bei der Installation des Setups behilflich sein. Klicken Sie nun auf „Weiter“ um die Installation zu starten.



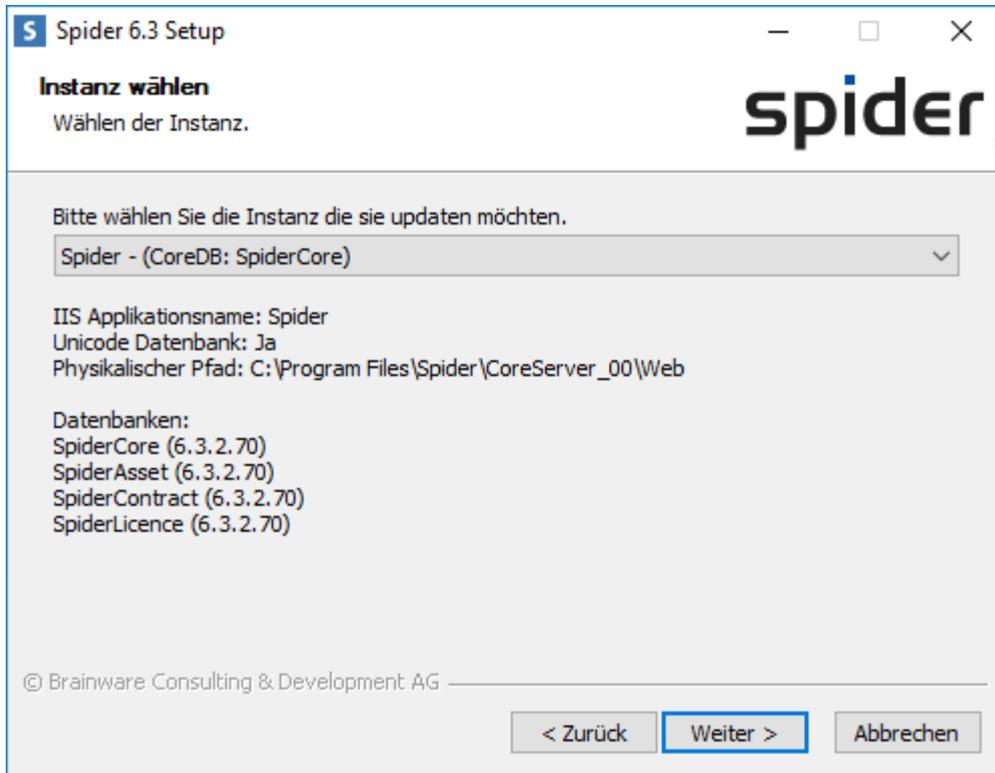
### 3.4.2 Aktualisierung

Falls bereits eine Instanz von Spider installiert ist, werden Sie aufgefordert sich für eine Art der Installation zu entscheiden.



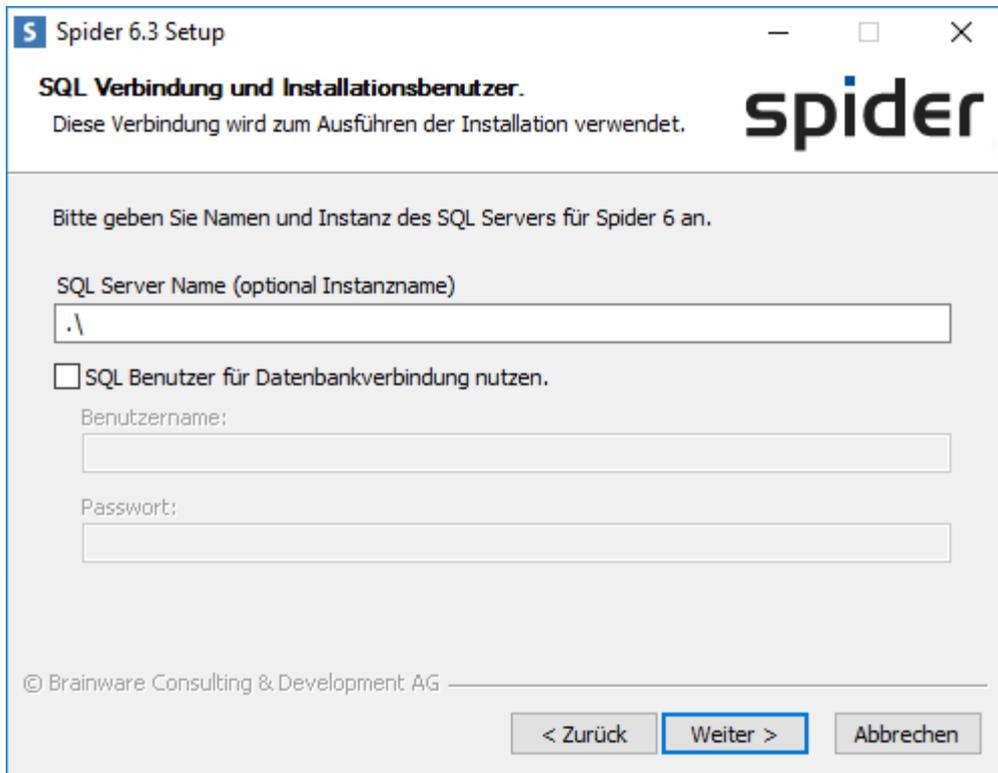
### 3.4.3 Auswahl der Instanz

Hier wird die Instanz ausgewählt, die aktualisiert werden soll. Ist nur eine Instanz vorhanden, ist diese im Auswahlfeld fest ausgewählt.



### 3.4.4 Datenbankserver

Im nun folgenden Schritt bittet Sie der Installationsassistent den Namen und die Instanz eines Microsoft SQL-Servers anzugeben, auf dem die Spider Datenbanken installiert werden sollen.



The screenshot shows a Windows-style window titled "Spider 6.3 Setup". The main heading is "SQL Verbindung und Installationsbenutzer." with a sub-note: "Diese Verbindung wird zum Ausführen der Installation verwendet." The "spider" logo is in the top right. The instruction reads: "Bitte geben Sie Namen und Instanz des SQL Servers für Spider 6 an." There is a text input field for "SQL Server Name (optional Instanzname)" containing ".\". Below it is a checkbox labeled "SQL Benutzer für Datenbankverbindung nutzen." which is unchecked. Underneath are three input fields: "Benutzername:", "Passwort:", and a third empty field. At the bottom left is the copyright notice "© Brainware Consulting & Development AG". At the bottom right are three buttons: "< Zurück", "Weiter >" (highlighted with a blue border), and "Abbrechen".

Der Benutzer, mit dem die Verbindung hergestellt wird, kann entweder der angemeldete (Windows-)Benutzer sein, oder es kann ein SQL Benutzer verwendet werden.

Für die Verwendung eines SQL Benutzers, ist das Kontrollkästchen **SQL Benutzer für Datenbankverbindung nutzen** zu markieren sowie die Felder **Benutzername** und **Passwort** mit den gewünschten Daten auszufüllen.

The screenshot shows the 'Spider 6.3 Setup' dialog box with the title 'SQL Verbindung und Installationsbenutzer'. Below the title, it says 'Diese Verbindung wird zum Ausführen der Installation verwendet.' and the 'spider' logo. The main instruction is 'Bitte geben Sie Namen und Instanz des SQL Servers für Spider 6 an.' There are three input fields: 'SQL Server Name (optional Instanzname)' containing '.\', a checked checkbox 'SQL Benutzer für Datenbankverbindung nutzen', a 'Benutzername:' field containing '<SQL Benutzer >', and a 'Passwort:' field with five dots. At the bottom, there are three buttons: '< Zurück', 'Weiter >' (highlighted with a red dashed border), and 'Abbrechen'. The footer contains '© Brainware Consulting & Development AG'.

**Hinweis** Die Benutzer müssen vorab auf dem Datenbankserver mit den entsprechenden Rechten eingerichtet werden. Abhängig von den Rechten, mit denen der Benutzer ausgestattet ist, stehen in den weiteren Folgedialogen verschiedene Optionen zur Verfügung.

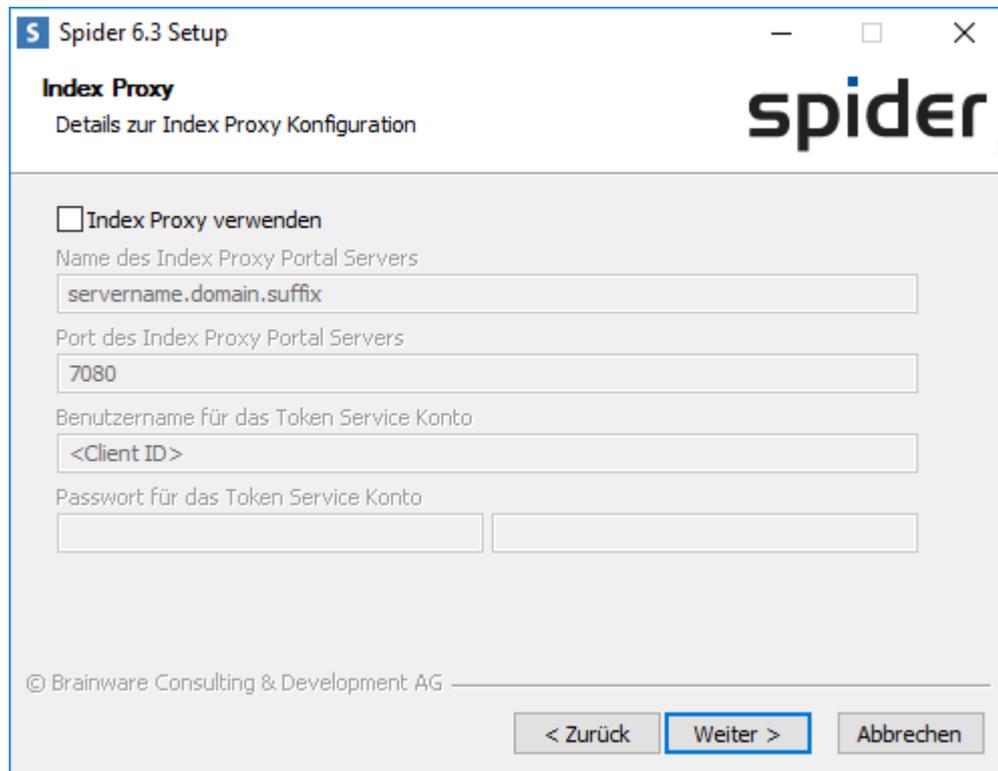
Feld	Beschreibung
SQL Server Name	Name des SQL Servers, ggfs. inklusive Instanznamen und Port z.B. SQLSERVER\SQLINSTANZ,1234
SQL Benutzer für Datenbankverbindung nutzen	Auswahl, ob der angemeldete Domainbenutzer verwendet wird oder ob ein SQL Benutzer verwendet werden soll.
Benutzername	Name des SQL Benutzers.
Passwort	Passwort des SQL Benutzers.

### 3.4.5 Index Proxy Server

Falls der Index Proxy Server vorher noch nicht konfiguriert worden ist, gibt es mit jeder Spider Aktualisierung die Möglichkeit dies nachzuholen.

Als nächstes kann gewählt werden, ob der Index Proxy während des Setups eingerichtet werden soll.

Der Index Proxy Server ist Teil der Enterprise Services Infrastructure (ESI).



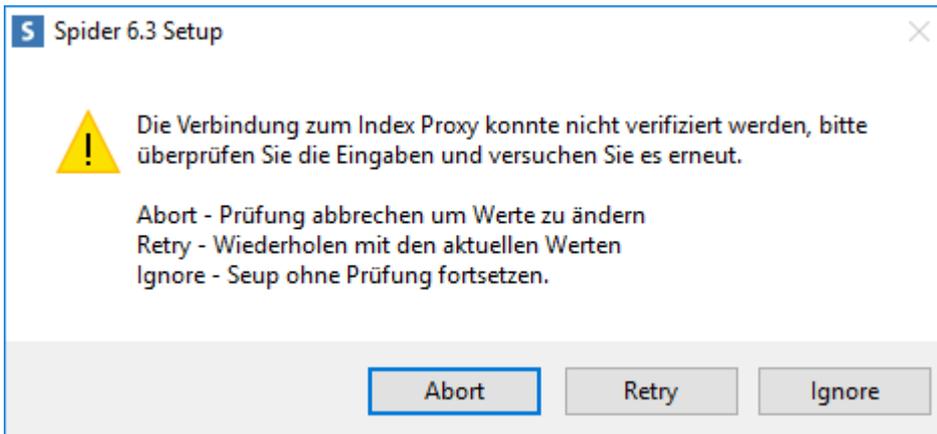
The screenshot shows the 'Spider 6.3 Setup' window with the 'Index Proxy' configuration screen. The window title is 'Spider 6.3 Setup' and the subtitle is 'Index Proxy Details zur Index Proxy Konfiguration'. The 'spider' logo is visible in the top right corner. The main content area contains a checkbox labeled 'Index Proxy verwenden' which is currently unchecked. Below this are four input fields: 'Name des Index Proxy Portal Servers' with the value 'servername.domain.suffix', 'Port des Index Proxy Portal Servers' with the value '7080', 'Benutzername für das Token Service Konto' with the value '<Client ID>', and 'Passwort für das Token Service Konto' with two empty input boxes. At the bottom left, there is a copyright notice '© Brainware Consulting & Development AG'. At the bottom right, there are three buttons: '< Zurück', 'Weiter >' (highlighted with a blue border), and 'Abbrechen'.

Durch das Markieren des Kontrollkästchens wird der Dialog aktiviert und die benötigten Angaben können eingetragen werden.

Feld	Beschreibung
Index Proxy verwenden	Aktivierung des Dialogs und Abfrage der Details
Name des Index Proxy Portal Servers	Name (FQDN) des Servers, auf dem der Index Proxy installiert ist,
Port des Index Proxy Portal Servers	Port (Default: 7080) auf dem der Index Proxy auf Anfragen reagiert
Benutzername des Token Service Kontos	Die Client ID des Benutzers der für die Verbindung zum Index Proxy verwendet werden soll, die Client ID wurde während der Installation des Index Proxy vergeben. Bei der Client ID ist auf Groß- und Kleinschreibung zu achten!
Passwort des Token Service Konto	Passwort das zur Client ID gehört.

Nach der Eingabe aller Daten und der Auswahl des **Weiter >** Buttons wird die Verbindung zum Index Proxy geprüft.

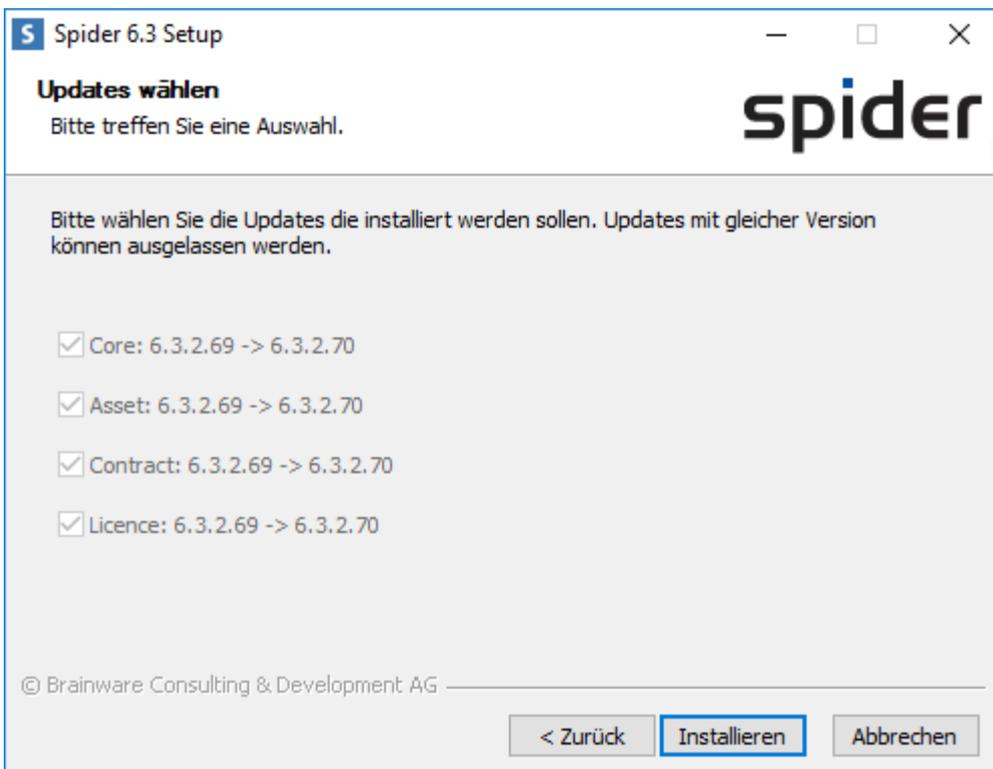
Sollte es zu einem Fehler kommen, wird die folgende Meldung angezeigt.



Schaltfläche	Beschreibung
Abort	Die Prüfung kann abgebrochen, die Werte korrigiert und dann erneut geprüft werden.
Retry	Erneuter Versuch mit den aktuellen Werten
Ignore	Der Dialog kann ohne positives Prüfergebnis verlassen werden. Die Installation wird mit den im Dialog angegebenen Werten durchgeführt.

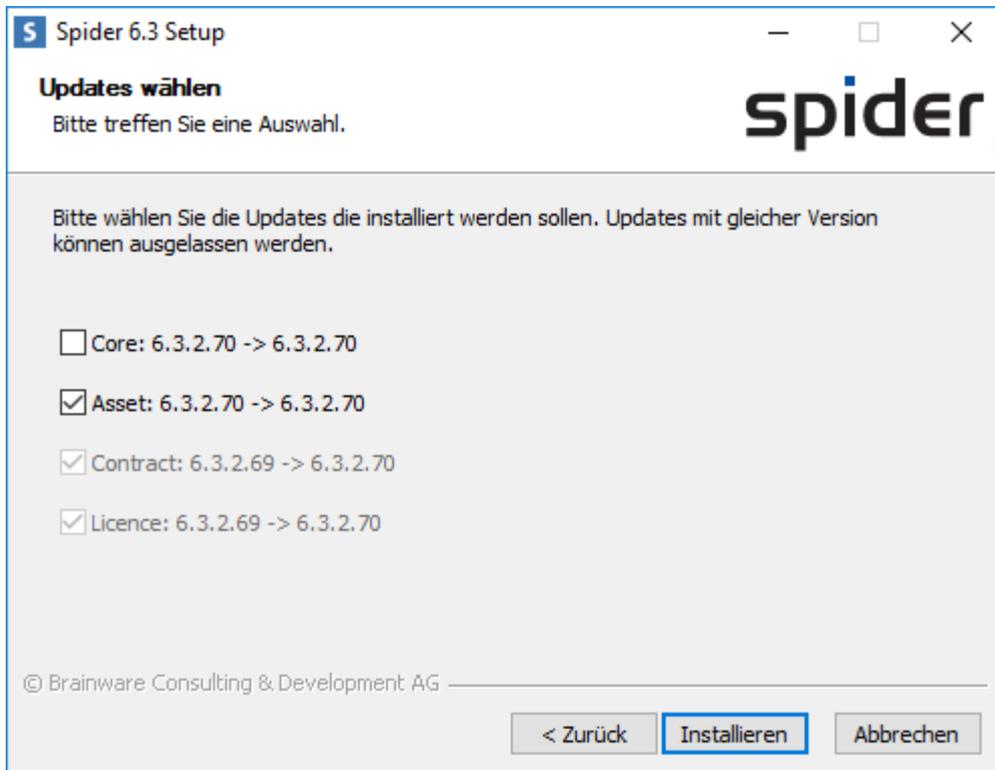
### 3.4.6 Updates wählen

Erfolgt die Aktualisierung zum ersten Mal, müssen alle Produkte zur Aktualisierung ausgewählt werden. Der Dialog ist entsprechend vorausgefüllt und lässt sich nicht mehr ändern.



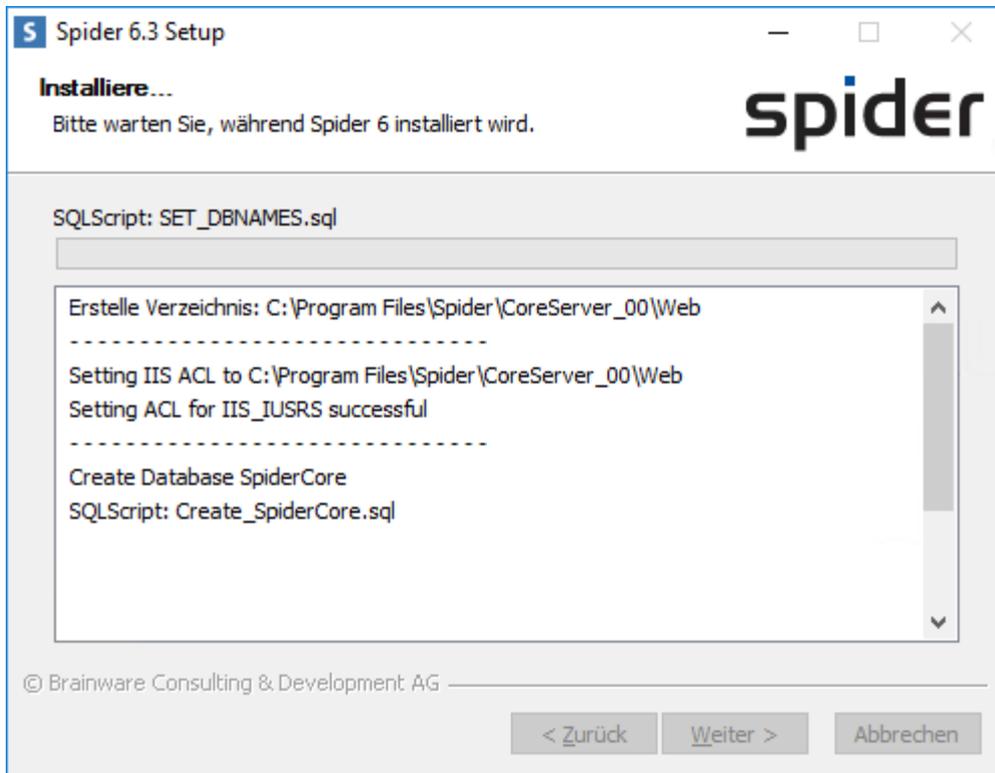
Falls eines der Produkte schon vorher aktualisiert wurde, kann es sein, dass nicht mehr alle Produkte zur Aktualisierung angeboten werden.

Wenn gewünscht, können diese trotzdem zur Aktualisierung ausgewählt werden.

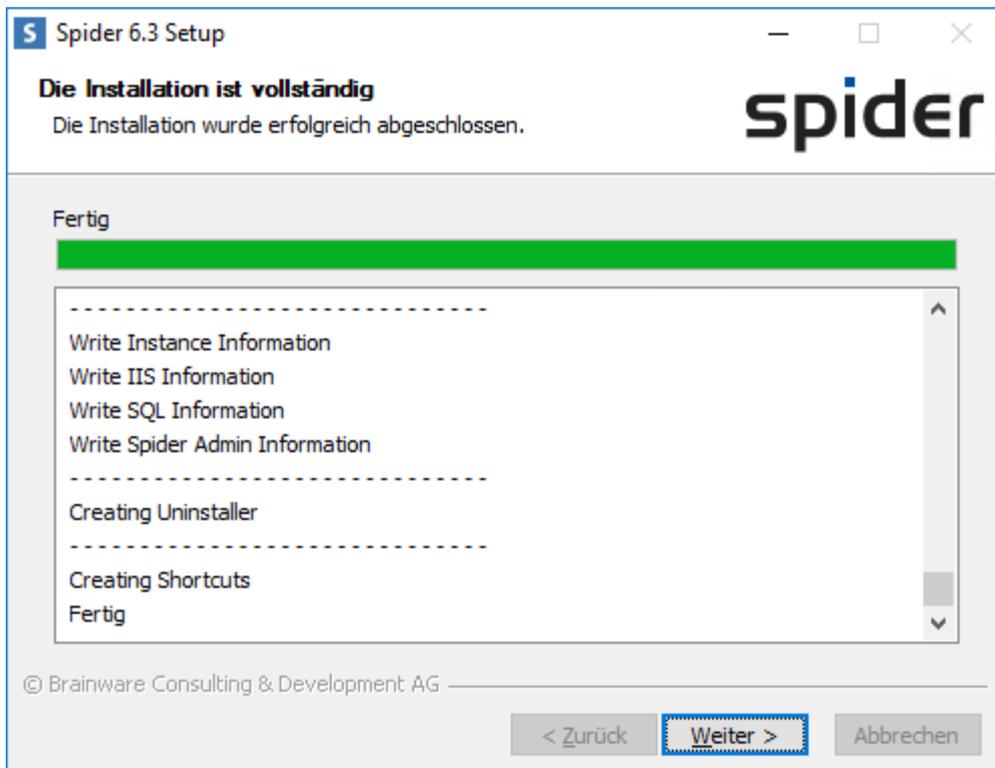


### 3.4.7 Installation

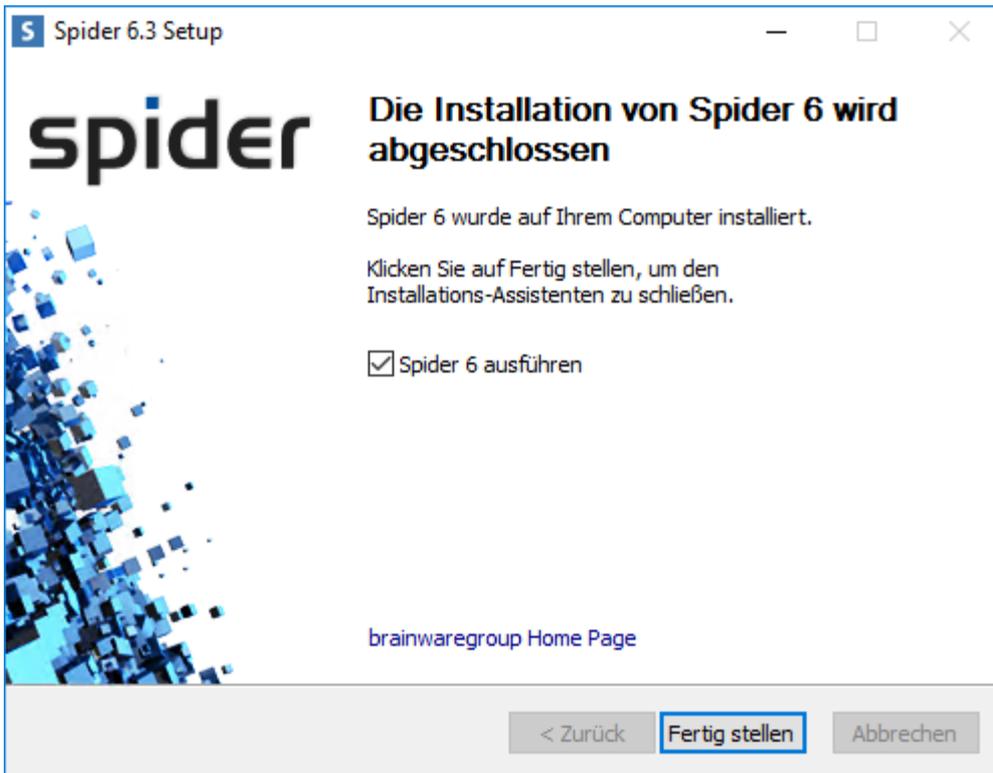
Der Installationsassistent installiert nun Spider. Dies kann einige Zeit in Anspruch nehmen. Bitte warten Sie, bis die Installation beendet ist.



Nach erfolgreichem Abschluss der Installation wird dies im Fortschrittsdialog angezeigt.



Das Setup ist erfolgreich abgeschlossen und kann beendet werden.



Nach Abschluss der Installation wird ein Logfile geschrieben. Details zu den Logfiles, sind im Kapitel [Logfiles](#) (siehe Seite 210) beschrieben.

## Spider Web API

Ab Spider 6.4 wird eine auf dem REST Standard basierende Programmierschnittstelle - Spider Web API - bereitgestellt. Sie dient einerseits der besseren Integration der Spider- und Columbus-Produkte untereinander, aber andererseits vor allem auch der Integration von Spider mit bestehenden Kunden- und Fremdsystemen. Die Spider Web API ermöglicht das Ausführen von Suchabfragen und das Anfordern von Berichten sowie das Anlegen, Ändern, Zuweisen, Lösen und Löschen von Objekten. Die Spider Web API ist fester Bestandteil von Spider 6.4 und steht nach der Installation sofort zur Verfügung. Die Spider Web API verfügt über einen eigenen Dokumentationsbereich mit Informationen zu allen Funktionen, ihren Parametern und Rückgabewerten. Zu Test- und Entwicklungszwecken steht der Spider Web API Explorer zur Verfügung. Dieser ermöglicht einen einfachen Einstieg in die Nutzung der Schnittstelle und soll auch die Analyse von eventuellen Fehlern erleichtern. Für tiefere Entwicklungsarbeiten, wie z.B. den Aufbau neuer Schnittstellen, steht Swagger zur Verfügung.

### 4.1 Technische Grundlagen

#### 4.1.1 Rest API

Die Spider Web API basiert auf dem Prinzip des restful Services. D.h. jede definierte URL repräsentiert eine Ressource, auf die eine oder mehrere HTTP-Methoden angewendet werden können. Verwendete HTTP -Methoden sind GET zum

Erhalten, PUT zum Bearbeiten, POST zum Hinzufügen und DELETE zum Löschen von Ressourcen. Die Kommunikation kann je nach Kundenkonfiguration über HTTP oder HTTPS erfolgen. Die Spider Web API empfängt und versendet Daten im JSON-Format. Die Daten sind UTF-8 codiert.

## 4.1.2 Versionierung

Die Funktionen der Spider Web API sind versioniert. Die Version der angeforderten Funktion wird in der URL angegeben. Wird die Version in der URL weggelassen, wird die neueste Version der Funktion verwendet.

### Beispiel

Assetsuche ohne Versionsangabe

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset
```

Assetsuche mit Versionsangabe (V1)

```
http(s)://{ServerName}/{SpiderApiInstance}/v1/{Mandator}/search/asset
```

## 4.1.3 Mandantenauswahl

Viele Funktionen der Spider Web API benötigen einen Mandanten für die Durchführung. Die Bezeichnung oder ID des Mandanten muss dann an der richtigen Stelle der URL mitgegeben werden.

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset
```

Die Liste der für den Benutzer verfügbaren Mandanten kann entsprechend der Beschreibung im Kapitel [Mandator](#) (siehe Seite 114) abgefragt werden.

## 4.1.4 Request

Jedem Request an die Spider Web API kann man Parameter mitgeben. Die möglichen Parameter unterscheiden sich je nach Ressource. Allerdings kann jedem Request der Parameter CultureCode mitgegeben werden, welcher die Sprache von Hinweisen und Fehlermeldungen sowie übersetzbarer Ergebnisse steuert.

Die Parameter werden im Allgemeinen als URL Parameter übermittelt. Nur die Daten und Parameter beim Erzeugen und Editieren von Entitäten müssen im Request-Body mitgegeben werden.

### Beispiel

Request mit Parametern in der URL für eine Asset-Suche:

```
GET http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset?culturecode=de&orderby=assetno&distinct=true
```

Request mit Parametern in dem Request Body zum Bearbeiten des Asset mit der ID 5:

```
PUT http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset/5
```

mit dem Body:

```
{
  "fields": {
    "AssetNo": "Test",
    "CPUCount": 1,
    "Status": 1
  }
}
```

## 4.1.5 Response

Bei jedem Request an die Spider Web API kommt eine standardisierte Response zurück. Diese enthält eine self-URL, die Parameter des Requests, die Bearbeitungsdauer des Requests und falls vorhanden Hinweise und Fehlermeldungen, die bei der Durchführung des Requests erzeugt wurden.

### Beispiel

Eine Asset-Suche liefert das Ergebnis

```
{
  "Count": 2204,
  "Result": [
    {
      "id": 2,
      "assetNo": "Access card-0002",
      "hostname": null,
      "assetType": "Zugriffskarte",
      "status": "Aktiv",
      "functionUnitID": null,
      "functionUnit": null
    }
  ],
  "IsResultTruncated": false,
  "Notifications": [],
  "Duration": 713,
  "SelfLink": "http://www.SpiderServer.com/SpiderWebApi/100/search/Asset?MaxRows=1&Firstrow=1&OrderBy=AssetNo",
  "Parameter": {
    "Conditions": [],
    "MaxRows": 1,
    "FirstRow": 1,
    "OrderBy": "AssetNo",
    "Distinct": false,
    "Fields": [],
    "AllFields": false,
    "CultureCode": null
  }
}
```

Berechtigte Requests der Spider Web API geben immer den HTTP Statuscode 200 zurück. Die einzige Ausnahme bilden erfolgreiche Requests zum Hinzufügen von Objekten. Diese werden mit dem HTTP Statuscode 201 beendet.

Unberechtigte Requests geben den HTTP Statuscode 401 zurück.

## 4.2 Authentifizierung

Für die Authentifizierung von Zugriffen gegenüber der Spider Web API wird ein Token verwendet. Dieser Token wird während seiner Lebenszeit mit jedem Request mitgegeben. Durch den Token müssen die Anmeldedaten nicht bei jedem Zugriff mitgegeben werden.

## 4.2.1 Generierung eines Token

---

Den Token zur Authentifizierung kann man durch einen GET-Request mit Basic Authentication erhalten. Der Token wird dann im Header der Response zurückgegeben.

Bei der Basic Authentication kann der Username und das Passwort direkt in der Request-URL mitgegeben werden.

```
http(s)://{UserName}:{Password}@{ServerName}/{SpiderApiInstance}/token
```

Alternativ kann Username und Passwort im Header des Requests übermittelt werden.

```
http(s)://{ServerName}/{SpiderApiInstance}/token
```

Hierfür wird Username und Passwort mit einem Doppelpunkt verknüpft. Der entstandene Text wird in Base64 umgewandelt und unter Authorization übergeben.

**Username:Passwort** wird durch Base64 zu **VXNlcm5hbWU6UGFzc3dvcnQ=** welches nach dem Schlüsselwort Basic unter Authorization im Header stehen muss.

```
Host: ServerName
User-Agent: WebAPI-Accessor
Authorization: Basic VXNlcm5hbWU6UGFzc3dvcnQ=
```

## 4.2.2 Verwendung eines Token

---

Der Token muss im Header jedes weiteren Requests mitgegeben werden. Die Spider Web API ermittelt über den Token den Spider Nutzer und die zugehörigen Berechtigungen.

### Beispiel

Im Request-Header wird unter Authorization der Token übergeben.

```
Host: ServerName
User-Agent: WebAPI-Accessor
Authorization: Token ed901b25-2f7e-4461-86bd-604212f4e859
Content-Length: 220
Content-Type: application/json; charset=utf-8
```

## 4.2.3 Gültigkeit eines Token

---

Der Token ist zur Steigerung der Sicherheit nur eine begrenzte Zeit gültig. Die genaue Länge in Minuten kann in der Spider Core Config eingestellt werden und ist im Standard mit 15 Minuten definiert.

Nach dem Ablauf des Tokens ist keine weitere Authentifizierung mit dem Token möglich. Bereits laufende Requests werden weiter abgearbeitet. Ein neuer Token kann über die Tokengenerierung erzeugt werden.

## 4.2.4 Verwaltung der Token

---

Die Token werden zentral verwaltet, wodurch in einem webfarm-Szenario alle Spider Web API Instanzen auf die gleichen Token zugreifen können. Daher ist es für einen Benutzer irrelevant, welche Instanz seine Anfrage bearbeitet.

Weiterhin können einem Benutzer mehrere gültige Token zugeordnet sein, da bspw. verschiedene Services mit dem gleichen Login gleichzeitig auf die Spider Web API zugreifen können.

## 4.3 Admin Bereich

---

Im Admin-Bereich der Spider Web API können berechtigte Benutzer administrative Objekte von Spider auflisten, auslesen, verknüpfen und teilweise bearbeiten, hinzufügen oder löschen.

### Einschränkungen

Der Admin-Bereich deckt nicht denselben Funktionsumfang wie die Spider Web Oberfläche ab. Daher gibt es keinen Zugriff auf Applikationsformulare, Menü-, Rollen-, Active-Directory-, Passwortregel- und Systemnachricht-Objekte.

### 4.3.1 Mandator

---

Mit der Spider Web API können Mandanten aufgelistet, gelesen, bearbeitet, hinzugefügt und gelöscht werden. Zudem sind die Zuordnung von Usern zu einem Mandanten und das entsprechende Trennen möglich.

### 4.3.2 User

---

Mit der Spider Web API können Benutzer aufgelistet, gelesen, bearbeitet, hinzugefügt und gelöscht werden. Zudem sind die Zuordnung von Mandanten, Profilen und Geschäftseinheiten zu einem User und das entsprechende Trennen möglich.

### 4.3.3 Profile

---

Mit der Spider Web API können Profile aufgelistet und gelesen werden. Zudem sind die Zuordnung von Benutzern und Geschäftseinheiten zu einem Profil und das entsprechende Trennen möglich. Das Löschen und Hinzufügen von Profilen ist nicht vorgesehen.

## 4.4 Entity Bereich

---

Im Entity-Bereich der Spider Web API können Benutzer auf die verschiedenen Spider Entitäten zugreifen. Die Berechtigung eines Benutzers an Entitäten und deren Feldern wird durch seine zugeordneten Rollen definiert.

### Einschränkung

Nur Entitäten, deren technische Bezeichnung über alle Spider Applikationen eindeutig ist, können über die Spider Web API angesprochen werden. Darüber hinaus sind einige Entitäten vom Zugriff explizit ausgenommen, weil sie bspw. nur eine verknüpfende Funktion zwischen zwei anderen Entitäten oder eine administrative Funktion haben.

### 4.4.1 Entity-Zugriff

---

Abhängig von seinen vorhandenen Berechtigungen kann ein Benutzer Entitäten auslesen, bearbeiten, hinzufügen oder löschen. Custom-Entitäten sind auch in der Spider Web API verfügbar, solange sie den Einschränkungen nicht widersprechen. Allerdings ist zu beachten, dass hier nur Businesslogik aus zugeordneten Custom-Regeln und nicht aus den Custom-Kernelobjekten angewandt wird.

Die Objekte werden über ihre technische Bezeichnung und ihre ID referenziert. Typisierte Objekte können über die Bezeichnung des Elternobjektes angesprochen werden.

### Beispiel

Ein Server mit der Asset-ID 1 kann direkt über den Typ angezeigt werden.

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/entity/server/1
```

Alternativ kann man das gleiche Ergebnis über das Elternobjekt Asset erhalten.

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/entity/asset/1
```

## 4.4.2 Entity-History-Zugriff

---

Für viele Entitäten führt Spider eine Änderungshistorie. Diese kann nach Änderungszeitraum und geänderten Feldern gefiltert und aufgelistet werden.

## 4.4.3 Entity-Dokumenten-Zugriff

---

Vielen Entitäten können in Spider Dokumente zugewiesen werden. Dieses erfolgt innerhalb einer Ordnerstruktur, die mit der Spider Web API navigiert werden kann. Auch die verschiedenen Versionen der Dokumente sind erreichbar. Weiterhin können Dokumente angelegt und neue Versionen hinzugefügt werden.

## 4.4.4 Entity-Wiedervorlagen-Zugriff

---

Einigen Entitäten können in Spider Wiedervorlagen zugewiesen werden. Diese sollen im Kontext der Entität angelegt, angezeigt, bearbeitet und gelöscht werden.

## 4.4.5 Entity-Relationship-Zugriff

---

Viele Entitäten in Spider sind miteinander verknüpft. Dieses kann sowohl über Relationship-Tabellen oder direkte Referenz über ein Feld der Entität geschehen. Mit der Spider Web API kann man die zu einer Entität zugeordneten Entitäten auflisten, eine neue Zuordnung erzeugen und eine solche bestehende Zuordnung auflösen.

### Einschränkungen

In zwei Sonderfällen ist der Zugriff auf Beziehungen zwischen Entitäten in der Spider Web API nicht möglich,

- wenn die Beziehung zwischen zwei Entitäten des gleichen Typs ist (bspw. Verknüpfung zweier Vertragspartner über eine Rechtsnachfolge) oder
- wenn die Beziehung über eine typisierte Beziehungsentität realisiert ist (bspw. Verknüpfung zweier Funktionseinheiten über eine Funktionseinheitenbeziehung).

### Relationship-Definition

Die Möglichkeit des Customizing von Entitäten und deren Beziehungen erfordert, dass die Liste der möglichen Entity-Relationships erweiterbar definiert ist. Diese Informationen sind in der Spider Core Datenbank innerhalb der Tabelle webapi.EntityRelationships festgehalten.

Eine Entity-Relationship ist in diesem Fall eine eindeutige Beziehung von einer Entität zu einer anderen Entität. Der Rückweg von der anderen Entität zu der einen Entität ist eine eigene Relationship.

Eine Relationship-Definition besteht immer aus

- einer MainEntity,
- einer RelatedEntity,
- einem Suchview,
- dem Feldnamen der MainEntityID im Suchview

- und der Applikation, auf deren Datenbank der Suchview ausgeführt werden muss.

Durch diese Informationen ist es möglich die zugeordneten Entitäten aufzulisten.

Zusätzlich wird bei einer Verknüpfung über eine eigene Entität

- der Name der Entität,
- der Feldname des ID-Feldes der MainEntity,
- der Feldname des ID-Feldes der RelatedEntity,
- falls nötig einem SQL-Skript zur Übersetzung des ID-Wertes von MainEntity (z.B. Ermittlung der FunctionUnitID bei der Relationship von Asset zu Employee),
- falls nötig einem SQL-Skript zur Übersetzung des ID-Wertes von RelatedEntity

Durch Angabe dieser Informationen können Zuordnungen erstellt und aufgelöst werden.

## 4.5 Search Bereich

Im Search-Bereich der Spider Web API können Benutzer auf die Suchen zu den verschiedenen Spider Entitäten zugreifen. Das Suchergebnis und die Suchfelder können durch die Berechtigung des Benutzers und seine zugeordneten Rollen beeinflusst werden.

### Beispiel

Ergebnis einer Assetsuche

```
{
  "count": 201,
  "result": [
    {
      "id": 2,
      "assetNo": "LAPTOP-OBST003.domain.corp",
      "hostname": "LAPTOP-OBST003",
      "assetType": "Laptop",
      "status": "Aktiv",
      "functionUnitID": 2,
      "functionUnit": "LAPTOP-OBST003.domain.corp"
    }
  ],
  "isResultTruncated": false,
  "notifications": [],
  "duration": 4409,
  "selfLink":
  "http://www.SpiderServer.com/SpiderWebApi/v1/0/search/asset?maxRows=1&firstRow=1&distinct=true&allFields=false",
  "parameter": {
    "conditions": [],
    "maxRows": 1,
    "firstRow": 1,
    "orderBy": "",
    "distinct": true,
    "fields": [],
    "allFields": false,
    "cultureCode": null
  }
}
```

## Einschränkung

Nur Entitäten, deren technische Bezeichnung über alle Spider Applikationen eindeutig ist, können im Search Bereich über die Spider Web API angesprochen werden. Weiterhin sind einige Entitäten vom Zugriff explizit ausgenommen, weil sie bspw. nur eine verknüpfende Funktion zwischen zwei anderen Entitäten oder eine administrative Funktion haben.

## 4.5.1 Parametrisierung

Die Parameter der Suche werden in der URL übergeben.

### Beispiel

Eine Liste der ersten 100 Assets mit allen Feldern sortiert nach der Assetnummer.

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset?allfields=true&firstrow=1&maxrows=100&orderby=assetno
```

Parameter	Beschreibung
AllFields	Der URL-Parameter muss einen booleschen Wert enthalten, der definiert ob alle Felder in der Suche mitgegeben werden sollen. Der Wert ist im Standard false.
Condition	<p>Dieser URL-Parameter kann mehrfach in der URL enthalten sein. Jeder Parameter enthält einen Filter. Ein Filter besteht aus</p> <ul style="list-style-type: none"> <li>• einer logischen Verknüpfung,</li> <li>• einem optionalen Flag für Negierung,</li> <li>• einem Feldnamen,</li> <li>• einem Vergleichstypen</li> <li>• und einem Vergleichswert.</li> </ul> <p>Die einzelnen Bestandteile müssen durch ein Pipe Zeichen („ “) separiert sein.</p> <p><b>Beispiel</b></p> <p>Die Liste von nicht aktiven Assets aus der Geschäftseinheit „A“:</p> <pre>http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset?condition=and not status = active&amp;condition=and  LegalEntityPath = A</pre>
CultureCode	Dieser URL-Parameter muss einen Text enthalten, der einen valides Sprachkürzel repräsentiert. Diese Spracheinstellung wird für übersetzbare Ergebnisse der Suche verwendet. Die Standardsprache entspricht der Serversprache.
Distinct	Der URL-Parameter muss einen booleschen Wert enthalten, der definiert ob inhaltlich gleiche Zeilen mehrfach im Ergebnis vorhanden sein dürfen. Der Wert ist im Standard false.

Parameter	Beschreibung
Fields	<p>Der URL-Parameter sollte eine kommaseparierte Liste von Feldnamen aus der Suche enthalten. Die Standardfelder unterscheiden sich je Suchentität.</p> <p>Der Parameter Fields wird ignoriert, wenn der Parameter AllFields auf true gesetzt ist.</p> <p><b>Beispiel</b></p> <p>Die Liste von nicht Assets mit den Feldern ID, Assetnr., Assettyp, AssetStatusID, Status, LegalEntityID und LegalEntityPfad:</p> <pre>http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/search/asset?fields=id,assetno,assettype,assetstatusid,status,legalentityid,legalentitypath</pre>
FirstRow	Der URL-Parameter muss einen numerischen Wert enthalten, der definiert ab welcher Zeile das Ergebnis übermittelt werden soll. Der Wert ist im Standard 1.
MaxRows	Der URL-Parameter muss einen numerischen Wert enthalten, der definiert wie viele Zeilen maximal im Ergebnis enthalten sein sollen. Der Wert ist im Standard 500.
OrderBy	Der URL-Parameter muss einen Feldnamen enthalten, der die Sortierspalte festlegt. Optional kann die Sortierrichtung durch Hinzufügen von „ASC“ für aufsteigende Sortierung und „DESC“ für absteigende Sortierung definiert werden. Im Standard wird die Sortierung der Datenbank überlassen.

## 4.6 Report Bereich

Im Report-Bereich der Spider Web API können Benutzer auf die in Spider definierten Berichte zugreifen. Auf einen Bericht kann über dessen ID oder seine GUID zugegriffen werden.

### Beispiel

Wenn der Bericht „Assets mit veraltetem Scandatum“ die ID 1 hat, kann er wie folgt erreicht werden:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/report/1
```

Alternativ kann man den gleichen Bericht unter seiner GUID abfragen:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/report/3428E63A-082B-3C9D-D318-04AFA7DCCFBD
```

Der Bericht würde ein ähnliches Ergebnis liefern:

```
{
  "meta": {
    "displayCategory": "Inventory",
    "displayName": "Assets with outdated scan date",
    "displayDescription": "Assets with outdated scan date (older than 90 days)",
    "guid": "3428e63a-082b-3c9d-d318-04afa7dccfbd",
    "category": "Inventory",
    "name": "Assets with outdated scan date",
    "description": "Assets, which have a scan date that is at least 90 days old",
    "id": 5,
    "applicationName": "Spider Asset",
    "applicationGuid": "38f96e97-c314-44b8-b137-c966d35a3ba0"
  },
  "count": 27,
  "result": [
```

```

{
  "assetID": 222,
  "asset": "DP000001",
  "hostName": "heraetrdy",
  "assetStatus": "Aktiv",
  "invLastScanDate": null,
  "swrUpdateDate": null
}
],
"isResultTruncated": false,
"notifications": [],
"duration": 553,
"selfLink": "http://www.SpiderServer.com/SpiderWebApi/v1/0/report/3428e63a-082b-3c9d-d318-04afa7dccfbd",
"parameter": {
  "conditions": [],
  "maxRows": 1,
  "firstRow": 1,
  "orderBy": "",
  "distinct": false,
  "fields": [],
  "allFields": false,
  "cultureCode": null
}
}

```

## Einschränkung

Die Spider Web API erlaubt nur Zugriff auf abfragebasierte Berichte.

### 4.6.1 Parametrisierung

Parameter des Berichtes werden in der URL übergeben.

Parameter	Beschreibung
AllFields	Der URL-Parameter muss einen booleschen Wert enthalten, der definiert ob alle Felder in dem Bericht mitgegeben werden sollen. Der Wert ist im Standard false.
Condition	<p>Dieser URL-Parameter kann mehrfach in der URL enthalten sein. Jeder Parameter enthält einen Filter. Ein Filter besteht aus</p> <ul style="list-style-type: none"> <li>• einer logischen Verknüpfung,</li> <li>• einem Flag für Negierung,</li> <li>• einem Feldnamen,</li> <li>• einem Vergleichstypen</li> <li>• und einem Vergleichswert.</li> </ul> <p><b>Beispiel</b></p> <p>Berichtsergebnisse müssen in der Assetnr. den Text „srv“ enthalten  condition=AND  AssetNo LIKE %srv%</p> <p>Berichtsergebnisse dürfen keinen „Active“ Status haben  Condition=AND NOT Status = Active</p>

Parameter	Beschreibung
CultureCode	Dieser URL-Parameter muss einen Text enthalten, der einen valides Sprachkürzel repräsentiert. Diese Spracheinstellung wird für übersetzbare Ergebnisse des Berichtes verwendet. Die Standardsprache entspricht der Serversprache.
Distinct	Der URL-Parameter muss einen booleschen Wert enthalten, der definiert ob Zeilen doppelt im Ergebnis vorhanden sein dürfen. Der Wert ist im Standard false.
Fields	Der URL-Parameter muss eine Kommaseparierte Liste von Feldnamen aus der Suche enthalten. Die Standardfelder unterscheiden sich je Suchentität. Der Parameter Fields wird ignoriert, wenn der Parameter AllFields auf true gesetzt ist.
FirstRow	Der URL-Parameter muss einen numerischen Wert enthalten, der definiert ab welcher Zeile das Ergebnis übermittelt werden soll. Der Wert ist im Standard 1.
MaxRows	Der URL-Parameter muss einen numerischen Wert enthalten, der definiert wie viele Zeilen maximal im Ergebnis enthalten sein sollen. Der Wert ist im Standard 500.
OrderBy	Der URL-Parameter muss einen Feldnamen enthalten, der die Sortierspalte festlegt. Optional kann die Sortierrichtung durch Hinzufügen von „ASC“ für aufsteigende Sortierung und „DESC“ für absteigende Sortierung definiert werden. Im Standard wird die Sortierung der Datenbank überlassen.

## 4.7 Meta Bereich

Im Meta-Bereich der Spider Web API können Benutzer auf die Metadaten für die anderen Bereiche zugreifen.

Diese Metadaten umfassen je nach Funktion neben den Listen der verfügbaren Mandanten, Entitäten, Suchen und Berichten auch Information über die Felder in Suchen und Berichten sowie zum Erzeugen und Bearbeiten von Entitäten.

### 4.7.1 Mandator

Im Meta-Bereich kann sich der Benutzer die ihm zugeordneten Mandanten auflisten lassen.

```
http(s)://{ServerName}/{SpiderApiInstance}/meta/mandator
```

### 4.7.2 Admin

Im Meta-Bereich kann sich der Benutzer die Metadaten zum Anlegen und Bearbeiten von Mandanten und Benutzern anzeigen lassen.

#### Beispiel

Die nötigen Metainformationen zum Anlegen eines Mandanten erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/meta/admin/mandator
```

Metainformationen zur Bearbeitung des Benutzers mit der ID 5 erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/meta/admin/user/5
```

### 4.7.3 Entity

Im Meta-Bereich kann sich der Benutzer die in der Web API verfügbaren Entitäten anzeigen lassen. Zusätzlich gibt es Funktionen zum Anzeigen der der Metadaten zum Anlegen und Bearbeiten.

#### Beispiel

Die Liste der möglichen Entitäten ist verfügbar unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entities
```

Die nötigen Metainformationen zum Anlegen eines Herstellers erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entity/manufacturer
```

Metainformationen zur Bearbeitung des Vertrags mit der ID 5 erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entity/contract/5
```

Außerdem gibt es eine Auflistung der verfügbaren, der zugehörigen Suchfeldmetadaten und Metadaten zum Anlegen und Bearbeiten von Beziehungen.

#### Beispiel

Die Liste der Entitäten, die mit Asset verknüpfbar sind, ist verfügbar unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entity/asset/relationship
```

Die nötigen Metainformationen zu Suchfeldern der Liste von Software, die zu einem Asset zugeordnet sind:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entity/asset/relationship/software
```

Zudem gibt es eine Auflistung von Spider EntityTypes, die nicht Entitäten sind und ihren jeweils möglichen Werten.

#### Beispiel

Die Liste der EntityTypes die in einem Vertrag verwendet werden, ist verfügbar unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/entity/contract/entityType
```

## 4.7.4 Search

---

Im Meta-Bereich kann sich der Benutzer die in der Web API verfügbaren Entitäten und die zugehörigen Ergebnisfelder anzeigen lassen.

#### Beispiel

Die Liste der möglichen Suchentitäten ist verfügbar unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/search
```

Die Feldinformationen zu einer Herstellersuche erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/search/manufacturer
```

## 4.7.5 Report

---

Im Meta-Bereich kann sich der Benutzer die in der Web API verfügbaren Berichte und die zugehörigen Ergebnisfelder anzeigen lassen.

#### Beispiel

Die Liste der möglichen Berichte ist verfügbar unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/report
```

Die Feldinformationen zu dem Bericht „Assets mit veraltetem Scandatum“ erhält man unter:

```
http(s)://{ServerName}/{SpiderApiInstance}/{Mandator}/meta/search/3428E63A-082B-3C9D-D318-04AFA7DCCFBD
```

## 4.8 Versionsinformationen

Falls ein Benutzer eine Information über die verwendete Spider-Version benötigt, so gibt es hierfür 2 Ressourcen.

### 4.8.1 Einfache Versionsinformation

Die einfache Versionsinformation liefert die 4-teilige Build-Nummer von Spider und ist unter der folgenden URL abfragbar.

```
http(s)://{ServerName}/{SpiderApiInstance}/version
```

#### Beispiel

```
{
  "build": "6.4.0.301",
  "notifications": [],
  "duration": 0,
  "selfLink": "http://www.SpiderServer.com/SpiderWebApi/version",
  "parameter": null
}
```

### 4.8.2 Erweiterte Versionsinformation

Die erweiterte Versionsinformation liefert zusätzlich zur Build-Nummer noch die Recognition-Version, die Edition und ob die Instanz eine Vollupdateversion ist. Sie ist unter folgende URL erreichbar.

```
http(s)://{ServerName}/{SpiderApiInstance}/versioninfo
```

#### Beispiel

```
{
  "build": "6.4.0.301",
  "setup": "6.4.0-4154089",
  "edition": "Spider Enterprise",
  "recognition": "1.1805.2.2",
  "isFullUpdate": true,
  "settingsVersion": "6.4.0.301",
  "notifications": [],
  "duration": 267,
  "selfLink": "http://www.SpiderServer.com/SpiderWebApi/versioninfo",
  "parameter": null
}
```

## 4.9 Swagger

Spider Web API verwendet Swagger als interaktive API Dokumentation. Swagger ist das weltweit größte Framework von API-Entwicklertools für die OpenAPI-Spezifikation (OAS) und ermöglicht die Entwicklung über den gesamten API-Lebenszyklus hinweg, von Design und Dokumentation bis hin zu Test und Bereitstellung.

Swagger ist unter folgender URL erreichbar:

```
http(s)://{ServerName}/{SpiderApiInstance}/swagger/ui/index
```

## 4.9.1 OpenAPI Document

---

Das OpenAPI Document ist ein Dokument (oder eine Gruppe von Dokumenten), das eine API definiert oder beschreibt. Es ist sowohl von Menschen als auch von Maschinen lesbar. Das Dokument kann von einem Entwickler verwendet werden, um automatisch eine Schnittstelle zur Spider Web API zu generieren.

Das Dokument ist über eine URL abrufbar, die im Kopfbereich der Swagger Oberfläche abgelesen werden kann. Die URL hat üblicherweise die folgende Form:

```
http(s)://{ServerName}/{SpiderApiInstance}/swagger/docs/v1
```

## 4.9.2 Swagger UI

---

Über die Swagger Oberfläche kann der Benutzer direkt die Funktionen der Spider Web API zugreifen.

**Hinweis** Wenn Sie Swagger im Internet Explorer benutzen möchten, müssen Sie den Kompatibilitätsmodus ausschalten. Swagger ist nicht mit dem eingeschalteten Kompatibilitätsmodus des Internet Explorer kompatibel.

---

### Token Authentifizierung

Die Funktionen benötigen auch beim Zugriff aus der Swagger Oberfläche eine gültige Authentifizierung. Einen gültigen Token kann der Benutzer unter

```
http(s)://{ServerName}/{SpiderApiInstance}/swagger/ui/index#!/Token/Authenticate_Authenticate
```

generieren. Dort muss im Feld Authorization eine gültige Basisauthentifizierung angegeben werden. Diese hat das Format

```
„Basic {Base64 Kodierung von „{UserName}:{Passwort}“}“
```

Zur Kodierung von Benutzername und Passwort in Base64 gibt es im Internet diverse Tools und Onlineresourcen.

**Beispiel**

GET
/token
The service returns an access token.

**Implementation Notes**

The service returns an access token to be used for authentication at other services. It takes the user credentials given by the basic authentication information in the header. If those are verified by the Spider application the access token will be generated and returned inside the response's header.

**Response Class (Status 200)**

OK

Model Example Value

```

{
  "notifications": [
    {
      "category": 0,
      "message": "string"
    }
  ],
  "duration": 0,
  "selfLink": "string",
  "parameter": {
    "cultureCode": "string"
  }
}

```

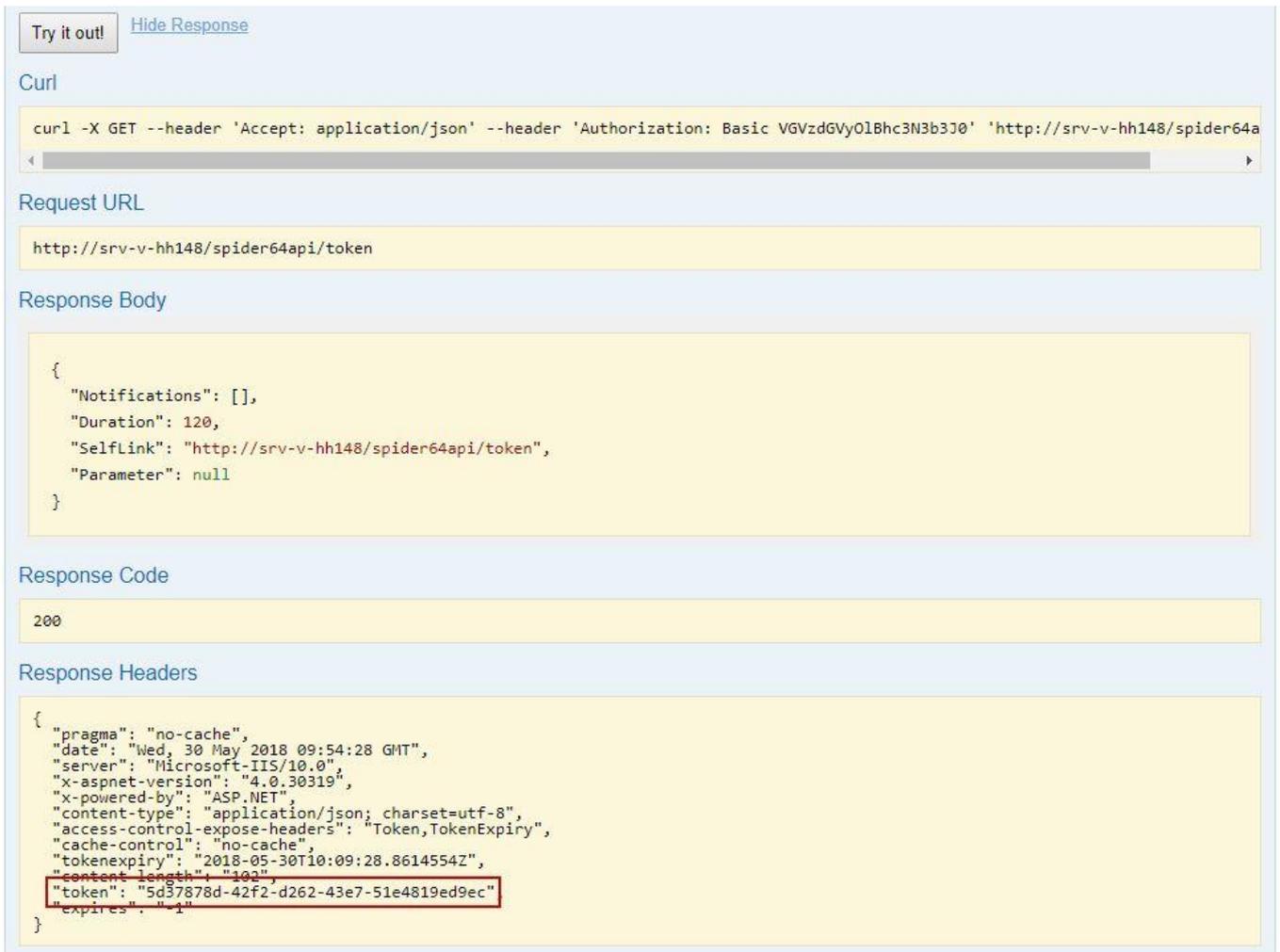
Response Content Type application/json ▼

**Parameters**

Parameter	Value	Description	Parameter Type	Data Type
parameter.cultureCode	<input style="width: 100%;" type="text"/>		query	string
Authorization	<input style="width: 100%;" type="text" value="Basic VGVzdGVyOIBhc3N3b3J0"/>	The basic authorization information, e.g. 'Basic TG9naW46UGFzc3dvcnQ='(The second part is the combination of 'userLogin:password' encoded in Base64)	header	string

Try it out!

Nach der Betätigung der Schaltfläche „Try it out!“ wird die Abfrage bearbeitet und das Ergebnis wird dargestellt. Der generierte Zugriffstoken ist bei erfolgreicher Bearbeitung im Bereich Response Headers aufgeführt.



The screenshot displays a REST client interface with the following sections:

- Try it out!** [Hide Response](#)
- Curl**

```
curl -X GET --header 'Accept: application/json' --header 'Authorization: Basic VGVzdGVyO1Bhc3N3b3J0' 'http://srv-v-hh148/spider64a
```
- Request URL**

```
http://srv-v-hh148/spider64api/token
```
- Response Body**

```
{
  "Notifications": [],
  "Duration": 120,
  "SelfLink": "http://srv-v-hh148/spider64api/token",
  "Parameter": null
}
```
- Response Code**

```
200
```
- Response Headers**

```
{
  "pragma": "no-cache",
  "date": "Wed, 30 May 2018 09:54:28 GMT",
  "server": "Microsoft-IIS/10.0",
  "x-aspnet-version": "4.0.30319",
  "x-powered-by": "ASP.NET",
  "content-type": "application/json; charset=utf-8",
  "access-control-expose-headers": "Token,TokenExpiry",
  "cache-control": "no-cache",
  "tokenexpiry": "2018-05-30T10:09:28.8614554Z",
  "content-length": "102",
  "token": "5d37878d-42f2-d262-43e7-51e4819ed9ec"
  "expires": "-1"
}
```

Der Authorization-Text für andere Abfragen ist in diesem Beispiel bis zum Ablauf des Tokens „Token 5d37878d-42f2-d262-43e7-51e4819ed9ec“.

GET /v1/{mandator}/meta/entities The service returns a collection of available entities.

**Implementation Notes**  
The service returns a collection of available entities.

**Response Class (Status 200)**  
OK

Model | Example Value

```

{
  "entities": [
    {
      "name": "string",
      "displayName": "string",
      "parentName": "string",
      "parentDisplayName": "string"
    }
  ],
  "notifications": [

```

Response Content Type

**Parameters**

Parameter	Value	Description	Parameter Type	Data Type
mandator	<input type="text" value="0"/>	The mandator for which the available entities will be determined.	path	string
parameter.cultureCode	<input type="text"/>		query	string
Authorization	<input type="text" value="Token 5d37878d-42f2-d262-43e7-51e4819ed9ec"/>	The token authorization information, e.g. 'Token 01234567-89AB-CDEF-0123-456789ABCDEF'	header	string

## 4.10 Web API Dokumentation

Die Spider Web API verfügt über eine nicht interaktive Dokumentation. Diese enthält Webseiten mit Informationen zu allen Funktionen, ihren Parametern und Rückgabewerten .

Die Dokumentation ist unter folgender URL erreichbar

[http\(s\)://{ServerName}/{SpiderApiInstance}/help](http(s)://{ServerName}/{SpiderApiInstance}/help)

## Beispiel 1: Hauptseite

# Spider Web API Help Page

## Introduction

Spider WEB API - Custom your own Http Web API CRUD Requests.

## AdminEntity

Mandator Web Interface

API	Description
POST v1/admin/{entityName}	The service creates a new admin entity.
POST admin/{entityName}	The service creates a new admin entity.
GET v1/admin/{entityName}/{entityID}?CultureCode={CultureCode}	The service returns the fields of an admin entity.
GET admin/{entityName}/{entityID}?CultureCode={CultureCode}	The service returns the fields of an admin entity.
PUT v1/admin/{entityName}/{entityID}	The service updates an admin entity.
PUT admin/{entityName}/{entityID}	The service updates an admin entity.
DELETE v1/admin/{entityName}/{entityID}?CultureCode={CultureCode}	The service deletes an admin entity.
DELETE admin/{entityName}/{entityID}?CultureCode={CultureCode}	The service deletes an admin entity.
GET v1/admin/{entityName}?MaxRows={MaxRows}&FirstRow={FirstRow}&OrderBy={OrderBy}&Distinct={Distinct}&AllFields={AllFields}&CultureCode={CultureCode}	The service returns the admin entities.
GET admin/{entityName}?MaxRows={MaxRows}&FirstRow={FirstRow}&OrderBy={OrderBy}&Distinct={Distinct}&AllFields={AllFields}&CultureCode={CultureCode}	The service returns the admin entities.

## Beispiel 2: Detailansicht

[Spider Help](#) [Page](#) [Home](#) [API](#)

# POST v1/{mandator}/entity/{entityName}

The service generates a new entity.

## Request Information

### URI Parameters

Name	Description	Type	Additional information
mandator	The mandator for whom the entity should be created.	string	Required
entityName	The name of the entity to be created.	string	Required

### Body Parameters

A parsed from the URL parameters.

[EntityFieldParameterModel](#)

Name	Description	Type	Additional information
Fields		Dictionary of string [key] and Object [value]	None.
CultureCode		string	None.

## Request Formats

**application/json, text/json**

Sample:

```
{
  "fields": {},
  "cultureCode": "sample string 1"
}
```

### Beispiel 3: Rückgabeobjekt

#### Response Formats

application/json, text/json

Sample:

```
{
  "entities": [],
  "notifications": [],
  "duration": 1,
  "selfLink": "http://webapihelppage2.com",
  "parameter": {
    "cultureCode": "sample string 1"
  }
}
```

## 4.11 Spider Web API Explorer

Der Spider Web API Explorer ist eine Web Anwendung, die zu Test- und Entwicklungszwecken einen erleichterten Zugriff auf alle relevanten Funktionen der Spider Web API ermöglicht.

Spider Web API Explorer ermöglicht Datenoperationen wie Erstellen, Suchen, Ändern und Löschen (CRUD: Create, Read, Update, Delete). Die Ergebnisse können wahlweise in JSON oder in HTML angezeigt werden.

### 4.11.1 Benutzermerkmale

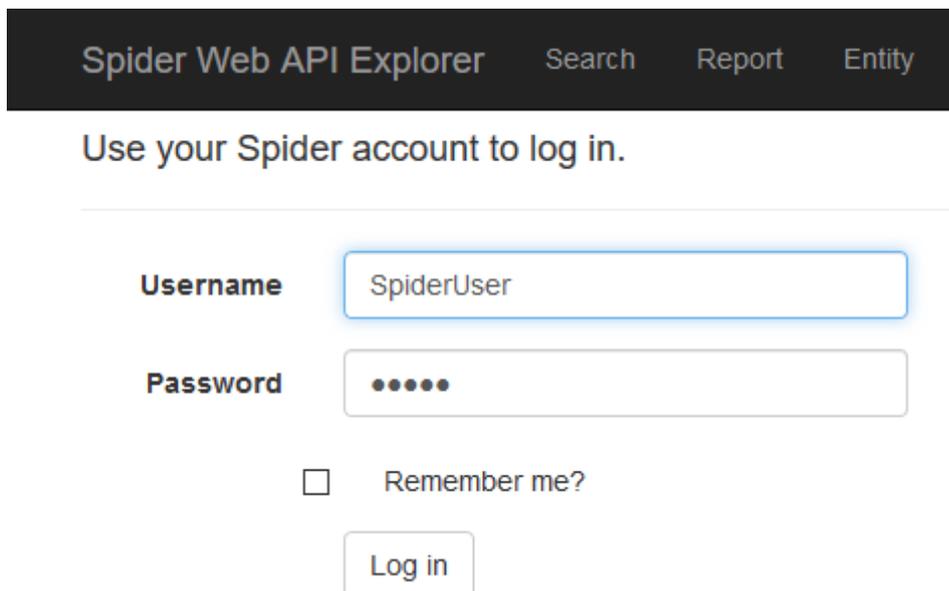
Der erwartete Benutzer der Spider Web API Explorer ist ein Tester, Qualitätsbeauftragter, ein Business Analyst, ein Administrator, oder ein Softwareentwickler der die Funktionalitäten der Spider web API testen möchte. Grundkenntnisse in Spider sind notwendig.

### 4.11.2 Authentifizierung

Die Authentifizierung am Web API Explorer erfolgt mittels eines Spider-Benutzerkontos. Benutzer haben im Web API Explorer und in der Web API die gleichen Berechtigungen.

Der API Explorer ist unter der folgenden URL erreichbar:

```
http(s)://{ServerName}/{SpiderApiExplorerInstance}
```



Spider Web API Explorer Search Report Entity

Use your Spider account to log in.

Username SpiderUser

Password ●●●●●

Remember me?

Log in

### 4.11.3 Sitzungsparemeter (Mandant, Sprache, Version)

Nach der Anmeldung werden die Sitzungsparemeter festgelegt. Hier muss ein Mandant ausgewählt werden und gleichzeitig können die Sprache und die API Version ausgewählt werden. Standardmäßig ist die letzte Version der API vorausgewählt. Wenn keine Sprache ausgewählt wird, gilt die API-Standardsprache Englisch.

Die Sitzungsparemeter können jederzeit geändert werden

### 4.11.4 Kommunikation mit der Spider Web API

Die Kommunikation mit der Spider Web API über das HTTP / HTTPS Protokoll mit einem Token Mechanismus gesichert. der Token wird in den Browser Cache gespeichert.

#### Response

Die Antwortseite der Der Spider API Explorer besteht aus 2 Tabs, HTML und JSON, hier können die Ergebnisse wahlweise in HTML oder in JSON angezeigt werden

#### Responsemodell

Folgende Felder werden zurückgegeben:

Feld	Beschreibung
Duration	Dauer der Anfrage ist Sekunden.
Selflink	Liefert ein URL der Abfrage zurück.
Notification	Liefert die Fehlermeldungen aus dem Kernel. (Category und Message)
CultureCode	Liefert die verwendete Sprache in der Abfrage zurück.

Die restlichen Felder sind von der Abfrage abhängig.

HTTP Statuscode: Der Spider Web API Explorer bekommt immer HTTP Statuscode 200 als Antwort aus der Spider Web API. Fehlermeldungen werden immer in dem Rückgabeobjekt verpackt. Unberechtigte Aufrufe geben den HTTP Statuscode 401 zurück.

#### Beispiel:

Die Antwort auf eine Asset-Suche

## 1. HTML Format:

## Response

**Duration** 1520 :ms  
**SelfLink** http://localhost/SpiderWebApi/100/search/Asset?MaxRows=10&Firstrow=1&Orderby=AssetNo  
**Parameter**  
**OrderBy** AssetNo  
**MaxRows** 10  
**FirstRow** 1  
**AllFields**   
**Distinct**   
**Count** 2205  
**IsResultTruncated**   
**Fields**  
**Conditions**

HTML

JSON

## Result

id	assetNo	hostname	assetType	status	functionUnitID	functionUnit
2	Access card-0002		Zugriffskarte	Aktiv		
3	Access card-0003		Zugriffskarte	Aktiv		
4	Access card-0004		Zugriffskarte	Aktiv		
5	Access card-0005		Zugriffskarte	Aktiv		
6	Access card-0006		Zugriffskarte	Aktiv		
7	Access card-0007		Zugriffskarte	Aktiv		
8	Access card-0008		Zugriffskarte	Aktiv		
9	Access card-0009		Zugriffskarte	Aktiv		
10	Access card-0010		Zugriffskarte	Aktiv		
11	Access card-0011		Zugriffskarte	Aktiv		

## 2. JSON Format

[HTML](#)[JSON](#)

### Response in formatted Json

```
{
  "Count": 2205,
  "Result": [
    {
      "id": 2,
      "assetNo": "Access card-0002",
      "hostname": null,
      "assetType": "Zugriffskarte",
      "status": "Aktiv",
      "functionUnitID": null,
      "functionUnit": null
    }
  ],
  "IsResultTruncated": false,
  "Notifications": [],
  "Duration": 469,
  "SelfLink": "http://localhost/SpiderWebApi/100/search/Asset?MaxRows=1&Firstrow=1",
  "Parameter": {
    "Conditions": [],
    "MaxRows": 1,
    "FirstRow": 1,
    "OrderBy": "",
    "Distinct": false,
    "Fields": null,
    "AllFields": false,
    "CultureCode": null
  }
}
```

### 4.11.5 Web API Explorer Bereiche

Es sind drei Bereiche vorhanden:

- Search API
- Report API

- Entity API

Spider Web API Explorer Search Report Entity

## Search API

---

Im Bereich „Search API“ kann nach Spider Entitäten gesucht werden. [Search Bereich](#) (siehe Seite 110)

Die Suche ist unter folgender URL erreichbar:

```
http(s)://{ServerName}/{SpiderApiExplorerInstance}/search
```

# Search API

Enter your API Request

<b>Entities</b>	Asset
<b>Fields</b>	AccountingApproachID int Actual book value € money Actual book value money Archived bit
<b>OrderBy</b>	AssetNo nvarchar
<b>MaxRows</b>	1
<b>FirstRow</b>	1
<b>Distinct</b>	<input type="checkbox"/>
<b>AllFields</b>	<input type="checkbox"/>
<b>Conditions</b>	
<b>Example</b>	OR NOT assetno like %25

**Send Request**

## Response

<b>Duration</b>	324 :ms
<b>SelfLink</b>	http://localhost/SpiderWebApi/100/search/Asset?MaxRows=1&Firstrow=1
<b>Parameter</b>	
<b>OrderBy</b>	
<b>MaxRows</b>	1
<b>FirstRow</b>	1
<b>AllFields</b>	<input type="checkbox"/>
<b>Distinct</b>	<input type="checkbox"/>
<b>Count</b>	2205
<b>IsResultTruncated</b>	<input type="checkbox"/>
<b>Fields</b>	
<b>Conditions</b>	

HTML

JSON

## Response in formatted Json

```

{
  "Count": 2205,
  "Result": [
    {
      "id": 2,
      "assetNo": "Access card-0002",
      "hostname": null,
      "assetType": "Zugriffskarte",
      "status": "Aktiv",
      "functionUnitID": null,
      "functionUnit": null
    }
  ],
  "IsResultTruncated": false,
  "Notifications": [],
  "Duration": 324,
  "SelfLink": "http://localhost/SpiderWebApi/100/search/Asset?MaxRows=1&Firstrow=1",
  "Parameter": {
    "Conditions": [],
    "MaxRows": 1,
    "FirstRow": 1,
    "OrderBy": "",
    "Distinct": false,
    "Fields": null,
    "AllFields": false,
    "CultureCode": null
  }
}

```

## Parameter

Die Parameter der Suche werden im Web-Frontend eingegeben, für genauere Informationen siehe Kapitel [Parametrisierung](#) (siehe Seite 111).

## Report API

---

Der Bereich „Report API“ ermöglicht die Suche nach verfügbaren Berichten für den ausgewählten Mandanten.

Die Report API ist unter folgende URL erreichbar:

http(s)://{ServerName}/{SpiderApiExplorerInstance}/report

Spider Web API Explorer   Search   **Report**   Entity   Change Session   Help

# Report API

Enter your API Request

<b>Report</b>	Please select a Report
<b>Fields</b>	Autodesk - Assets with Autodesk Programs - 45 ( Spider Asset ) Autodesk - Qualified Autodesk Inventory - 46 ( Spider Asset )
<b>OrderBy</b>	Clearing - Double translations - 106 ( Spider Localization )
<b>MaxRows</b>	Compliance - Product overlicensing - 73 ( Spider Licence )
<b>FirstRow</b>	Compliance - Product underlicensing - 74 ( Spider Licence )
<b>Distinct</b>	Compliance - Productversion compliance - 72 ( Spider Licence ) Compliance - Productversion compliance per Legal Entity - 71 ( Spider Licence ) Compliance - Required Licenses per Device - 75 ( Spider Licence )
<b>AllFields</b>	Contracts - Contract overview - 97 ( Spider Contract ) Contracts - Contracts according to contractual parties - 94 ( Spider Contract )
<b>Conditions</b>	Contracts - Contracts with missing dates - 95 ( Spider Contract )
<b>Example</b>	Contracts - Contracts with missing responsibilities - 96 ( Spider Contract ) Contracts - Master agreements - 98 ( Spider Contract ) Contracts - Overview of Contractuel Parties - 93 ( Spider Contract ) Contracts - Remaining time to maturity - 99 ( Spider Contract )
<b>Send Request</b>	

## Response

**Duration** 3675 :ms  
**SelfLink** http://localhost/SpiderWebApi/100/report/10?MaxRows=10&Firstrow=1&Orderby=Asset  
**Parameter**  
**OrderBy** Asset  
**MaxRows** 10  
**FirstRow** 1  
**AllFields**   
**Distinct**   
**Count** 1305  
**IsResultTruncated**   
**Fields**  
**Conditions**

HTML

JSON

## Meta

**Name** Assets with outdated scan date  
**ApplicationName** Spider Asset  
**Description** Assets with outdated scan date (older than 90 days)  
**Guid** 3428e63a-082b-3c9d-d318-04afa7dccfbf  
**ID** 10  
**Category** Inventory

## Result

assetID	asset	hostName	assetStatus	invLastScanDate	swrUpdateDate
1902	Azure client-0002		Aktiv		
1903	Azure client-0003		Aktiv		
1904	Azure client-0004		Aktiv		
1905	Azure client-0005		Aktiv		
1906	Azure client-0006		Aktiv		
1907	Azure client-0007		Aktiv		
1908	Azure client-0008		Aktiv		
1909	Azure client-0009		Aktiv		
1910	Azure client-0010		Aktiv		
1911	Azure client-0011		Aktiv		

## Parameter

Die Parameter der Report API werden analog zu der Search API im Web-Frontend eingegeben. Für genauere Informationen siehe Kapitel [Parametrisierung](#) (siehe Seite 113)

## Entity API

Der Entity Bereich ermöglicht Datenoperationen (CRUD), wie Erstellen, Lesen, Ändern und Löschen von Entitäten. Für genauere Informationen siehe Kapitel [Entity Bereich](#) (siehe Seite 108)

Die Entity API ist unter folgende URL erreichbar

```
http(s)://{ServerName}/{SpiderApiExplorerInstance}/entity
```

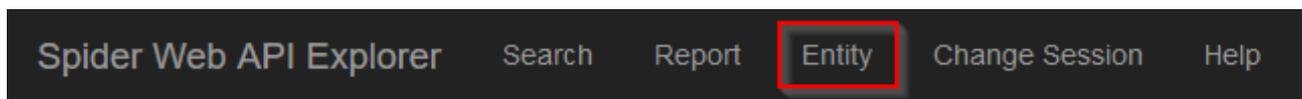
Die Entity API besteht aus sechs Bereichen:

### Get Entity

In diesem Bereich können Entitäten abgerufen werden.

**Parameter:** Entity Name, und die ID des Entitys.

**Beispiel:** Entity Asset mit ID=1 abrufen und das Ergebnis in JSON zeigen.



## Entity API

Enter your API Request

Entity

**Get** Create Update Delete Types TypeValues

Entities

EntityID

Send Request

## Response

**Duration** 642 :ms

**SelfLink** <http://localhost/SpiderWebApi/100/entity/Asset/1>

HTML

JSON

## Response in formatted Json

```
{
  "Fields": {
    "type": "Access card",
    "id": 1,
    "scObjectID": 29,
    "mandatorID": 100,
    "legalEntityID": 1,
    "assetNo": "Access card-0001",
    "uuid": null,
    "domain": null,
    "netBiosDomain": null,
    "hostName": null,
    "name": null,
    "assetStatusID": 1,
    "commentStatus": null,
    "serialNo": null,
    "modelName": null,
    "inventoryNo": null,
    "comment": "Test - SCObjectID 29 - Access card - 001",
    "assetStatus": "Active"
  }
}
```

## Create Entity

In diesem Bereich können Entitäten erstellt werden.

**Parameter:** Entity Name, Entity Fields als JSON.

**Beispiel:** Entity Desktop Erstellen:

1. Entity Desktop auswählen.
2. Nutzdaten in JSON formatiert eingeben:

```
{ "fields":  
  { "Mandator":0,  
    "AssetNo": "123456789",  
    "HostName": "MyHostname",  
    "AssetStatusID":1,  
    "CommentStatus": "Just want to try to create a desktop",  
    "SerialNo": "123649875547945789",  
  }  
}
```

3. Schaltfläche <Send Request> klicken.

The screenshot shows the Spider Web API Explorer interface. At the top, there is a navigation bar with 'Spider Web API Explorer', 'Search', 'Report', and 'Entity' (highlighted with a red box). Below this is the 'Entity API' section. A text input field contains 'Entity'. Below the input field are buttons for 'Get', 'Create' (highlighted with a red box), 'Update', 'Delete', 'Types', and 'TypeValues'. Under the 'Entities' section, a dropdown menu is open, showing 'Desktop' selected. Below the dropdown, a JSON request body is displayed in a text area:

```
{  
  "fields":  
  {  
    "Mandator":1,  
    "AssetNo":"123456",  
    "HostName":"My HostName",  
    "Name":"Desktop Created by  
    Web API",  
    "AssetStatusID":1,  
  }  
}
```

At the bottom of the interface, there is a blue 'Send Request' button.

## Result

Entity created

**Duration** 4694 :ms  
**SelfLink** http://localhost/SpiderWebApi/100/entity/Desktop  
**ID** 2886

HTML

JSON

### Fields

Key	Value
mandator	1
assetNo	123456
hostName	My HostName
name	Desktop Created by Web API
assetStatusID	1

## Update Entity

In diesem Bereich können Entitäten geändert werden.

Parameter: Entity Name, Entity Fields als JSON und die ID der Entity.

Beispiel: Hostname der Entity Desktop mit der ID=1 ändern

1. Entity Asset auswählen.
2. Nutzdaten in JSON formatiert eingeben:

```
{"fields": {  
  "Hostname": "The hostname of My Test Desktop has changed"  
}}
```

3. Die ID der Entity eingeben.

4. Schaltfläche <Send Request> klicken.

The screenshot shows the Spider Web API Explorer interface. At the top, there is a navigation bar with 'Spider Web API Explorer', 'Search', 'Report', and 'Entity' (highlighted with a red box). Below this is the 'Entity API' section. A form titled 'Enter your API Request' contains several fields: 'Entity' (with 'Desktop' selected), 'EntityID' (with '2886' entered), and 'JSON' (with a JSON object: `{"fields": {"Hostname": "The hostname of My Test Desktop has changed"}}`). The 'Update' button is highlighted with a red box. A 'Send Request' button is located at the bottom of the form.

### Result

**Duration** 619 :ms  
**SelfLink** <http://localhost/SpiderWebApi/100/entity/Desktop/2886>

HTML	JSON
<b>Fields</b>	
<b>Key</b>	<b>Value</b>
hostname	The hostname of My Test Desktop has changed

## Delete Entity

In diesem Bereich können Entitäten gelöscht werden.

**Parameter:** Entity Name und die ID der Entity.

**Beispiel:** Entity Desktop mit der ID=1 löschen

1. Entity Asset auswählen.
2. Die ID des Entitäts eingeben.
3. Schaltfläche <Send Request> klicken.

Spider Web API Explorer Search Report Entity

## Entity API

Enter your API Request

Entity

Get Create Update Delete Types TypeValues

Entities Desktop

EntityID 286

Send Request

## EntityTypes Bereich

Dieser Bereich ist Bestandteil der Entity Metadaten, siehe Kapitel [Entity Bereich](#) (siehe Seite 108). EntityTypes beschreiben vordefinierte Zustände und Werte, die einzelne Felder von Entitäten annehmen können.

**Parameter:** Entity Name und EntityType.

**Beispiel:** Vorhandene EntityTypes für Asset Anzeigen:

1. Entity Asset auswählen.

- Schaltfläche <Send Request> klicken.

Spider Web API Explorer Search Report **Entity**

## Entity API

Enter your API Requeste

Entity

Get Create Update Delete **Types** TypeValues

**Entities** Asset

**Types** AssetStatusID - AssetStatusID + (t)

Send Request

### Response

**Duration** 6 :ms  
**SelfLink** http://localhost/SpiderWebApi/100/meta/entity/Asset/entityType

HTML JSON

EntityTypes	Name	ColumnName	KeyColumn	ValueColumn	MandatorFilter	SObjectFilter	Translate
	AssetStatusID	AssetStatusID	ID	Name	False	False	True
	CurrencyCode	CurrencyCode	Code	EntityTypeValue	False	False	False
	MandatorID	MandatorID	ID	Name	False	False	False
	PVD_FinancingType	FinancingType	Value	Text	False	False	True
	PVD_ImportSource	ImportSourceID	Value	Text	False	False	False
	PVD_OsType	OsTypeID	Value	Text	False	False	True

### EntityTypeValues Bereich

EntityTypeValues sind die ausdefinierten Zustände und Werte die einzelne Felder von Entitäten annehmen können.

**Parameter:** Entity Name und EntityType.

**Beispiel:** Vorhandene EntityValues für EntityType AssetStatusID Anzeigen:

- Entity Asset auswählen.

2. Entity Type AssetStatusID auswählen.
3. Schaltfläche <Send Request> klicken.

Response

Duration 46 :ms  
 SelfLink http://localhost/SpiderWebApi/100/meta/entity/Asset/entityType/AssetStatusID

HTML JSON

EntityTypes							
	Name	ColumnName	KeyColumn	ValueColumn	MandatorFilter	SCObjectFilter	Translate
	AssetStatusID	AssetStatusID	ID	Name	False	False	True

EntityValues							
id	name	initial	unknown	active	archive	defaultArchive	description
1	Active	True	False	True	False	False	1. Asset is in operation
2	Disposed	False	False	False	True	True	6. Asset was irretrievably sold
3	In repair	False	False	False	False	False	4. Asset is beeing repaired
4	In stock	False	False	True	False	False	3. Asset is in stock
5	Inactive	False	False	False	False	False	2. Asset is not in operation
6	Stolen	False	False	False	True	False	5. Asset was irretrievably stolen
7	ActiveDefined	False	False	True	False	False	7. Asset is in operation (manual handling)
8	NotReported	False	False	False	False	False	8. Asset is not active (set by inventory tool data)

## 4.12 Code-Beispiele

Im Folgenden wird anhand einiger Code-Beispiele die Nutzung der verschiedenen Funktionsbereiche der Spider Web API demonstriert. Die Beispiele sind in C# geschrieben und verwenden die Drittanbieter-Bibliothek Restsharp als Rest-Client.

Die Code Beispiele können auch über folgenden Community Artikel eingesehen werden:

<https://community.flexera.com/t5/Spider-Knowledge-Base/Spider-Web-API-code-examples/ta-p/220723>

### 4.12.1 Authentifizierung

Zur Authentifizierung gegenüber der Spider Web API wird ein Token benötigt, welcher in jeder Anforderung enthalten sein muss.

Der Authentifizierungstoken kann über eine GET-Anforderung mittels Basisauthentifizierung erhalten werden. Der Token wird dann im Header der Server-Antwort zurückgegeben.

#### Beispiel: GetToken

```
private string GetUserTokenExample() {
    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";
    // Spider Username
    var username = "YourSpiderUsername";
    //Spider Password
    var password = "YourSpiderPassword";
    // Create a BasicAuthenticator RestClient with your API Url, user and Password
    // The rest Client is an restsharp Client in the example

    RestClient client = new RestClient
    {
        BaseUrl = new Uri(yourApiUrl),
        Authenticator = new HttpBasicAuthenticator(username, password)
    };

    // Create a Rest request and pass the API resource Name 'token'

    RestRequest request = new RestRequest {Resource = "token"};

    // Optional, trust the security certificate
    ServicePointManager.ServerCertificateValidationCallback +=
        (sender, certificate, chain, sslPolicyErrors) => true;

    // Send request to the Web API

    IRestResponse response = client.Execute<HttpResponseMessage>(request);

    // Optional: Exception handling
    if (response.ErrorException != null){
        const string message = "Error retrieving response. Check inner details for more info.";
        var yourException = new ApplicationException(message, response.ErrorException);
        throw yourException;
    }

    // Get the token from the headers of the response
```

```

string yourSpiderAPIToken = "";
if (response?.Headers == null) return "";
// The token is located in a http Header with name 'Token'

foreach (Parameter header in response.Headers.ToList())
    switch (header?.Name)
    {
        case "Token":
if (header.Value != null) yourSpiderAPIToken = header.Value.ToString();
            break;
            default: continue;
    }

//Return the Token
return yourSpiderAPIToken;
}

```

## 4.12.2 Entity Bereich

Der Entity-Bereich ermöglicht Datenoperationen (CRUD) wie das Erstellen, Lesen, Ändern und Löschen von Entitäten.

### Beispiel: Entity lesen

```

private void GetEntityExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/entity/asset/{entityid}";
    // example 0/entity/Asset/1

    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // the token you have from the first Token example, and you will use this token in every request

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize your Rest Request

    RestRequest request = new RestRequest
    {
        Resource = url,
        RequestFormat = DataFormat.Json
    };
    //Format Header
    request.AddHeader("Accept", "application/json, text/json");
    //Authentication add your token as an Authentication to your request object.
    request.AddHeader("Authorization", "Token " + token);

    //Optional, Trust the security certificate

```

```
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);
// the Response should look like the following
//   "fields": {
//     "type": "Laptop",
//     "id": 1,
//     "scObjectID": 35,
//     "mandatorID": 0,
//     "legalEntityID": 0,
//     "assetNo": "LAPTOP-BRBR002.domain.corp",
//     "uuid": "4c4c4544-0037-5110-805a-b5c04f315131",
//     "domain": "domain.corp",
//     "netBiosDomain": "DOMAIN",
//     "hostName": "LAPTOP-BRBR002",
//     "name": null,
//     "assetStatusID": 5,
//     "commentStatus": null,
//     "serialNo": "57QZ1Q1",
//     "modelName": "Latitude E6320",
//     "inventoryNo": null,
//     "comment": "Geändert!",
//     "deliveryNoteID": null,
//     "functionUnitID": 1,
//     "accountingApproachID": null,
//     "serviceLevelID": null,
//     "locationID": null,
//     "purchaseDate": "2018-07-01T00:00:00",
//     "businessValue": 2000.0000,
//     "businessValueEUR": 2000.0000,
//     "costCentreID": null,
//     "commentCostCentre": null,
//     "contract": null,
//     "contractCommercial": null,
//     "contractService": null,
//     "purchaseCostCentreID": null,
//     "bookMainNumber": null,
//     "bookSecondNumber": null,
//     "actualBookValue": null,
//     "actualBookValueEUR": null,
//     "yearDepreciation": null,
//     "yearDepreciationEUR": null,
//     "currencyCode": "EUR",
//     "invLastScanDate": "2018-05-29T06:47:19",
//     "invMessage": null,
//     "invExists": true,
//     "swrDeviceID": 1,
//     "sysInsertUser": "Spider-System",
//     "sysUpdateUser": "tf",
//     "sysInsertDate": "2018-05-31T14:27:04.12",
//     "sysUpdateDate": "2018-06-21T13:34:35.387",
//     "sysTemplate": false,
//     "sysTemplateActive": false,
//     "sysTemplateID": null,
```

```
// "sysArchived": false,
// "sysTypeChangeId": null,
// "swrUpdateDate": "2018-05-31T14:58:39.307",
// "cpuLogicalCount": 4,
// "graphicMemoryMB": null,
// "cpuCount": 1,
// "cpuSpeedInMhz": 2601,
// "corePerCPU": 2,
// "responsibleEmployeeID": 116,
// "diskTotalInGb": 466,
// "importSourceID": null,
// "osTypeID": null,
// "ramInMb": 8073,
// "socketCount": null,
// "orderDate": "2018-06-01T00:00:00",
// "warrantyExpires": "2020-05-31T00:00:00",
// "invoiceDate": "2018-06-04T00:00:00",
// "installDate": null,
// "manufacturer": "Dell Inc.",
// "invScanSystem": "Generic - ColumbusInventoryAgent.exe 7.5.4.22",
// "cpuArchitecture": "x64",
// "graphicAdapter": null,
// "testksy": null,
// "supplier": null,
// "cpuType": "Intel(R) Core(TM) i5-2540M CPU @ 2.60GHz",
// "mlTestETField1": "1",
// "financingType": "Investment",
// "invoiceNumber": "abcde-1",
// "orderPosition": null,
// "orderNumber": null,
// "serviceTag": null,
// "qualifiedDesktop": false,
// "imei": null,
// "phoneNumber": null,
// "cusT_MLTest_NEU": null,
// "displaySize": null,
// "applicationGUID": "38f96e97-c314-44b8-b137-c966d35a3ba0"
// },
// "notifications": [],
// "duration": 790,
// "selfLink": "http://yourserver/SpiderAPI/0/entity/Asset/1",
// "parameter": {
//   "cultureCode": null
// }
//}

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}
```

## Beispiel: Entity erstellen

```
private void CreateEntityExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/entity/{entity}/";
    // example 0/entity/tablet
    // json to create the asset
    //{
    //  "fields":
    //  {
    //    "hostname":"Test Hostname",
    //    "Mandator":0,
    //    "assetNo": "Test no",
    //    "name": "Test Asset name",
    //    "assetStatusID": 1,
    //    "commentStatus": "comment test",
    //    "serialNo": "123456",
    //    "modelName": "Latitude E6320",
    //  }
    //}
    // Escape the " with \ in your create entity json

    var jsonToSend =
        "{ \"fields\":{\r\n {\"hostname\": \"Test Hostname\", \r\n \"Mandator\":0, \r\n \"assetNo\": \"Test no\", \r\n \"name\": \"Test Asset name\", \r\n \"assetStatusID\": 1, \r\n \"commentStatus\": \"comment test\", \r\n \"serialNo\": \"123456\", \r\n \"modelName\": \"Latitude E6320\", \r\n}\r\n}";
    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // the token you have from the first Token example, and you will use this token in every request

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize your Rest Request set the http post method

    RestRequest request = new RestRequest
    {
        Resource = url,
        RequestFormat = DataFormat.Json,
        Method = Method.POST
    };
    //Format Header
    request.AddHeader("Accept", "application/json, text/json");
    //Authentication add your token as an Authentication to your request object.
    request.AddHeader("Authorization", "Token " + token);

    //Optional, Trust the security certificate
    //ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

    //Set the body of the request
    if (!string.IsNullOrEmpty(jsonToSend))
    {
        request.AddParameter("application/json; charset=utf-8", jsonToSend, ParameterType.RequestBody);
    }
}
```

```

    }

    //Send the Request
    IRestResponse response = client.Execute(request);

    // the result should look like the following

    //{
    //  "ID": 42154,
    //  "Notifications": [
    //    {
    //      "Category": 2,
    //      "Message": "Column 'Mandator' could not be found."
    //    }
    //  ],
    //  "Duration": 1080,
    //  "SelfLink": "http://yourWebServer/SpiderAPI/0/entity/Tablet",
    //  "Parameter": {
    //    "Fields": {
    //      "hostname": "Test Hostname",
    //      "mandator": 0,
    //      "assetNo": "Test no",
    //      "name": "Test asset name",
    //      "assetStatusID": 1,
    //      "commentStatus": "comment test",
    //      "serialNo": "123456",
    //      "modelName": "Latitude E6320"
    //    },
    //    "CultureCode": null
    //  }
    //}

    // Optional: Exception handling
    if (response.ErrorException != null)
    {
        const string message = "Error retrieving response. Check inner details for more info.";
        var yourException = new ApplicationException(message, response.ErrorException);
        throw yourException;
    }
}

```

## Beispiel: Entity ändern

```

private void UpdateEntityExample(string token)
{
    // The url of mandator you from the Web API help page
    var url = "{mandatorid}/entity/{entity}";
    // example 0/entity/tablet

    // Escape the " with \ in your update entity json

    var jsonToSend = "{\field:\":{\hostname\":" New Hostname\"}";

    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";
}

```

```

// the token you have from the first Token example, and you will use this token in every request

// Create a Rest Client

RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

// Initialize your Rest Request set the http post method

RestRequest request = new RestRequest
{
    Resource = url,
    RequestFormat = DataFormat.Json,
    Method = Method.PUT
};
//Format Header
request.AddHeader("Accept", "application/json, text/json");
//Authentication add your token as an Authentication to your request object.
request.AddHeader("Authorization", "Token " + token);

//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Set the body of the request
if (!string.IsNullOrEmpty(jsonToSend))
{
    request.AddParameter("application/json; charset=utf-8", jsonToSend, ParameterType.RequestBody);
}

//Send the Request
IRestResponse response = client.Execute(request);

// the result should look like the following

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}

```

## Beispiel: Entity löschen

```

private void DeleteEntityExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/entity/asset/{entityid}";
    // example 0/entity/Asset/1

    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // Create a Rest Client

```

```

RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

// Initialize your Rest Request

RestRequest request = new RestRequest
{
    Resource = url,
    RequestFormat = DataFormat.Json,
    Method = Method.DELETE
};
//Format Header
request.AddHeader("Accept", "application/json, text/json");
//Authentication add your token as an Authentication to your request object.
request.AddHeader("Authorization", "Token " + token);

//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}

```

### 4.12.3 Search Bereich

Die Suchfunktionen dienen zum Suchen, Filtern und Finden von Spider Entitäten .

#### Beispiel:Search

```

private void SearchExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/search/Asset?Orderby=AssetNo&MaxRows=10&Firstrow=1";
    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize your Rest Request

    RestRequest request = new RestRequest
    {
        Resource = url,

```

```
    RequestFormat = DataFormat.Json
};
//Format Header
request.AddHeader("Accept", "application/json, text/json");

//Authentication add your token as an Authentication to your request object.
request.AddHeader("Authorization", "Token " + token);

//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}
}

// the response content should look like the following
// "Result": [
// {
//     "id": 11,
//     "assetNo": "ESXHOST003.domain.CORP",
//     "hostname": "ESXHOST003",
//     "assetType": "Server",
//     "status": "Aktiv",
//     "functionUnitID": 11,
//     "functionUnit": "ESXHOST003.domain.CORP",
//     "serialNo": null
// },
// {
//     "id": 12,
//     "assetNo": "ESXHOST004.domain.CORP",
//     "hostname": "ESXHOST004",
//     "assetType": "Server",
//     "status": "Aktiv",
//     "functionUnitID": 12,
//     "functionUnit": "ESXHOST004.domain.CORP",
//     "serialNo": null
// },
// {
//     "id": 10,
//     "assetNo": "ESXHOST005.domain.CORP",
//     "hostname": "ESXHOST005",
//     "assetType": "Server",
//     "status": "Aktiv",
//     "functionUnitID": 10,
//     "functionUnit": "ESXHOST005.domain.CORP",
//     "serialNo": null
// },
// }
```

```
// }
// ],
// "IsResultTruncated": true,
// "Notifications": [],
// "Duration": 720,
// "SelfLink": "http://yourserver/SpiderAPI/0/search/Asset?MaxRows=10&Firstrow=1&Orderby=AssetNo",
// "Parameter": {
//   "Conditions": [],
//   "MaxRows": 10,
//   "FirstRow": 1,
//   "OrderBy": "AssetNo",
//   "Distinct": false,
//   "Fields": null,
//   "AllFields": false,
//   "CultureCode": null
// }
// }
// }
```

## 4.12.4 Report Bereich

Der Report-Bereich ermöglicht das Generieren, Filtern und Abrufen von in Spider hinterlegten Berichten.

### Beispiel: Report

```
private void GetReportsExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/report/{reportid}";
    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // the token you have from the first Token example, and you will use this token in every request

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize your Rest Request

    RestRequest request = new RestRequest
    {
        Resource = url,
        RequestFormat = DataFormat.Json
    };
    //Format Header
    request.AddHeader("Accept", "application/json, text/json");
    //Authentication add your token as an Authentication to your request object.
    request.AddHeader("Authorization", "Token " + token);

    //Optional, Trust the security certificate
    //ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;
```

```

//Send the Request
IRestResponse response = client.Execute(request);
// the Response should look like the following
//{{
//  "Meta": {
//    "DisplayCategory": "Compliance",
//    "DisplayName": "Productversion compliance per Legal Entity",
//    "DisplayDescription": "Shows Compliance overview per Productversion and Legal Entity",
//    "Guid": "b35ab679-55a5-4219-a9fe-230e7934f68d",
//    "Category": "Compliance",
//    "Name": "Productversion compliance per Legal Entity",
//    "Description": "Shows Compliance overview per Legal Entity",
//    "ID": 52,
//    "ApplicationName": "Spider Licence",
//    "ApplicationGuid": "135ea316-a932-44c9-bffb-fbd128754884"
//  },
//  "Count": 92,
//  "Result": [
//  {
//    "manufacturer": "Microsoft",
//    "productVersionID": 19,
//    "productVersion": "Access 2016 (Device, Win)",
//    "legalEntity": null,
//    "complianceICON": "PROBLEM",
//    "licencePosition": -1,
//    "licenceRequirement": 1,
//    "licenceEntitlement": 0,
//    "downgradeAllocation": 0,
//    "poolingAllocation": 0,
//    "consumed": 0,
//    "remainingLicencesShort": 0
//  },
//  {
//    "manufacturer": "Microsoft",
//    "productVersionID": 10445,
//    "productVersion": "Access 2019 (Device, Win)",
//    "legalEntity": "Zentrale/Software Entwicklung/Administration 3A",
//    "complianceICON": "OK",
//    "licencePosition": 0,
//    "licenceRequirement": 0,
//    "licenceEntitlement": 5,
//    "downgradeAllocation": -5,
//    "poolingAllocation": 0,
//    "consumed": 0,
//    "remainingLicencesShort": 0
//  },
//  ],
//  "IsResultTruncated": false,
//  "Notifications": [],
//  "Duration": 527,
//  "SelfLink": "http://yourserver/Spiderapi/mandatorid/report/52?MaxRows=10&Firstrow=1&Orderby=Consumed",
//  "Parameter": {
//    "Conditions": [],
//    "MaxRows": 10,

```

```

//      "FirstRow": 1,
//      "OrderBy": "Consumed",
//      "Distinct": false,
//      "Fields": null,
//      "AllFields": false,
//      "CultureCode": null
//    }
//  }
//}

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}
}

```

## 4.12.5 Meta Bereich

Im Meta-Bereich der Spider Web-API finden Sie Informationen zu den Feldern in Suchergebnissen und Berichten sowie Informationen zum Erstellen und Bearbeiten von Entitäten.

Zusätzlich zu den Listen der verfügbaren Mandanten, Entitäten, Suchen und Berichte enthalten diese Metadaten je nach Funktion weitere Informationen.

### Beispiel: GetMandator

```

private MandatorObject GetMandatorsExample(string token)
{
    // The url of mandator you from the Web API Help
    var url = "meta/mandator";

    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // The token you have from the first token example, and you will use this token in each request

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize Rest Request
    RestRequest request = new RestRequest
    {
        Resource = url,
        RequestFormat = DataFormat.Json
    };

    //Format the header
    request.AddHeader("Accept", "application/json, text/json");

    //Authentication: Add your token as an Authentication header to your request object.
    request.AddHeader("Authorization", "Token " + token);
}

```

```

//Optional, trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);

// Optional: Exception handling

if (response.ErrorException != null {
const string message = "Error retrieving response. Check inner details for more info.";
var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}

//you can finally deserialize the result in your own object
// As an example the see below the MandatorObject definition
MandatorObject result = new MandatorObject();
if (response.StatusCode == HttpStatusCode.OK)
    {
        result = JsonConvert.DeserializeObject<MandatorObject>(response?.Content);

        return result;
    }

public class MandatorObject
    {
        public System.Data.DataTable Result { get; set; }
        public string CultureCode { get; set; }
    }

```

## Beispiel: GetMetaSearchAsset

```

private void GetMetaSearchAssetExample(string token)
    {
        // The url of mandator you from the web api help page
        var url = "{mandatorid}/meta/search/asset";
        // The Url of the Spider Web API
        var yourApiUrl = "http://yourWebServer/SpiderWebApi";

        // Create a Rest Client

        RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

        // Initialize your Rest Request

        RestRequest request = new RestRequest
        {
            Resource = url,
            RequestFormat = DataFormat.Json
        };
        //Format the header
        request.AddHeader("Accept", "application/json, text/json");
    }

```

```

//Authentication: Add your token as an Authentication header to your request object.
request.AddHeader("Authorization", "Token " + token);

//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);
//the content should look like the following

//{
//  "fields": [
//    {
//      "inResult": false,
//      "fieldName": "ID",
//      "displayName": "ID",
//      "dataType": "int"
//    },
//    {
//      "inResult": true,
//      "fieldName": "AssetNo",
//      "displayName": "AssetNo",
//      "dataType": "nvarchar"
//    },
//  ],
//  "notifications": [],
//  "duration": 55,
//  "selfLink": "http://yourServer/SpiderWebApi/0/meta/search/Asset",
//  "parameter": {
//    "cultureCode": null
//  }
//}

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}

```

## Beispiel: GetMetaReport

```

private void GetMetaReportExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/meta/report/{reporid}";

    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // the token you have from the first Token example, and you will use this token in every request

```

```
// Create a Rest client

RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

// Initialize your Rest Request

RestRequest request = new RestRequest
{
    Resource = url,
    RequestFormat = DataFormat.Json
};
//Format the header
request.AddHeader("Accept", "application/json, text/json");

//Authentication. add your token as an Authentication header to your request
request.AddHeader("Authorization", "Token " + token);

//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);
// the Response should look like the following
//      {
//          "meta": {
//              "displayCategory": "Compliance",
//              "displayName": "Productversion compliance per Legal Entity",
//              "displayDescription": "Shows Compliance overview per Productversion and Legal Entity",
//              "guid": "b35ab679-55a5-4219-a9fe-230e7934f68d",
//              "category": "Compliance",
//              "name": "Productversion compliance per Legal Entity",
//              "description": "Shows Compliance overview per Legal Entity",
//              "id": 52,
//              "applicationName": "Spider Licence",
//              "applicationGuid": "135ea316-a932-44c9-bffb-fbd128754884"
//          },
//          "fields": [
//              {
//                  "inResult": false,
//                  "fieldName": "ManufacturerID",
//                  "displayName": "Manufacturer",
//                  "dataType": "int"
//              },
//              {
//                  "inResult": true,
//                  "fieldName": "Manufacturer",
//                  "displayName": "Manufacturer",
//                  "dataType": "nvarchar"
//              },
//              {
//                  "inResult": false,
//                  "fieldName": "ProductVersionID",
//                  "displayName": "ProductVersionID",
//                  "dataType": "int"
//              },
//          ],
```

```
// {
//   "inResult": true,
//   "fieldName": "ProductVersion",
//   "displayName": "Product version",
//   "dataType": "nvarchar"
// },
// {
//   "inResult": false,
//   "fieldName": "LegalEntityID",
//   "displayName": "Legal Entity",
//   "dataType": "int"
// },
// {
//   "inResult": true,
//   "fieldName": "LegalEntity",
//   "displayName": "Legal Entity",
//   "dataType": "nvarchar"
// },
// {
//   "inResult": true,
//   "fieldName": "ComplianceICON",
//   "displayName": "ComplianceICON",
//   "dataType": "nvarchar"
// },
// {
//   "inResult": true,
//   "fieldName": "LicencePosition",
//   "displayName": "LicencePosition",
//   "dataType": "int"
// },
// {
//   "inResult": true,
//   "fieldName": "LicenceRequirement",
//   "displayName": "License Requirement",
//   "dataType": "int"
// },
// {
//   "inResult": true,
//   "fieldName": "LicenceEntitlement",
//   "displayName": "LicenceEntitlement",
//   "dataType": "int"
// },
// {
//   "inResult": true,
//   "fieldName": "DowngradeAllocation",
//   "displayName": "DowngradeAllocation",
//   "dataType": "int"
// },
// {
//   "inResult": true,
//   "fieldName": "PoolingAllocation",
//   "displayName": "PoolingAllocation",
//   "dataType": "int"
// },
// {
```

```

// "inResult": true,
// "fieldName": "Consumed",
// "displayName": "Consumed",
// "dataType": "int"
// },
// {
// "inResult": true,
// "fieldName": "RemainingLicencesShort",
// "displayName": "RemainingLicencesShort",
// "dataType": "int"
// }
// ],
// "notifications": [],
// "duration": 479,
// "selfLink": "http://yourWebServer/SpiderAPI/0/meta/report/52",
// "parameter": {
// "cultureCode": null
// }
// }
//}

// Optional: Exception handling
if (response.ErrorException != null)
{
    const string message = "Error retrieving response. Check inner details for more info.";
    var yourException = new ApplicationException(message, response.ErrorException);
    throw yourException;
}
}

```

## Beispiel: GetMetaEntities

```

private void GetMetaEntitiesExample(string token)
{
    // The url of mandator you from the web api help page
    var url = "{mandatorid}/meta/entities";
    // The Url of the Spider Web API
    var yourApiUrl = "http://yourWebServer/SpiderWebApi";

    // Create a Rest Client

    RestClient client = new RestClient {BaseUrl = new Uri(yourApiUrl)};

    // Initialize your Rest Request

    RestRequest request = new RestRequest
    {
        Resource = url,
        RequestFormat = DataFormat.Json
    };
    //Format the Header

    request.AddHeader("Accept", "application/json, text/json");

    //Authentication add your token to your request object.
    request.AddHeader("Authorization", "Token " + token);
}

```

```
//Optional, Trust the security certificate
//ServicePointManager.ServerCertificateValidationCallback += (sender, certificate, chain, sslPolicyErrors) => true;

//Send the Request
IRestResponse response = client.Execute(request);

//the response content should look like the following
//{{
//  "entities": [

//    {
//      "name": "Article",
//      "displayName": "Article",
//      "parentName": "",
//      "parentDisplayName": ""
//    },
//    {
//      "name": "LicenceAllocation",
//      "displayName": "LicenceAllocation",
//      "parentName": "",
//      "parentDisplayName": ""
//    },
//    {
//      "name": "LicenceKey",
//      "displayName": "License key",
//      "parentName": "",
//      "parentDisplayName": ""
//    },
//    {
//      "name": "Product Licence",
//      "displayName": "Product Licence",
//      "parentName": "Licence",
//      "parentDisplayName": "License"
//    },
//    {
//      "name": "Product Change Licence",
//      "displayName": "Product Change Licence",
//      "parentName": "Licence",
//      "parentDisplayName": "License"
//    },
//    {
//      "name": "OEM Licence",
//      "displayName": "OEM Licence",
//      "parentName": "Licence",
//      "parentDisplayName": "License"
//    },
//  ],
//  "notifications": [],
//  "duration": 7,
//  "selfLink": "http://yourServer/SpiderWebApi/0/meta/entities",
//  "parameter": {
//    "cultureCode": null
```

```
// }  
//}  
  
// Optional: Exception handling  
  
if (response.Exception != null)  
{  
    const string message = "Error retrieving response. Check inner details for more info.";  
    var yourException = new ApplicationException(message, response.Exception);  
    throw yourException;  
}  
}
```

## Software Services (Kataloge u. Recognition)

---

### 5.1 Systemvoraussetzungen

---

#### 5.1.1 Applikationsserver

---

Die Installation von Recognition kann entweder auf dem System ausgeführt werden, auf dem Spider installiert worden ist oder auf einem separaten System.

Es werden zusätzliche Dienste für die Funktionalität der Recognition installiert.

- EDC-Monitor
- Spider Data Receiver
- Software Recognition Center Upload (optional)

Die Datenbanken müssen auf demselben Server (und Instanz) liegen, auf dem auch die Datenbanken für Spider liegen.

Im Verlauf des Setups wird die Verbindung zum Datenbankserver abgefragt.

Folgende Gegebenheiten müssen beachtet werden:

- Das Setup muss mit administrativen Rechten ausgeführt werden.
- Recognition (RC) und Spider Data Collector müssen auf getrennten Systemen laufen. Dies können auch getrennte virtuelle Umgebungen sein.
- Auf dem Spider Applikationsserver darf daher entweder RC oder der Spider Data Collector installiert werden
- Port für die Kommunikation von Software Recognition mit anzuschließendem(n) Spider Data Collector(en), der Port ist frei wählbar, Standardmäßig wird Port **1685** verwendet.
- Bei Produktivsystemen müssen Adresse und Port für den Upload zum Software Recognition Center von dem Server aus erreichbar sein
  - Adresse: **edcsrv1.brainwaregroup.com**
  - Port: **1337**

---

**Hinweis**      Testsysteme ohne Upload zum Software Recognition Center.

Hier können Software Recognition und Spider Data Collector parallel installiert werden.

---

## 5.1.2 Datenbankserver

---

Als Datenbankserver wird ein Microsoft SQL Server ab der Version 2012 benötigt. Für ein erfolgreiches Setup sind folgende Voraussetzungen auf dem Datenbankserver zu schaffen.

Der Datenbankserver muss der gleiche sein auf dem die Spider Datenbanken installiert sind.

- Im SQL Server Configuration Manager muss für "SQL Server Configuration" das TCP/IP Protokoll aktiviert sein.
- Es wird ein Datenbanklogin für den SQL-Installationsbenutzer benötigt.
- Sofern der SQL-Installationsbenutzer nicht das Recht besitzt Datenbanken einzurichten, muss diese vor der Installation bereitgestellt werden.

Sofern der SQL-Installationsbenutzer nicht das Recht besitzt weitere SQL-Logins auf dem Datenbankserver einzurichten, muss der erforderliche SQL-Login vorab eingerichtet und für das Setup bereitgestellt werden. Es wird empfohlen, denselben Benutzer zu verwenden der auch für die Spider Installation verwendet wurde, da die Spider Dienste auch auf die Software Recognition Datenbank zugreifen müssen.

- Der SQL-Installationsbenutzer benötigt mindestens das Recht zum Ausführen der SQL-Skripte zum Anlegen und Ändern der Datenbankobjekte und der Datenbankeigenschaften (Extended Properties) in den Spider Datenbanken.

### Benötigte Datenbank

---

Zum Anlegen der Datenbanken im Zuge der Installation benötigt der SQL-Installationsbenutzer entsprechende Rechte. Können diese nicht gewährt werden, müssen die Produktdatenbanken vor Ausführung des Setups angelegt werden. Das Setup unterstützt die Verwendung von leeren Datenbanken, die im Vorfeld angelegt werden. Speziell wenn die Datenbanken nicht mit den Standardeigenschaften angelegt werden sollen, können die erforderlichen Datenbanken so mit den gewünschten Einstellungen bereitgestellt werden.

Folgende Datenbanken werden für die Installation benötigt:

- SpiderRecognition

<b>Wichtig</b>	Es wird empfohlen das Suffix (im Beispiel später: "CoreServer_00_"), das für die Spider Installation verwendet wurde, auch für die Recognition Datenbank zu übernehmen. Das macht die Identifikation auf dem SQL Server einfacher.
----------------	--

Folgendes ist zu beachten:

- Der Name kann individuell gewählt werden. Der Datenbankname darf nicht mit einer Zahl beginnen.
- Alle Spider Datenbanken müssen dieselbe Datenbank Kollation aufweisen. Es muss eine Case-Insensitive-Collation (CI-Collation) gewählt werden, bspw. SQL\_Latin1\_General\_CP1\_CI\_AS.
- Die Datenbanken müssen vor der Installation komplett leer sein.

### Benötigter SQL-Benutzer

---

Wenn der SQL-Installationsbenutzer nicht das Recht hat neue Logins auf dem Datenbankserver anzulegen, muss dieser vorab angelegt werden. Für das Setup wird ein SQL-Benutzer benötigt. Die Anmeldedaten werden während der Installation abgefragt. Der Benutzer muss der gleiche sein wie für die Spider Installation, denn die Dienste von Spider müssen auf die Software Recognition Datenbank zugreifen können.

## Übersicht erforderlicher Rechte

---

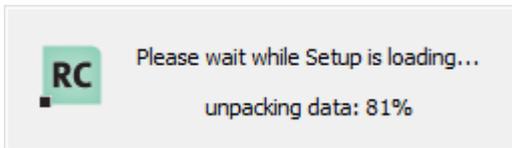
Die erforderlichen Rechte für die Ausführung des Setups entnehmen Sie bitte der unten stehenden Tabelle:

Task	Rechte an	Recht(e)
Erstellen von Logins	Server	ALTER ANY LOGIN
Erstellen von Datenbanken	Server	CREATE DATABASE CREATE ANY DATABASE ALTER ANY DATABASE
Erstellen Extended Properties	Datenbank	ALTER

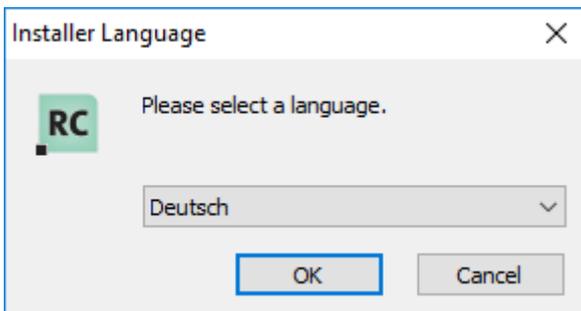
## 5.2 Installation

### 5.2.1 Start der Installation

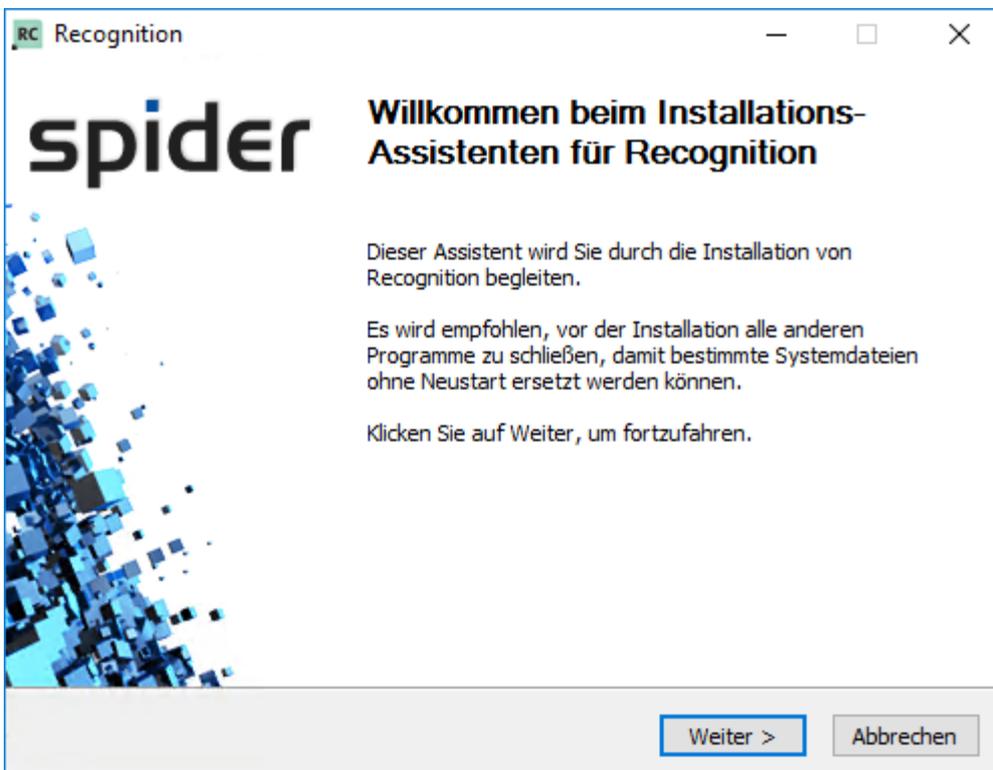
Starten Sie die Installation, indem Sie die Recognition Setup.exe ausführen



Zunächst werden Sie dazu aufgefordert, die Sprache für den Installationsassistenten auszuwählen. Klicken Sie hierfür auf die gewünschte Sprache in der Auswahlbox und bestätigen Sie Ihre Eingabe mit „OK“.



Anschließend öffnet sich der Installationsassistent für Spider. Er wird Ihnen in der gewählten Sprache bei der Installation des Setups behilflich sein. Klicken Sie nun auf „Weiter“ um die Installation zu starten.



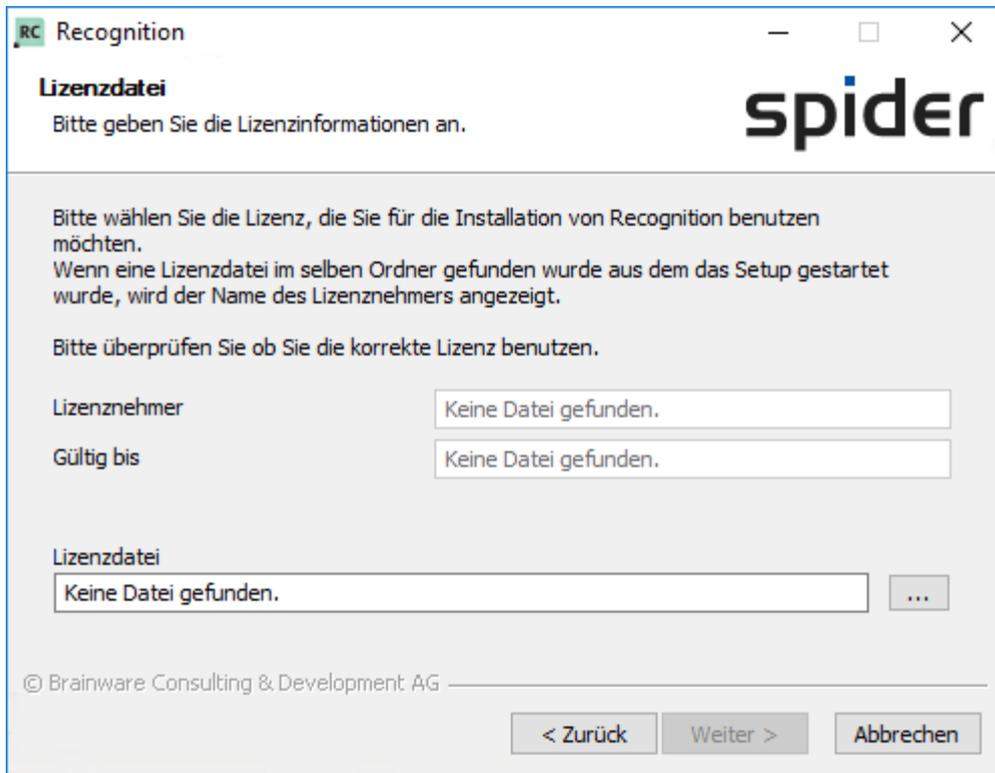
## 5.2.2 Lizenzvereinbarung

Im nächsten Schritt werden Sie gebeten die Lizenzvereinbarung vor Beginn der Installation zu lesen und zu akzeptieren. Das Setup lässt sich nur mit akzeptierter Lizenzvereinbarung weiter ausführen.

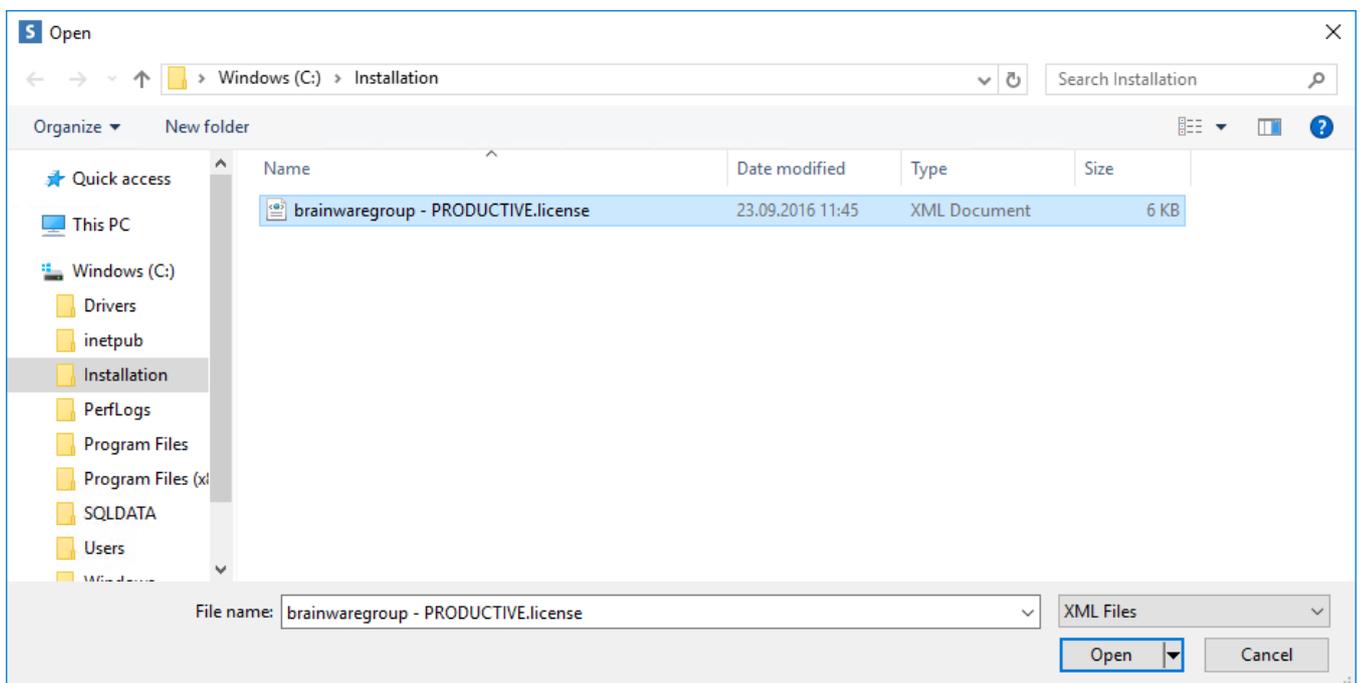


## 5.2.3 Lizenzdatei

Wählen Sie die passende Lizenzdatei aus, um das Setup fortzuführen. Ohne Lizenzdatei lässt sich das Setup nicht weiter fortführen



Klicken auf die "..." Schaltfläche:



Auswahl der Lizenz und dann "Open".

Die ausgewählte Lizenz wird geladen und die Details angezeigt.

RC Recognition

**Lizenzdatei**  
Bitte geben Sie die Lizenzinformationen an.

spider

Bitte wählen Sie die Lizenz, die Sie für die Installation von Recognition benutzen möchten.  
Wenn eine Lizenzdatei im selben Ordner gefunden wurde aus dem das Setup gestartet wurde, wird der Name des Lizenznehmers angezeigt.

Bitte überprüfen Sie ob Sie die korrekte Lizenz benutzen.

Lizenznehmer

Gültig bis

Lizenzdatei  
 ...

© Brainware Consulting & Development AG

< Zurück **Weiter >** Abbrechen

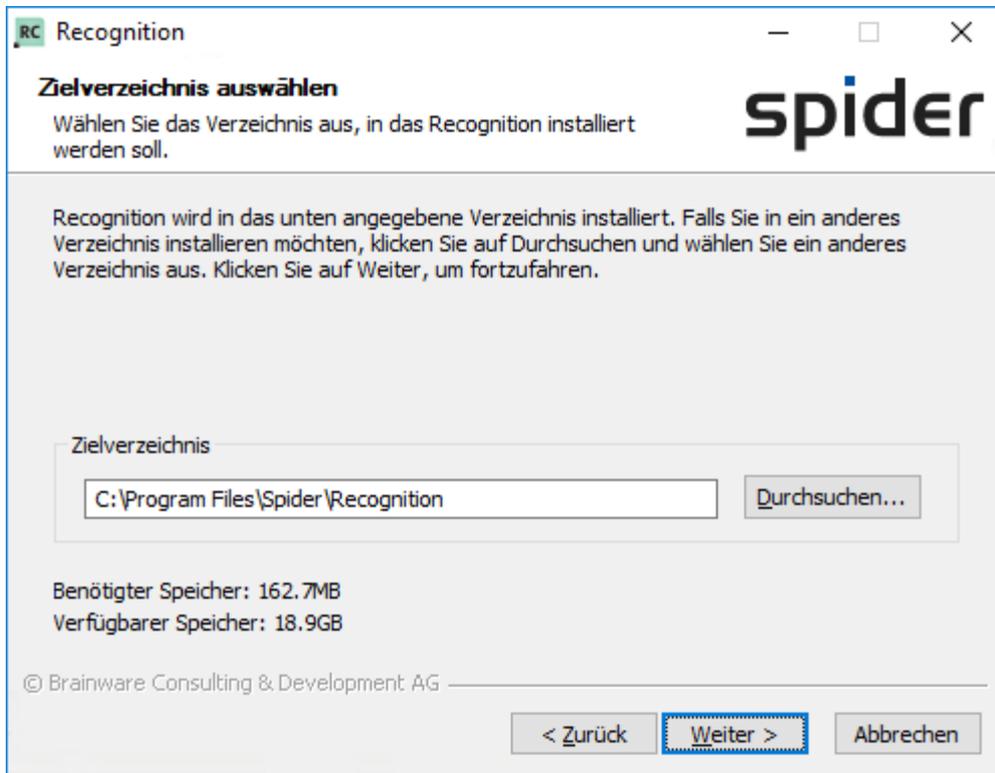
---

**Hinweis** Falls eine Lizenz im gleichen Verzeichnis gefunden wird, aus der das Setup gestartet wurde, wird diese (sofern sie gültig ist) automatisch in den Dialog geladen.

---

## 5.2.4 Installationsverzeichnis

Im nun folgenden Schritt können Sie ein Zielverzeichnis für die Installation auswählen. Den vorgeschlagenen Standardpfad können Sie ändern.



## 5.2.5 Datenbankserver

Im nun folgenden Schritt bittet Sie der Installationsassistent den Namen und die Instanz eines Microsoft SQL-Servers anzugeben, auf dem die Recognition Datenbank installiert werden soll.

RC Recognition

**SQL Verbindung und Installationsbenutzer.**  
Diese Verbindung wird zum Ausführen der Installation verwendet.

Bitte geben Sie Namen und Instanz des SQL Servers für Recognition an.

SQL Server Name (optional Instanzname)  
.\

SQL Benutzer für Datenbankverbindung nutzen.

Benutzername:  
Passwort:

© Brainware Consulting & Development AG

< Zurück Weiter > Abbrechen

Der Benutzer, mit dem die Verbindung hergestellt wird, kann entweder der angemeldete (Windows-)Benutzer sein, oder es kann ein SQL Benutzer verwendet werden.

Soll ein SQL Benutzer verwendet werden, ist das Kontrollkästchen **SQL Benutzer für Datenbankverbindung nutzen** zu markieren sowie die Felder **Benutzername** und **Passwort** mit den gewünschten Daten auszufüllen.

RC Recognition

**SQL Verbindung und Installationsbenutzer.**  
Diese Verbindung wird zum Ausführen der Installation verwendet. **spider**

Bitte geben Sie Namen und Instanz des SQL Servers für Recognition an.

SQL Server Name (optional Instanzname)  
.\

SQL Benutzer für Datenbankverbindung nutzen.

Benutzername:  
<SQL Benutzer >

Passwort:  
●●●●●

© Brainware Consulting & Development AG

< Zurück Weiter > Abbrechen

**Hinweis** Die Benutzer müssen vorab auf dem Datenbankserver mit den entsprechenden Rechten eingerichtet werden. Abhängig von den Rechten, mit denen der Benutzer ausgestattet ist, stehen in den weiteren Folgedialogen verschiedene Optionen zur Verfügung.

Feld	Beschreibung
SQL Server Name	Name des SQL Servers, ggfs. inklusive Instanznamen und Port z.B. SQLSERVER\SQLINSTANZ,1234
SQL Benutzer für Datenbankverbindung nutzen	Auswahl, ob der angemeldete Domainbenutzer verwendet wird oder ob ein SQL Benutzer verwendet werden soll.
Benutzername	Name des SQL Benutzers.
Passwort	Passwort des SQL Benutzers.

## 5.2.6 Datenbank & Data Receiver

In diesem Dialog werden die Angaben zur Recognition Datenbank, der zu verbindenden Core Datenbank der Spider Installation, dem Datenverzeichnis, Recognition Upload und Customer ID gemacht.

Feld	Beschreibung
Datenbank	Per Dropdown Auswahl, ob eine neue Datenbank erstellt werden soll oder eine bestehende (leere!) Datenbank wählen
Core DB	Name der Core Datenbank der zu verknüpfenden Spider Installation auswählen
Datenverzeichnis	Das Arbeitsverzeichnis in dem entgegengenommene Daten des/der Data Collector(en) abgelegt und weiterverarbeitet werden.
Port	Auf diesem Port empfängt der Data Receiver die Daten der angeschlossenen Data Collectoren (Standard-Port: 1685)
Recognition Center Upload	Kontrollkästchen mit dem bestimmt wird ob unerkannte Software zur Erkennung an Flexera übertragen werden soll.
Upload ID	ID (aus der Lizenzdatei) mit der sich zum Upload bei Flexera identifiziert wird.
Customer ID	Customer ID die später dem/den Data Collector(en) mitgegeben werden kann um sich gegenüber dieser Installation von Recognition zu authentifizieren.

## 5.2.7 SQL-Login, Service Benutzer

Dieser SQL-Benutzer wird für den Zugriff auf die Software Recognition-Datenbank verwendet.

Die Spider-Services greifen mit den Core-Server-Benutzerdaten auch auf die Recognition Datenbanken zu!

Es muss der Standardssystemuser «spider» bzw. der Benutzers, der bei der Installation von Spider verwendet wurde, angegeben werden.

RC Recognition

**SQL Login**  
Bitte geben Sie einen Benutzer zum anmelden an.

spider

Bestehenden Benutzer verwenden

Benutzername: Spider

Passwort: ●●●●●

Neuen Benutzer erstellen

Benutzername: Recognition

Passwort: [ ] Passwort: [ ]

Bitte wählen Sie ein Passwort das den Komplexitätsrichtlinien ihrer Domain und/oder Ihres SQL Servers entspricht! Bitte stellen sie unbedingt sicher, dass bestehende Spider-Services sowie die Spider Web Application mit ihren Benutzereinstellungen Zugriff auf die Software Recognition Datenbank haben.

© Brainware Consulting & Development AG

< Zurück Weiter > Abbrechen

**Achtung** Bei der Anlage eines neuen Benutzers muss dieser auch an Spider berechtigt werden! Damit die Spider Services auch auf die Recognition Datenbank zugreifen können, muss auch der Benutzer, den die Services verwenden auf der Recognitiondatenbank berechtigt werden.

RC Recognition

**SQL Login**  
Bitte geben Sie einen Benutzer zum anmelden an.

spider

Bestehenden Benutzer verwenden

Benutzername: Spider  
Passwort:

Neuen Benutzer erstellen

Benutzername: Recognition  
Passwort: .....  
Passwort: .....

Bitte wählen Sie ein Passwort das den Komplexitätsrichtlinien ihrer Domain und/oder Ihres SQL Servers entspricht! Bitte stellen sie unbedingt sicher, dass bestehende Spider-Services sowie die Spider Web Application mit ihren Benutzereinstellungen Zugriff auf die Software Recognition Datenbank haben.

© Brainware Consulting & Development AG

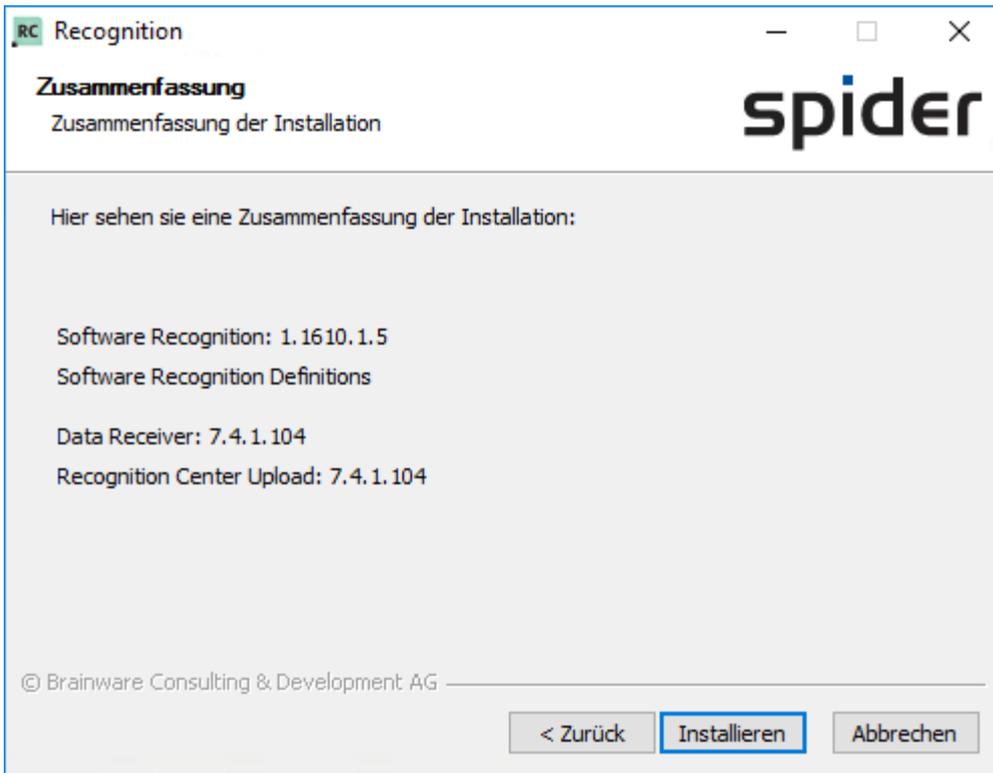
< Zurück   Weiter >   Abbrechen

## 5.2.8 Zusammenfassung

Die Zusammenfassung zeigt dass:

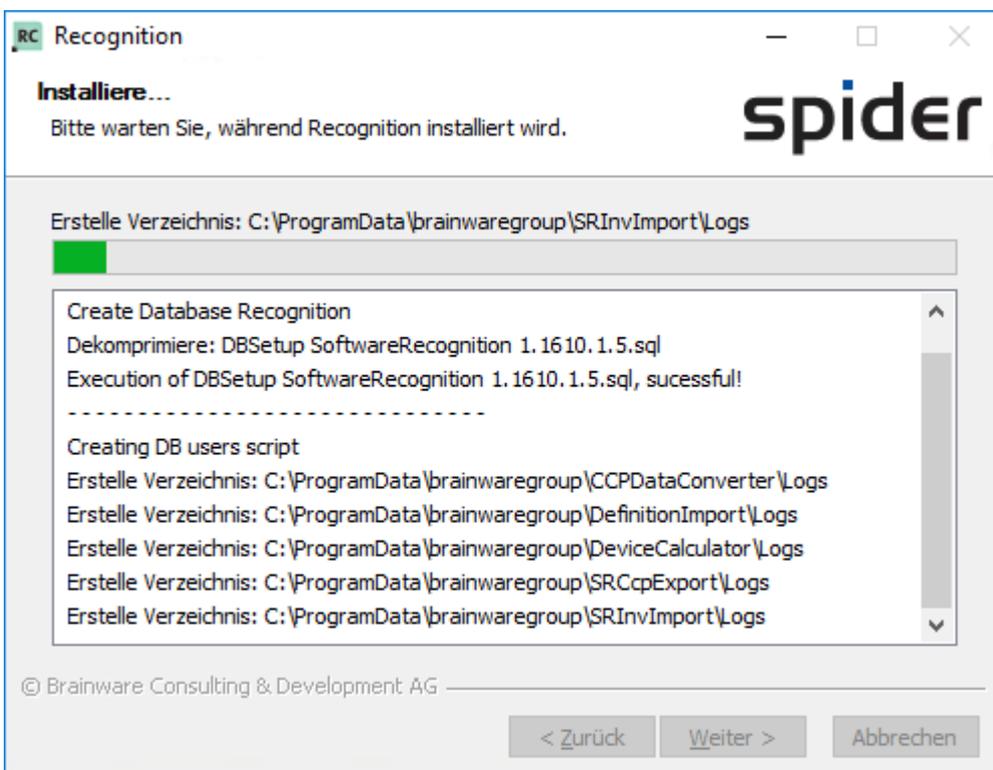
- die Version von Software Recognition
- die Software Recognition Definitionen
- die Data Receiver Komponenten und Version
- der Upload zu dem Spider Recognition Center mit Version

installiert werden.

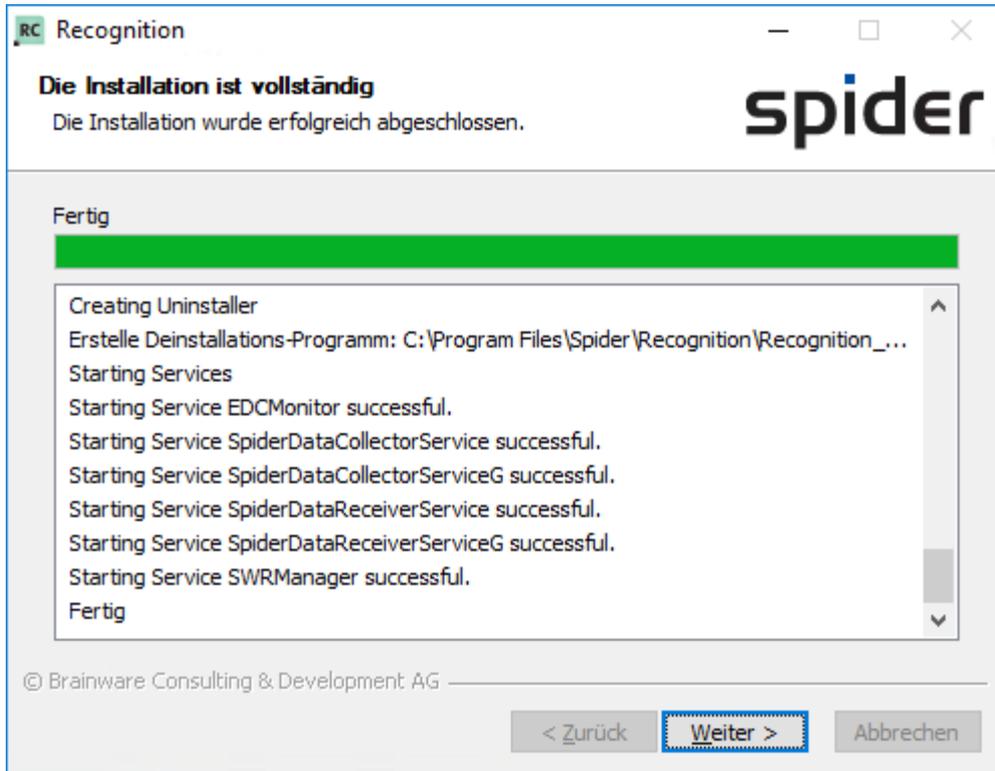


## 5.2.9 Installation

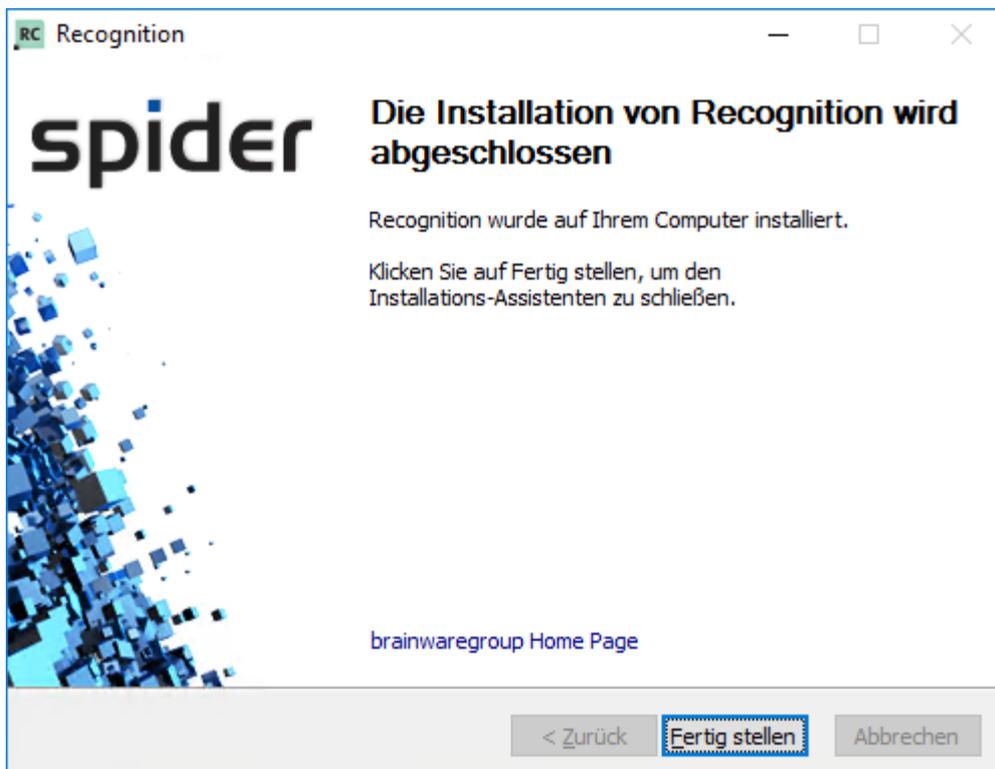
Der Installationsassistent installiert nun Recognition. Dies kann einige Zeit in Anspruch nehmen. Bitte warten Sie, bis die Installation beendet ist.



Nach erfolgreichem Abschluss der Installation, wird dies im Fortschrittsdialog angezeigt.



Das Setup ist erfolgreich abgeschlossen und kann beendet werden



Nach Abschluss der Installation wird ein Logfile geschrieben. Details zu den Logfiles, sind im Kapitel [Logfiles](#) (siehe Seite 210) beschrieben.

## 5.3 Erweiterte Konfiguration

### 5.3.1 Data Receiver

Der Data Receiver ist die Komponente, die Lieferungen der/des Data Collector(en) entgegennimmt.

Der Port und der Ablageort der empfangenen Dateien können verändert werden.

Die Konfiguration erfolgt über die Datei <Installationsverzeichnis>\Recognition\DataReceiver\SpiderDataReceiver.cfg

```

; OTB Connection Settings:
; -----
; OTBPort - default setting is "80"

[Connection]
OTBPort=1685

; General settings:
; -----
; DataDirectory - is the data directory to save files in that should be uploaded - default if not set is the folder where the
EDCSRV.exe is installed (environment variables are resolved)

[General]
DataDirectory=C:\RecognitionData\DataReceiver

```

Sektion	Variable	Beschreibung
Connection	OTBPort	Port auf dem der Data Receiver auf die Verbindungen der/des Data Collector(en) hört
General	DataDirectory	Pfad in dem die empfangenen Dateien abgelegt werden.

**Achtung** Wenn die Variable DataDirectory angepasst wird, ist der Dienst "Spider Data Receiver" durchzustarten. Nur dann wird die Änderung auch aktiv.

Zusätzlich muss die Konfiguration des [EDC-Monitors](#) (siehe "" auf Seite 178) angepasst werden, damit dieser die zu verarbeitenden Dateien findet.

### SSL verschlüsselte Übertragung

Die Kommunikation mit dem Data Collector kann per SSL verschlüsselt werden (RSA 2048Bit).

Dabei übernimmt der übergeordnete Data Receiver die Aufgabe SSL für seine angeschlossenen Data Collectoren bereit zu stellen.

Nach der Installation des Data Receivers, liegen die benötigten Dateien in: %ProgramData%\Columbus

Die Verwendung von SSL kann durch die folgenden Registry Einträge ermöglicht und erzwungen werden.

```
Key: HKEY_LOCAL_MACHINE\SOFTWARE\<Wow6432Node>\BrainWare\Columbus\7\
"OTBEncryptionUseSSL"="1"
"OTBEncryptionForceUseSSL"="0"
```

## 5.3.2 EDC-Monitor

Der EDC Monitor ist verantwortlich für den Import, der durch den Data Receiver empfangenen Dateien, in die Recognition Datenbank.

Die Konfiguration des EDC-Monitors ist in der Datei <Installationspfad>\Recognition\EDCMonitor\EDCMonitor.exe.config abgelegt.

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <connectionStrings>
    <!-- Connection string to the software recognition database -->
    <add name="SoftwareRecognitionConnectionString" connectionString="server=server.domain.local; database=Recognition;
uid=Spider; pwd=abcde; application name=SpiderEDCMonitor;Persist Security Info=True" />
  </connectionStrings>
  <appSettings>
    <!-- Working directory of the EDC Monitor
    Has to be the path where EDCMonitor.exe is located -->
    <add key="EDCMonitorWorkingDirectory" value="C:\Program Files\Spider\Recognition\EDCMonitor" />
    <!-- Working directory of Data Receiver.
    Following files are expected in the directory:
    * SpiderDataReceiver.cfg
    * SpiderDataReceiverRegister.csv
    * SpiderDataReceiverConnection.csv -->
    <add key="DataReceiverWorkingDirectory" value="C:\Program Files\Spider\Recognition\DataReceiver" />
    <!-- Location, where files are committed to software recognition.
    The data directory of the EDC Server and this directory should be located on the same HDD. -->
    <add key="SoftwareRecognitionImportDirectory" value="C:\RecognitionData\EDC-Monitor" />
    <!-- Delay between monitoring processes (in minutes). -->
    <add key="Interval" value="1" />
    <!-- Directory where log files will be written
    Standard: %PROGRAMDATA%\brainwaregroup\EDCMonitor\Logs
    -->
    <add key="FeedbackDirectory" value="C:\ProgramData\brainwaregroup\EDCMonitor\Logs" />
  </appSettings>
</configuration>
```

Setting	Beschreibung
connectionsStrings. SoftwareRecognitionConnectionString	Angaben zur Verbindung auf die Recognition Datenbank, bei Verwendung von Integrated Security muss uid=...;pwd=...; durch "IntegratedSecurity = True ersetzt werden.
appSettings. EDCMonitorWorkingDirectory	Pfad in dem die EDCMonitor.exe abgelegt ist.
appSettings. DataReceiverWorkingDirectory	Pfad in dem die SpiderDataReceiver.exe und deren Konfigurationsdatei abgelegt sind.

Setting	Beschreibung
appSettings. SoftwareRecognitionImportDirectory	Pfad in dem der EDC-Monitor seine Dateien ablegt.
appSettings. Interval	Intervall in dem der EDC-Monitor aktiv wird, neue Dateien ermittelt und ggfs. importiert.
appSettings. FeedbackDirectory	Pfad in dem Logdateien abgelegt werden.

### 5.3.3 Betrieb mit Domänenkonto

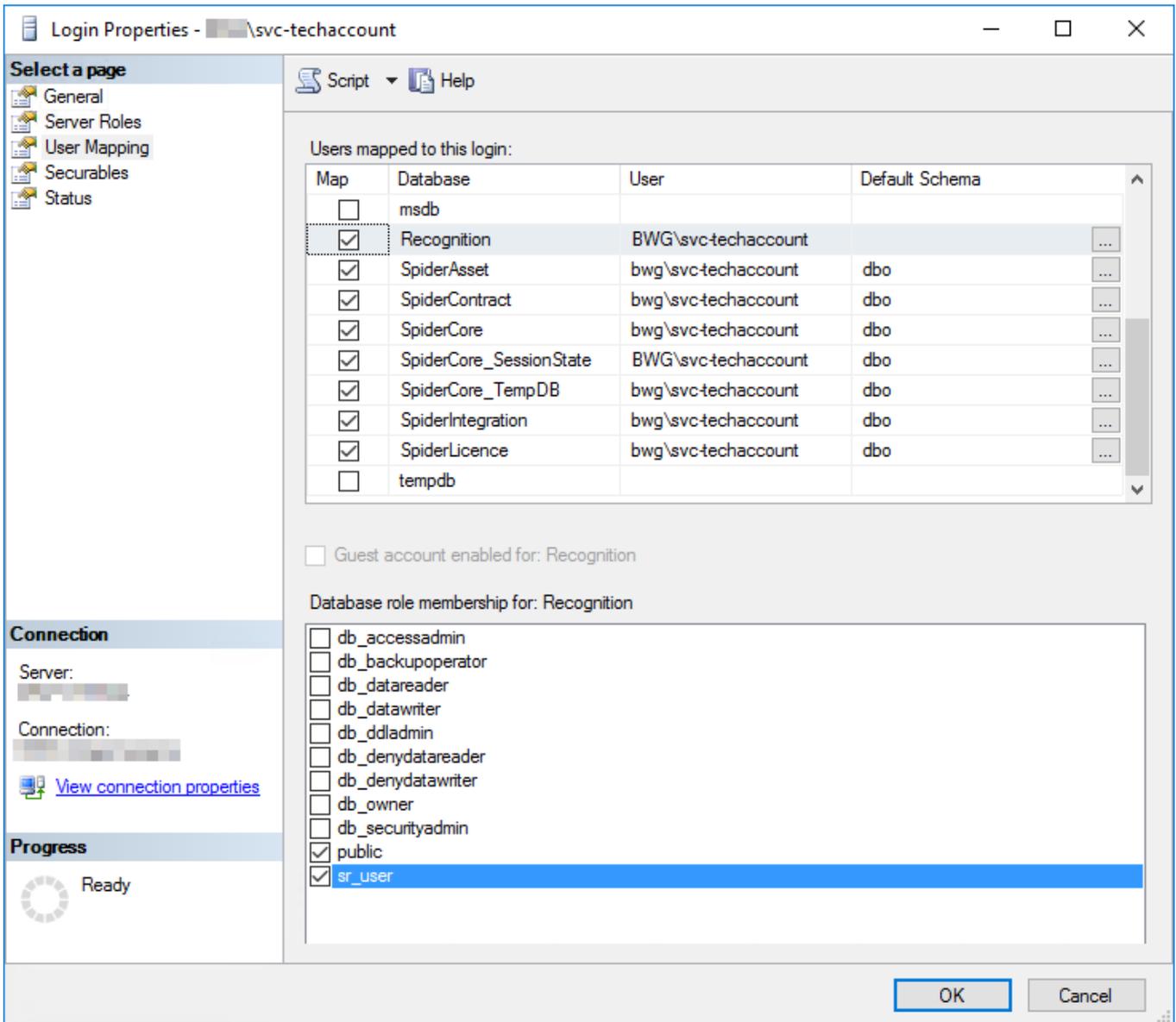
#### SQL Server vorbereiten

Die Einrichtung erfolgt analog der Beschreibung zur Einrichtung von Spider [SQL Server vorbereiten](#) (auf Seite 44).

Zusätzlich muss für die Recognition Datenbank der Benutzer neu bzw. zusätzlich berechtigt werden.

- Eintragen der **Rollen Mitgliedschaften** für die Recognition Datenbank durch Setzen des Hakens bei **Map** und Auswählen der Mitgliedschaft im unteren Teil des Dialogs.

Datenbank	Database Role Membership
Recognition	public, sr_user



## Ordnerberechtigungen

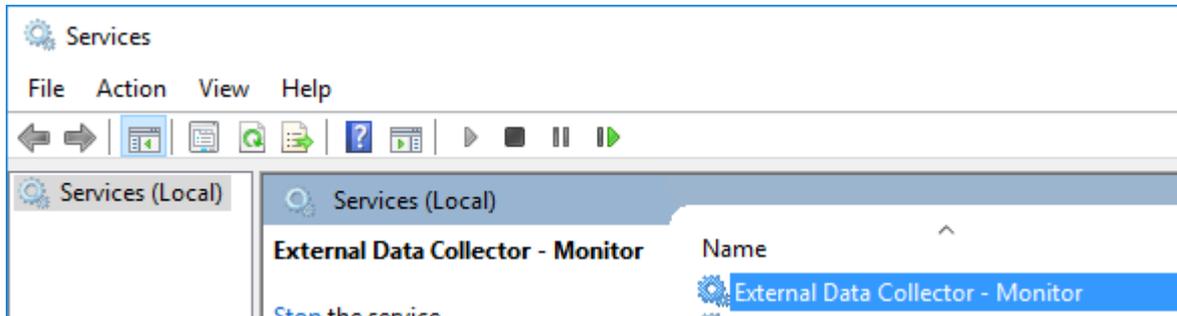
Damit der Zugriff auf die abgelegten Dateien möglich ist, können dem Benutzer entweder administrative Rechte auf dem Computer gegeben werden auf dem der EDC-Monitor installiert ist, oder aber man berechtigt den EDC-Monitor direkt oder per Gruppenmitgliedschaft auf die benötigten Ordner.

Ordner	Berechtigung
Installationsverzeichnis des EDC Monitors C:\Program Files\Spider\Recognition\EDCMonitor	Modify
Installationsverzeichnis des Data Receivers C:\Program Files\Spider\Recognition\DataReceiver	Modify
Datenverzeichnis <a href="#">Datenbank &amp; Data Receiver</a> (auf Seite 172)	Modify auf Datenverzeichnis und alle Unterordner

## Dienstkonto anpassen

Die Einrichtung erfolgt analog der Einrichtung von Spider [Dienstkonto anpassen](#) (siehe "" auf Seite 54).

Der anzupassende Dienst ist der **External Data Collector - Monitor**



## Connection String anpassen

Für den Zugriff des Services ist die Datei ConnectionStrings.config zuständig.

Ablageort des Connection Strings

Komponente	Ablageort
EDC-Monitor	C:\Program Files\Spider\Recognition\EDCMonitor\EDCMonitor.exe.config

Die Einträge uid=... und pwd=... müssen ersetzt werden durch **Integrated Security=True**.

### ConnectionStrings.config

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <connectionStrings>
    <!-- Connection string to the software recognition database -->
    <add name="SoftwareRecognitionConnectionString" connectionString="server=<SQLServer>;
      database=Recognition;
      uid=Spider;
      pwd=abcde;
      application name=SpiderEDCMonitor;
      Persist Security Info=True" />
  </connectionStrings>
  ...
</configuration>
```

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <connectionStrings>
    <!-- Connection string to the software recognition database -->
```

```
<add name="SoftwareRecognitionConnectionString" connectionString="server=<SQLServer>;
    database=Recognition;
    Integrated Security=True;
    application name=SpiderEDCMonitor;
    Persist Security Info=True" />
</connectionStrings>
...
</configuration>
```

---

**Hinweis** Nachdem alle Anpassungen durchgeführt wurden, muss der Service zur Aktivierung der Änderungen neu gestartet werden.

---

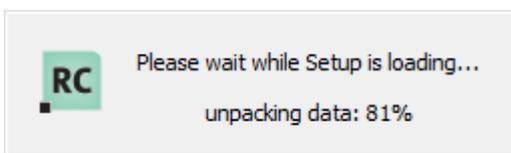
## 5.4 Aktualisierung

---

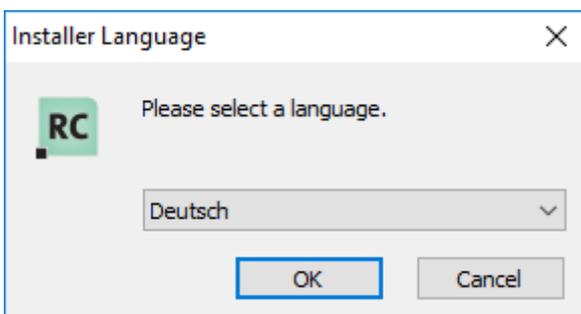
### 5.4.1 Start des Updates

---

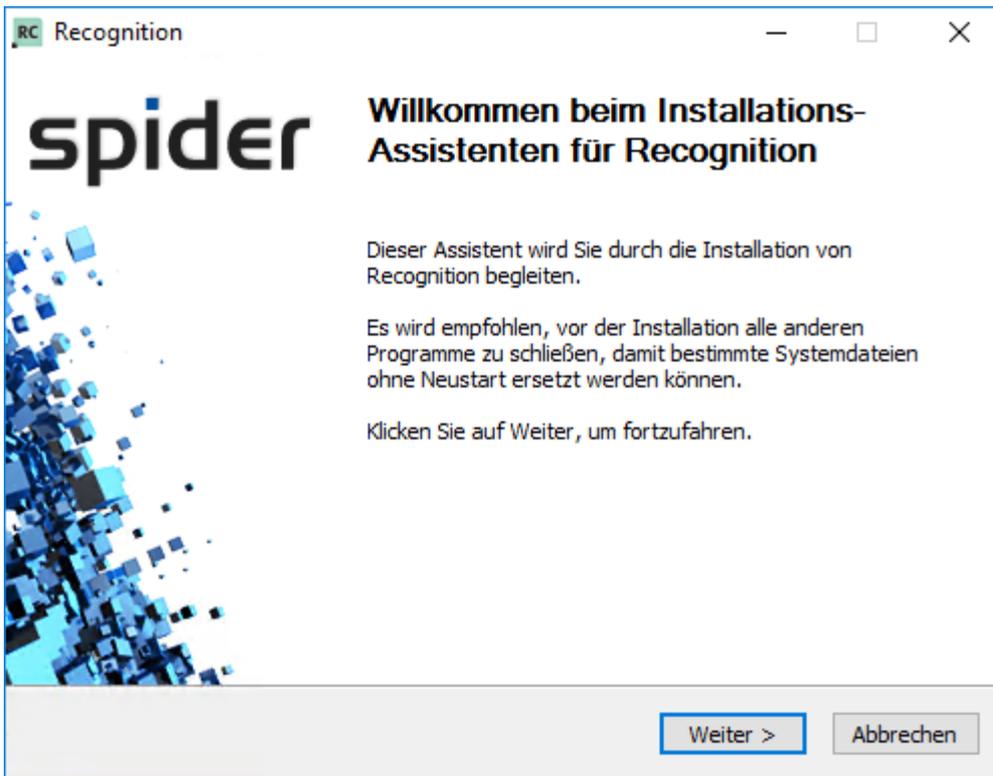
Starten Sie die Installation, indem Sie die Recognition Setup.exe ausführen



Zunächst werden Sie dazu aufgefordert, die Sprache für den Installationsassistenten auszuwählen. Klicken Sie hierfür auf die gewünschte Sprache in der Auswahlbox und bestätigen Sie Ihre Eingabe mit „OK“.



Anschließend öffnet sich der Installationsassistent für Spider. Er wird Ihnen in der gewählten Sprache bei der Installation des Setups behilflich sein. Klicken Sie nun auf „Weiter“ um die Installation zu starten.



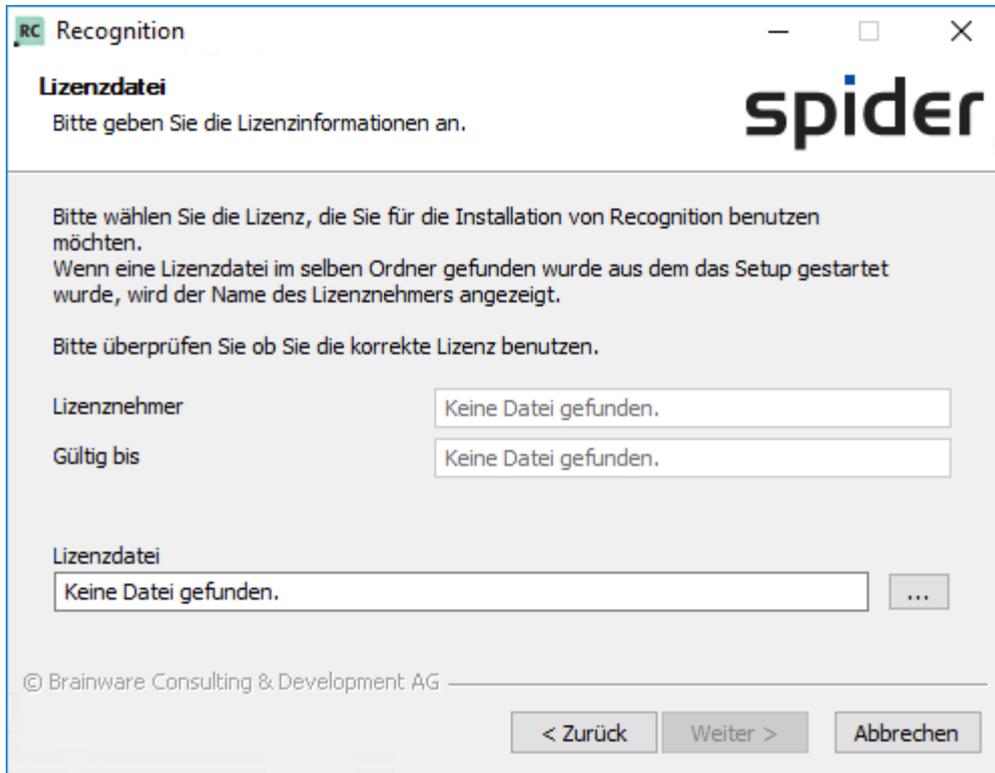
## 5.4.2 Lizenzvereinbarung

Im nächsten Schritt werden Sie gebeten die Lizenzvereinbarung vor Beginn der Installation zu lesen und zu akzeptieren. Das Setup lässt sich nur mit akzeptierter Lizenzvereinbarung weiter ausführen.

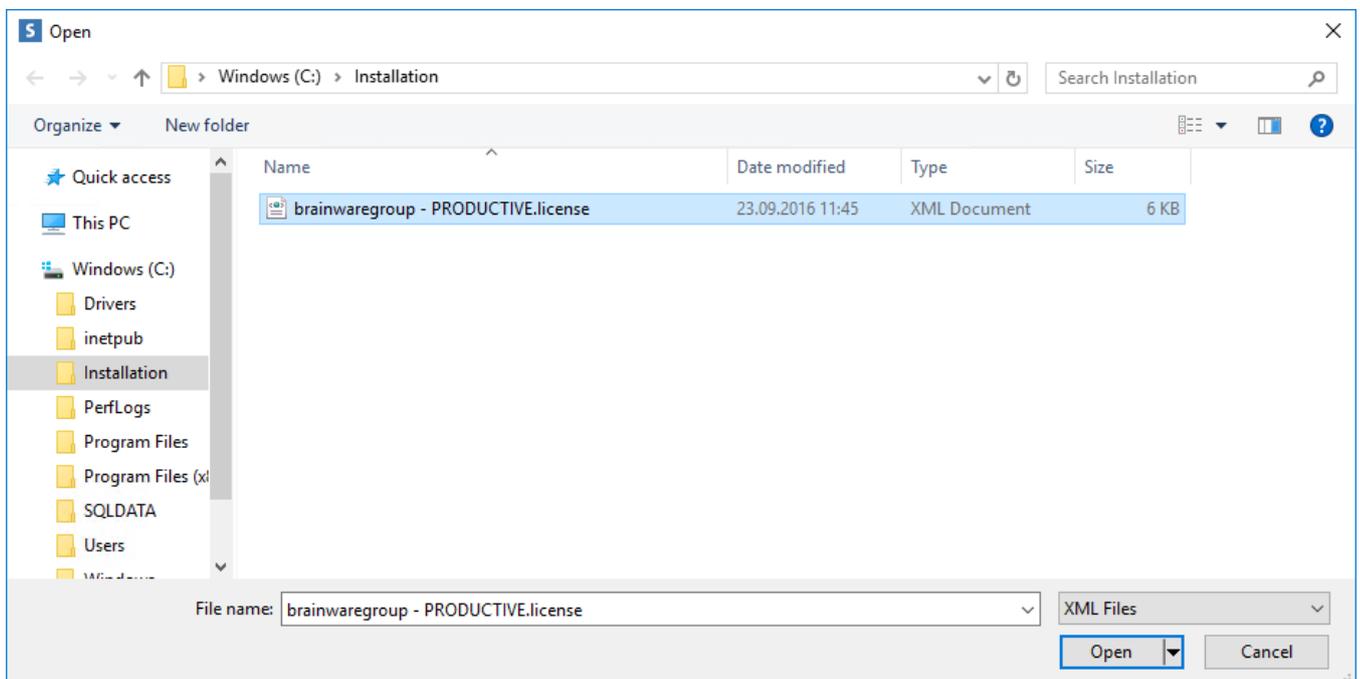


### 5.4.3 Lizenzdatei

Wählen Sie die passende Lizenzdatei aus, um das Setup fortzuführen. Ohne Lizenzdatei lässt sich das Setup nicht weiter fortführen



Klicken auf die "... " Schaltfläche:



Auswahl der Lizenz und dann "Open".

Die ausgewählte Lizenz wird geladen und die Details angezeigt.

RC Recognition

### Lizenzdatei

Bitte geben Sie die Lizenzinformationen an.

Bitte wählen Sie die Lizenz, die Sie für die Installation von Recognition benutzen möchten.  
Wenn eine Lizenzdatei im selben Ordner gefunden wurde aus dem das Setup gestartet wurde, wird der Name des Lizenznehmers angezeigt.

Bitte überprüfen Sie ob Sie die korrekte Lizenz benutzen.

Lizenznehmer

Gültig bis

Lizenzdatei  ...

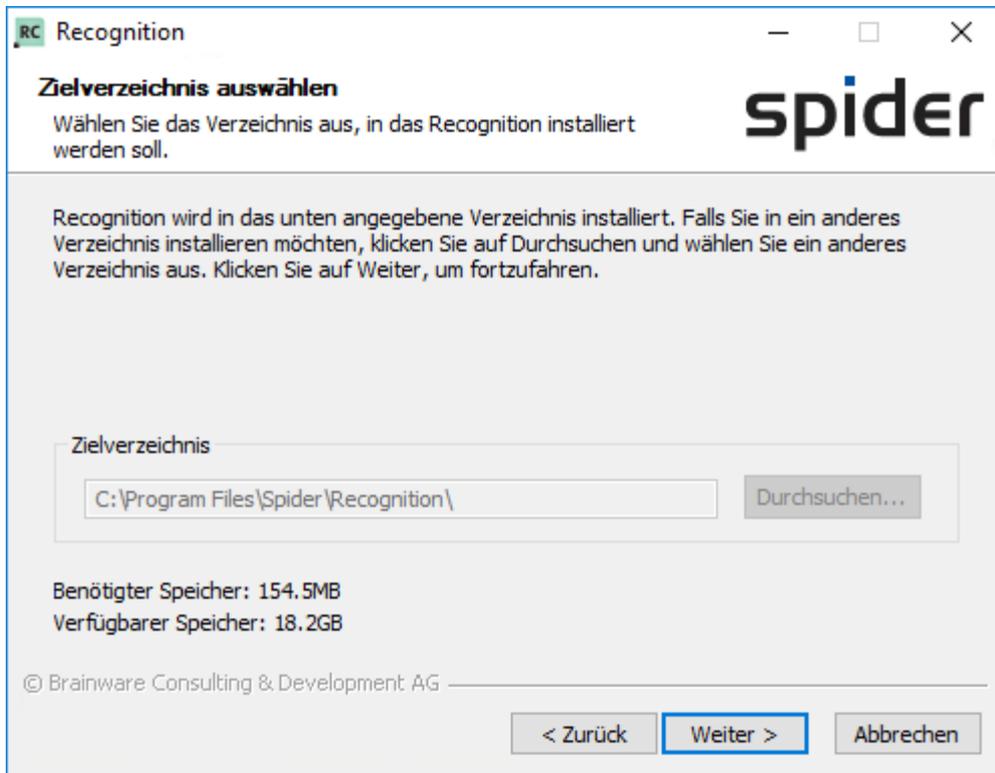
© Brainware Consulting & Development AG

< Zurück Weiter > Abbrechen

**Hinweis** Falls eine Lizenz im gleichen Verzeichnis gefunden wird, aus der das Setup gestartet wurde, wird diese (sofern sie gültig ist) automatisch in den Dialog geladen.

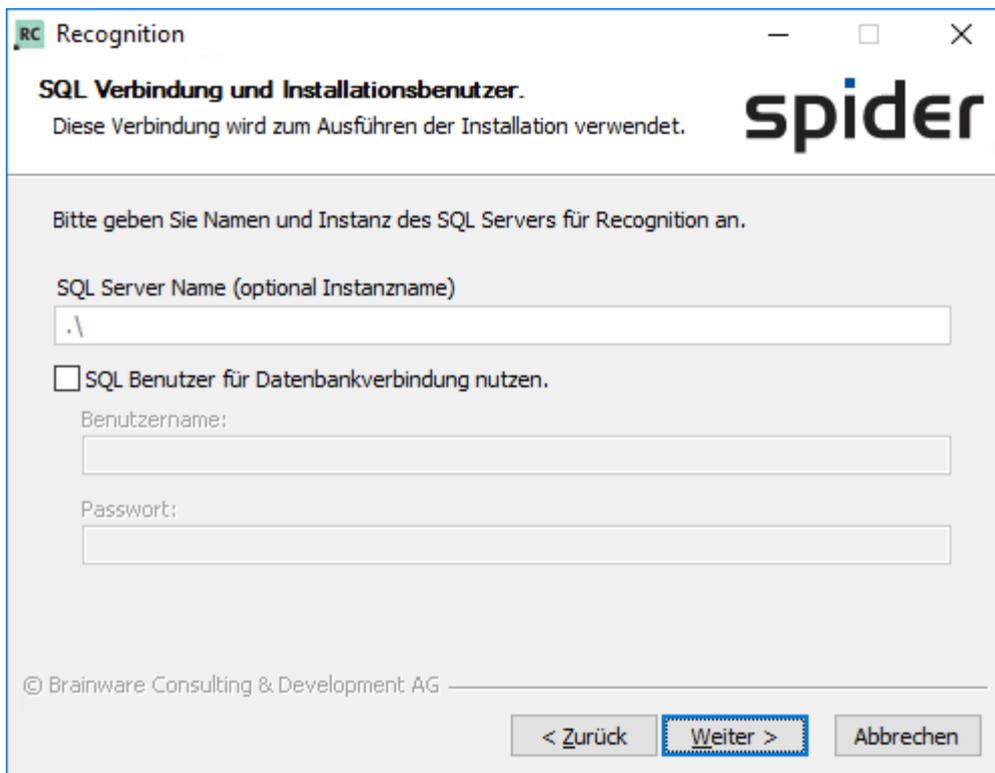
## 5.4.4 Installationsverzeichnis

Bei der Aktualisierung wird die Auswahl des Installationsverzeichnisses unterbunden. Ein Update ist nur im bestehenden Pfad möglich.



## 5.4.5 Datenbankserver

Während einer Aktualisierung ist es nicht möglich den vorher verwendeten SQL Server zu verändern.



The screenshot shows a Windows dialog box titled "Recognition" with the "spider" logo. The main heading is "SQL Verbindung und Installationsbenutzer." Below it, a subtitle reads "Diese Verbindung wird zum Ausführen der Installation verwendet." The dialog prompts the user to "Bitte geben Sie Namen und Instanz des SQL Servers für Recognition an." There is a text input field for "SQL Server Name (optional Instanzname)" containing ".\". Below this is a checkbox labeled "SQL Benutzer für Datenbankverbindung nutzen." which is currently unchecked. Underneath the checkbox are two text input fields: "Benutzername:" and "Passwort:". At the bottom left, there is a copyright notice: "© Brainware Consulting & Development AG". At the bottom right, there are three buttons: "< Zurück", "Weiter >" (which is highlighted with a blue border), and "Abbrechen".

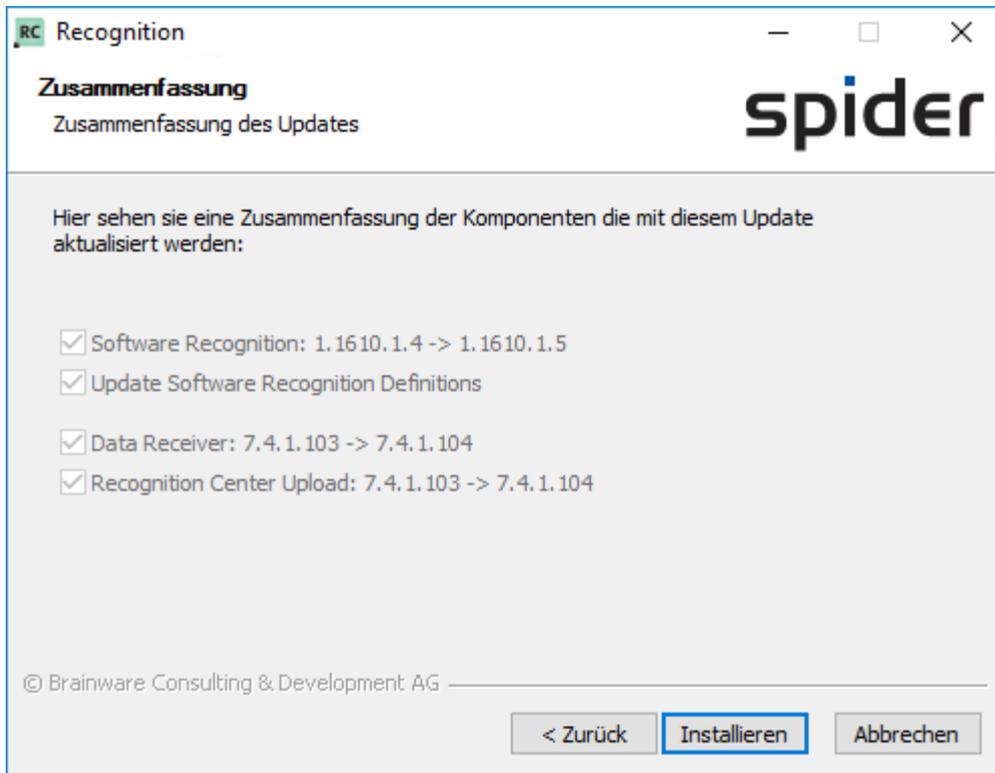
Der Benutzer, mit dem die Verbindung hergestellt wird, kann entweder der angemeldete (Windows-)Benutzer sein, oder es kann ein SQL Benutzer verwendet werden.

Soll ein SQL Benutzer verwendet werden, ist das Kontrollkästchen **SQL Benutzer für Datenbankverbindung nutzen** zu markieren sowie die Felder **Benutzername** und **Passwort** mit den gewünschten Daten auszufüllen.

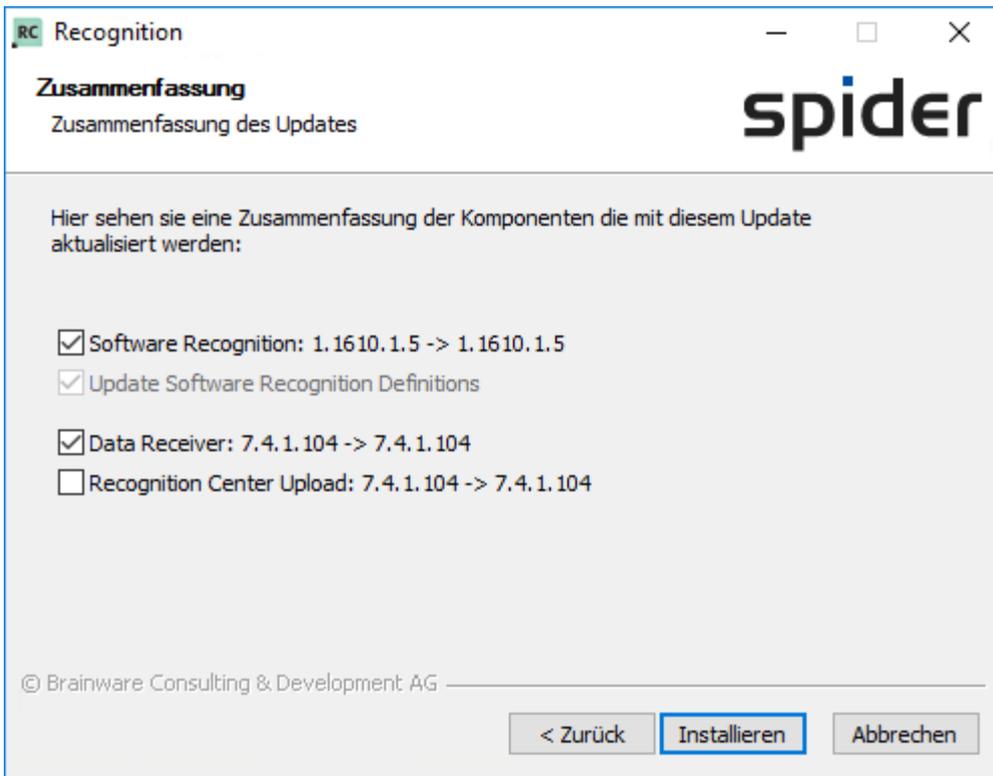
Feld	Beschreibung
SQL Server Name	Name des SQL Servers, nicht veränderbar!
SQL Benutzer für Datenbankverbindung nutzen	Auswahl, ob der angemeldete Domainbenutzer verwendet wird oder ob ein SQL Benutzer verwendet werden soll.
Benutzername	Name des SQL Benutzers.
Passwort	Passwort des SQL Benutzers.

## 5.4.6 Zusammenfassung

Erfolgt die Aktualisierung zum ersten Mal, müssen alle Produkte zur Aktualisierung ausgewählt werden. Der Dialog ist entsprechend vorausgefüllt und lässt sich nicht mehr ändern.

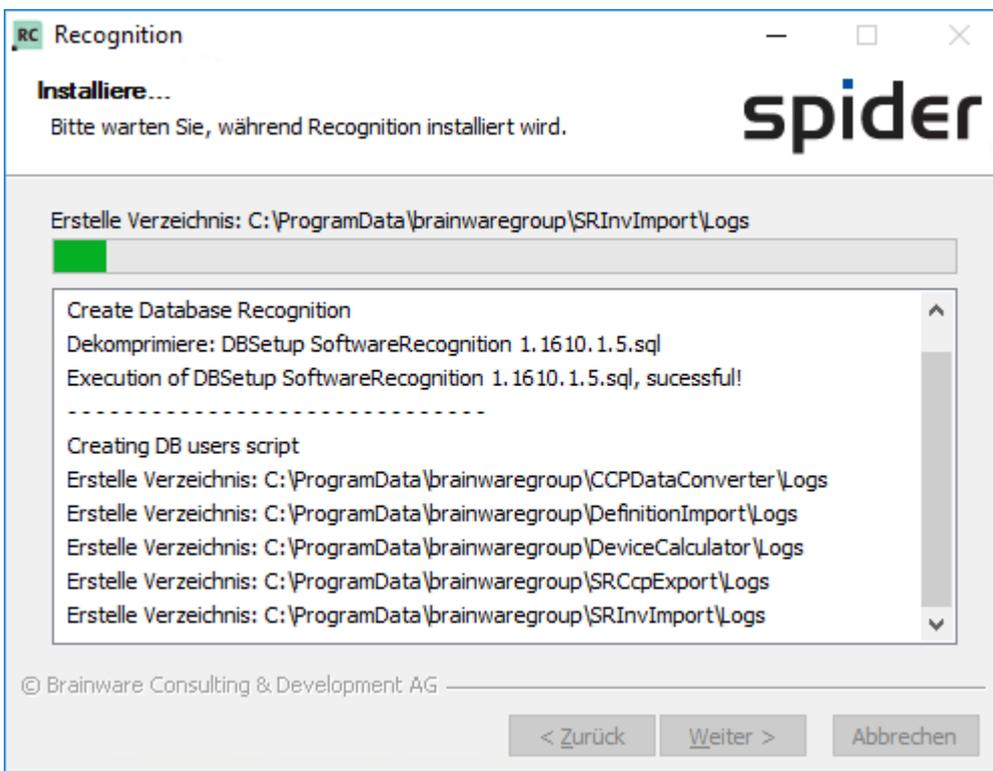


Falls eines der Produkte schon vorher aktualisiert wurde, kann es sein, dass nicht mehr alle Produkte zur Aktualisierung angeboten werden. Diese können dann trotzdem zur Aktualisierung ausgewählt werden.

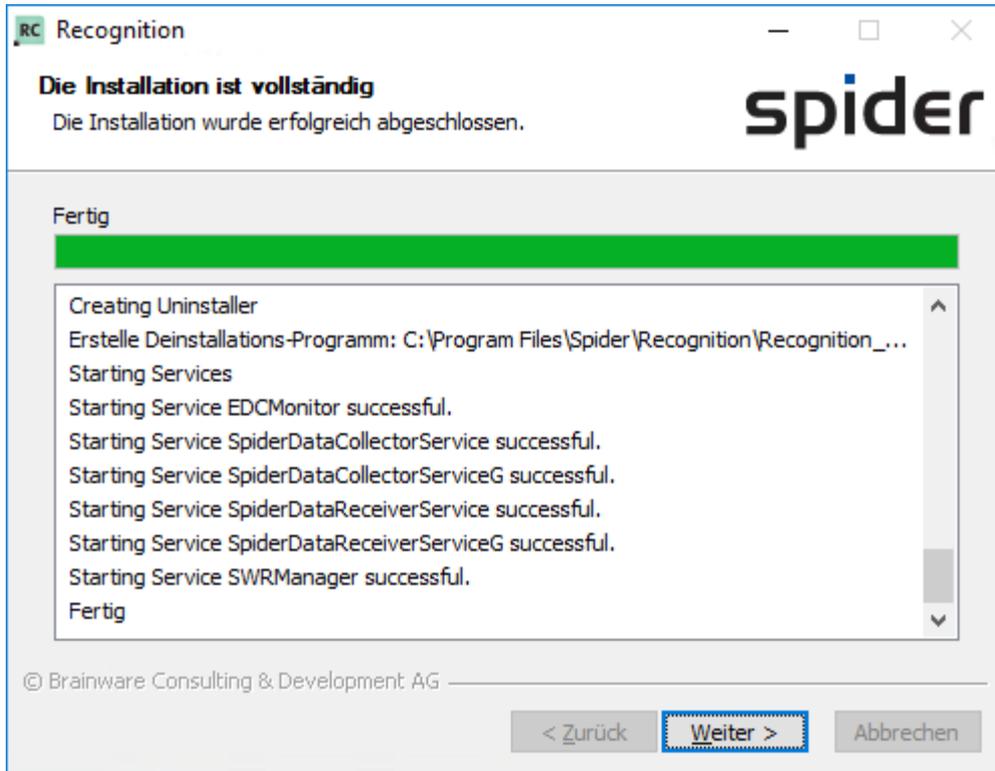


## 5.4.7 Installation

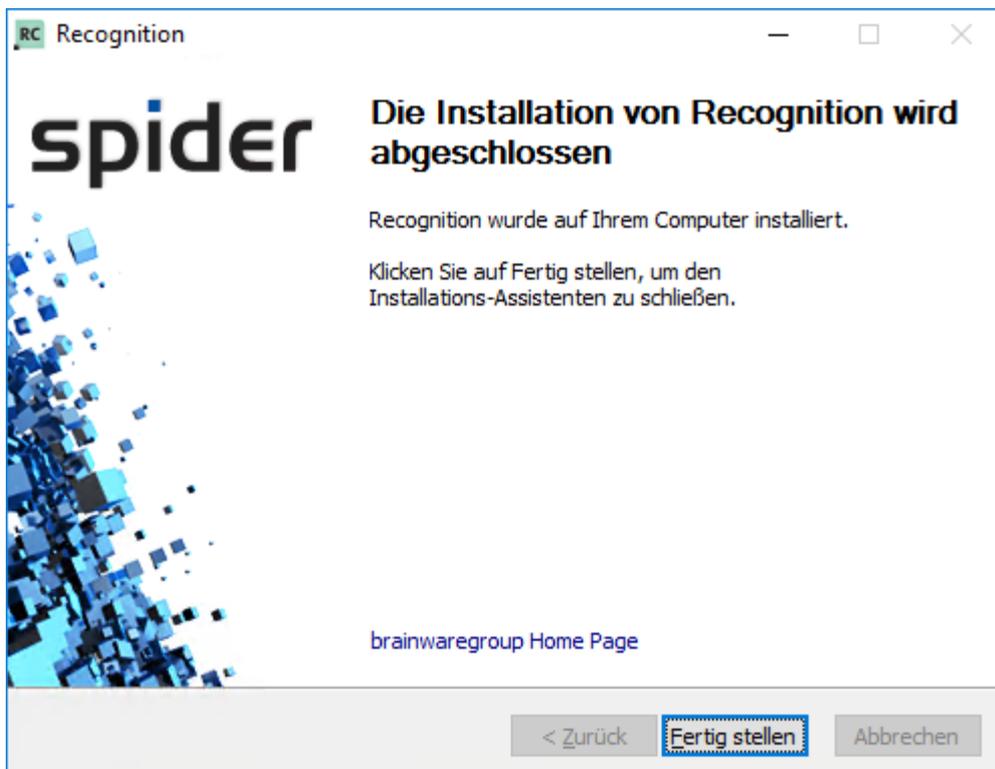
Der Installationsassistent installiert nun Recognition. Dies kann einige Zeit in Anspruch nehmen. Bitte warten Sie, bis die Installation beendet ist.



Nach erfolgreichem Abschluss der Installation, wird dies im Fortschrittsdialog angezeigt.



Das Setup ist erfolgreich abgeschlossen und kann beendet werden



Nach Abschluss der Installation wird ein Logfile geschrieben. Details zu den Logfiles, sind im Kapitel [Logfiles](#) (siehe Seite 210) beschrieben.

## Datacenter-Modul (novaratio Appliance)

### 6.1 Sizing der virtuellen Maschine

Das Sizing der virtuelle Maschine ist von vielen Parametern abhängig. Als Orientierung können die folgenden Angaben dienen, welche vier Einsatzszenarien unterscheiden:

- **TEST:**  
Nutzung der Datacenter Appliance als temporäres Testsystem
- **SMALL:**  
Nutzung der Datacenter Appliance in einer Umgebung mit bis zu 1.000 Systemen
- **MEDIUM:**  
Nutzung der Datacenter Appliance in einer Umgebung mit 1.000 bis 5.000 Systemen
- **LARGE:**  
Nutzung der Datacenter Appliance in einer Umgebung mit 5.000 bis 10.000 Systemen

Für Umgebung mit mehr als 10.000 Systemen können keinen pauschalen Vorgaben gemacht werden.

Für diese Einsatzszenarien werden jeweils die folgenden Angaben gemacht:

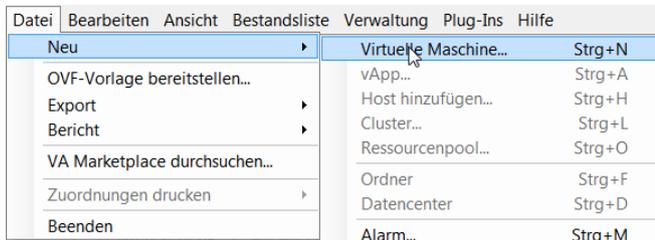
- **# CPU Threads:** Anzahl der CPU-Threads für die Appliance
- **Memory RAM:** Hauptspeicher für die Appliance
- **Plattenplatz:** Größe des Filesystems der Appliance
- **Netzwerk:** Anzahl Netzwerkinterfaces

Sizing	TEST	SMALL	MEDIUM	LARGE
Anzahl Systeme		bis 1.000	1.000 bis 5.000	5.000 bis 10.000
# CPU Threads	2	4	12	16
Memory RAM in GB	4	10	20	32
Filesystemgröße in GB	150	250	500	750
# Gigabit Netzwerkinterfaces	1	2	2	2

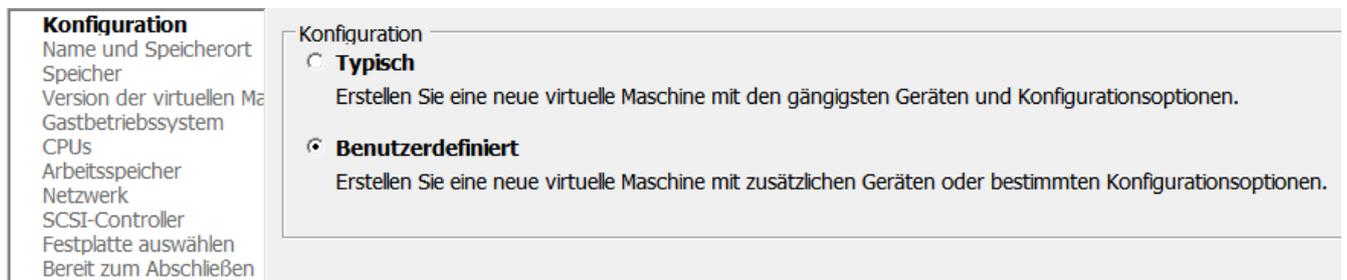
## 6.2 Inbetriebnahme der Appliance

### 6.2.1 Importieren der VM

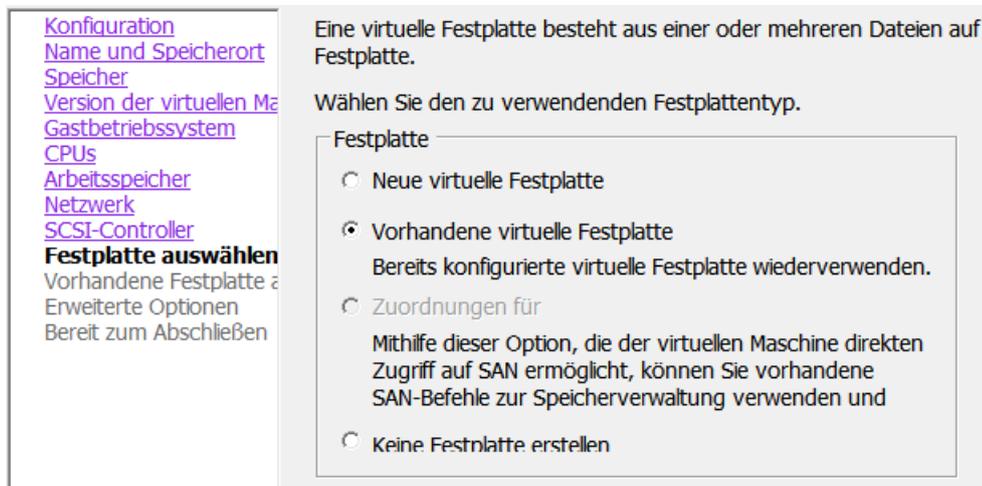
Die virtuelle Maschine wird als Appliance für VMware zur Verfügung gestellt. Nachfolgend wird kurz erläutert, wie die virtuelle Maschine in eine bestehende VCenter Umgebung mittels vSphere Client eingebunden werden kann. Hierzu klickt man zunächst über Datei -> Neu auf „Virtuelle Maschine...“



Anschließend wählt man als Konfiguration „Benutzerdefiniert“ aus.



Die nachfolgenden Einstellungen sind selbsterklärend. Wichtig ist, dass bei „Festplatte auswählen“ der Punkt „Vorhandene virtuelle Festplatte“ ausgewählt wird.



Die zuvor auf dem eingerichteten Datastore abgelegt VM kann nun nach einem Klick auf „Weiter“ ausgewählt werden. Für den Import per Web Client bietet VMware einen eigenen KB-Artikel inklusive Videoanleitung unter folgender Adresse an: [https://kb.vmware.com/selfservice/microsites/search.do?language=en\\_US&cmd=displayKC&externalId=2032904](https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2032904)

## 6.2.2 Betriebssystem User

Für den Betrieb der Appliance und der Applikation wurden Benutzer auf Betriebssystemebene angelegt:

User	Passwort
erunbook	erunbook
root	nova123

Die Appliance wird standardmäßig mit dem Hostname „vm-spider-inv“ ausgeliefert. Die Netzwerkeinstellungen sind so konfiguriert, dass dem Server per DHCP eine IP Adresse zugewiesen wird. Nachfolgend wird kurz dargestellt, wie diese Einstellungen für kundenspezifische Umgebungen angepasst werden können.

## 6.2.3 Anpassung des Hostnamens

Die Änderung des Hostnames der Appliance erfolgt mit dem Befehl „hostname“. Der Aufruf erfolgt als Benutzer root und wirkt sofort, allerdings nur bis zum nächsten Neustart der Appliance:

```
root@vm-spider-inv:~# hostname <neuer hostname>
```

Damit die Anpassung des Hostnamens über den Neustart hinaus wirksam bleibt, muss zusätzlich die Datei /etc/hostname als root-Benutzer mit einem Texteditor angepasst werden. Dies kann zum Beispiel mit dem Editor vi erfolgen:

```
root@vm-spider-inv:~# vi /etc/hostname
<neuer hostname>
:x
root@vm-spider-inv:~# _
```

## 6.2.4 Anpassung der IP-Adresse

Die IP-Adresse der Appliance ist standardmäßig auf die Nutzung von DHCP eingestellt. Für die Nutzung einer statischen IP-Adresse müssen als root-Benutzer in der Datei /etc/network/interfaces Anpassungen vorgenommen werden:

```
root@vm-spider-inv:~# vi /etc/network/interfaces
```

Im Auslieferungszustand der Appliance sieht der Abschnitt „The primary network interface“ wie folgt aus:

```
# The primary network interface
auto eth0
iface eth0 inet dhcp
#iface eth0 inet static
#address 172.19.120.99
#netmask 255.255.255.0
#gateway 172.19.120.1
#dns-nameservers 172.19.120.1
```

Für die Nutzung einer statischen IP-Adresse muss die DHCP-Zeile auskommentiert werden. Die Zeilen „iface“, „adresse“, „netmask“, „gateway“ und „dns-nameservers“ müssen aktiviert und mit den gewünschten Werten befüllt werden.

Im folgenden Beispiel soll die Appliance die IP-Adresse 172.19.120.99 nutzen. Das Routing erfolgt über das Gateway 172.19.120.1, die Subnetzmaske ist auf 255.255.255.0 gesetzt und als DNS-Server wird das System mit der IP-Adresse 172.19.120.1 genutzt:

```
# The primary network interface
auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
address 172.19.120.99
netmask 255.255.255.0
gateway 172.19.120.1
dns-nameservers 172.19.120.1
```

Nach den Änderungen an dieser Datei müssen die Netzwerkdienste der Appliance neugestartet werden:

```
root@vm-spider-inv:~# systemctl restart ifup@eth0
```

Nach diesem Befehl sollte mit dem Befehl ifconfig überprüft werden, ob die Einstellungen korrekt übernommen wurden.

```
root@vm-spider-inv:~# ifconfig
...
```

## 6.2.5 Erzeugen eines Zertifikats

Das Tool, um ein eRunbook Zertifikat zu generieren oder zu aktualisieren befindet sich bereits auf dem eRunbook Server. Hierfür navigiert man auf dem eRunbook Server in das Verzeichnis /opt/eRunbook/product/tools und startet anschließend als User erunbook das Skript eRunbook\_gen\_ssl\_certificate.sh.

Für die Erstellung des Zertifikates werden einige Informationen abgefragt. Hierbei ist darauf zu achten, dass für die Abfrage des „Common Name“ (in der Abbildung Fett hervorgehoben) der FQDN des eRunbook Servers erwartet wird.

```
erunbook@vm-dev-mj:~ > cd /opt/eRunbook/product/tools
erunbook@vm-dev-mj:~/product/tools ./eRunbook_gen_ssl_certificate.sh

writing RSA key
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:Germany
Locality Name (eg, city) []:Koblenz
Organization Name (eg, company) [Internet Widgits Pty Ltd]:nova ratio AG
Organizational Unit Name (eg, section) []:eRunbook Dev
Common Name (eg, YOUR name) []:vm-dev-mj.novaratio
Email Address []:.

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:.
An optional company name []:.
```

```
Signature ok
subject=/C=DE/ST=Germany/L=Koblenz/O=nova ratio AG/OU=eRunbook Dev/CN=vm-dev-mj.novaratio
Getting Private key
```

```
Please restart Webserver now
erunbook@vm-dev-mj:~/product/tools >
```

Abschließend muss lediglich der Webbrowser neugestartet werden. Hierfür steht ein Skript bereit, mit dem der Apache Webserver gestartet werden kann. Dieses Skript befindet sich im Verzeichnis `/opt/eRunbook/product/admin/` und wird mit dem Parameter „restart“ aufgerufen:

```
erunbook@vm-dev-mj:~/product/admin > cd /opt/eRunbook/product/admin/
erunbook@vm-dev-mj:~/product/admin > ./eRunbook_apache restart
erunbook@vm-dev-mj:~/product/admin >
```

Das neu erstellte eRunbook Zertifikat wird mit dem Namen „`httpd.server.crt`“ angelegt und befindet sich im Verzeichnis `/opt/eRunbook/software/additional/etc`

Das auf diese Weise erstellte Zertifikat ist ein „Self-Signed SSL Certificate“. Das bedeutet, dass die Webbrowser beim Zugriff auf Seiten dieses Webserver in der Regel nach einer Bestätigung fragen.

## 6.3 Einrichten des Datacenter-Moduls

### 6.3.1 Allgemeiner Hinweis

Bei den folgenden Konfigurationsschritten werden z.T. direkte Aktionen über ein API auf dem Datacenter-Module ausgeführt. Die Einrichtung bzw. Löschung eines Mandanten aus Spider resultiert in mehreren Aktionen im Datacenter-Modul. Je nach Ausstattung und Auslastung des Datacenter-Moduls kann ein einzelner, im Hintergrund laufender Prozessschritt mehrere Minuten in Anspruch nehmen.

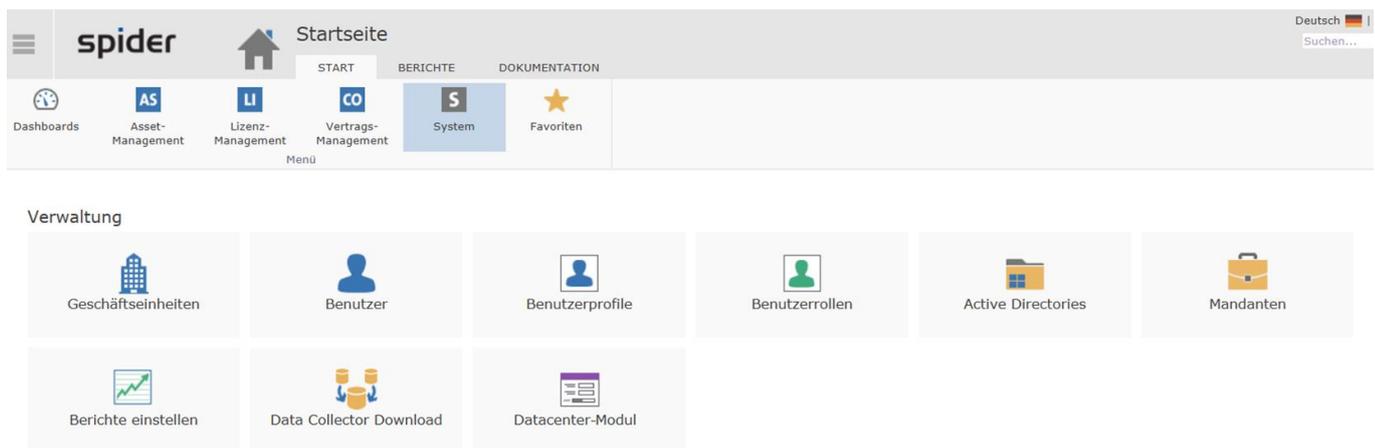
Über das Columbus Datacenter Inventory kann die Multi Platform Inventarisierung erfolgen. Die Lieferung der Inventardaten kann wahlweise an einen Spider Data Collector (SDC) Endpunkt oder einen Columbus Infrastrukturserver erfolgen. Von hier aus werden die Inventarisierungsdateien als Columbus Datacenter Container Datei (cdc-Datei) an das Recognition Modul geliefert.

Nachdem die Einrichtung des Datacenter-Moduls erfolgt ist, können mit der nächsten Datenlieferung die \*.cdc – Dateien verarbeitet werden. Das erfolgt i.d.R. einmal am Tag und nimmt je nach Umfang der untersuchten Systemlandschaft einige Zeit in Anspruch.

### 6.3.2 Registrierung

Um die eingehenden Rohdateien im \*.cdc – Format verarbeiten zu können, muss der Zugriff auf ein Datacenter-Modul (novaratio Appliance) konfiguriert sein. Dabei handelt es sich um die technischen Verbindungsparameter einer API.

Dafür steht der Menüpunkt Datacenter-Modul im Hauptmenü im Bereich System zur Verfügung.



Beim Klicken auf diesen Menüpunkt öffnet sich eine Verwaltungsseite, über die die Einrichtung eines Datacenter-Moduls erfolgen kann. Typischer Weise wird nur ein Datacenter-Modul eingerichtet. In Mehrmandantensystemen kann es aus Gründen der Lastverteilung erforderlich sein, mehrere Datacenter-Module einzurichten.

Über den Ribbon-Button „Anlegen“ kann ein Datacenter-Modul eingerichtet werden. Neben einem frei zu vergebenden Namen zur internen Orientierung muss die URL zur API sowie die Zugangsdaten des technischen Benutzers eingegeben werden.

The screenshot shows the Spider web interface for the 'Datacenter-Modul'. At the top, there is a navigation bar with the 'spider' logo, the title 'Datacenter-Modul', and a 'START' button. On the left, there is a ribbon with an 'Anlegen' button. Below the ribbon, there are two main sections: 'Datacenter-Modul-Liste' and 'Datacenter-Modul-Details'. The 'Liste' section shows a table with one entry: 'Test5'. The 'Details' section is a form with the following fields: Name, API URL, API Systembenutzer, and Passwort API Systembenutzer.

Nach dem Registrieren wird die Detailseite zum Datacenter-Modul angezeigt. Im linken Bereich werden eingerichtete Datacenter-Module angezeigt. Die Details zum ersten existierenden Datacenter-Module werden beim Öffnen der Seite automatisch geladen. Für das aktuell ausgewählte Datacenter-Modul kann mit der Schaltfläche „Test“ unmittelbar die Konnektivität geprüft werden.

The screenshot shows the Spider web interface for the 'Datacenter-Modul' after registration. The navigation bar is the same. The ribbon now includes buttons for 'Neu', 'Speichern', 'Löschen', and 'Testen'. The 'Datacenter-Modul-Liste' section shows the 'Test5' entry selected. The 'Datacenter-Modul-Details: "Test5"' section is a form with the following fields: ID (intern), Name, API URL, API Systembenutzer, Passwort API Systembenutzer, Angelegt von, Angelegt am, Bearbeitet von, Bearbeitet am, and Zugewiesene Mandanten.

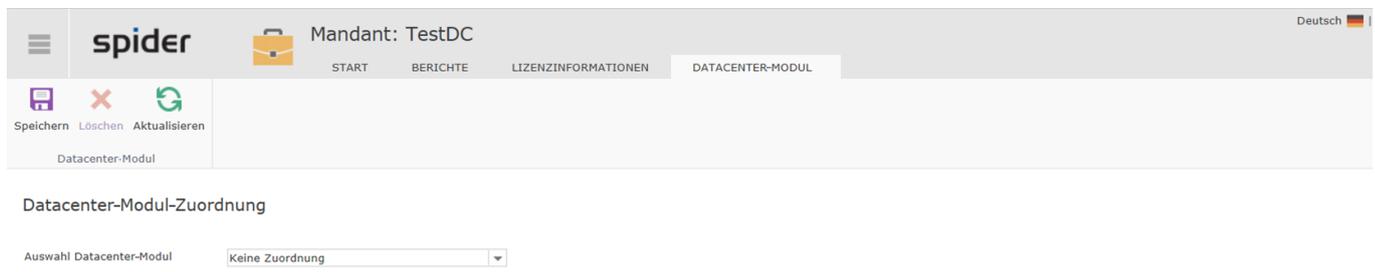
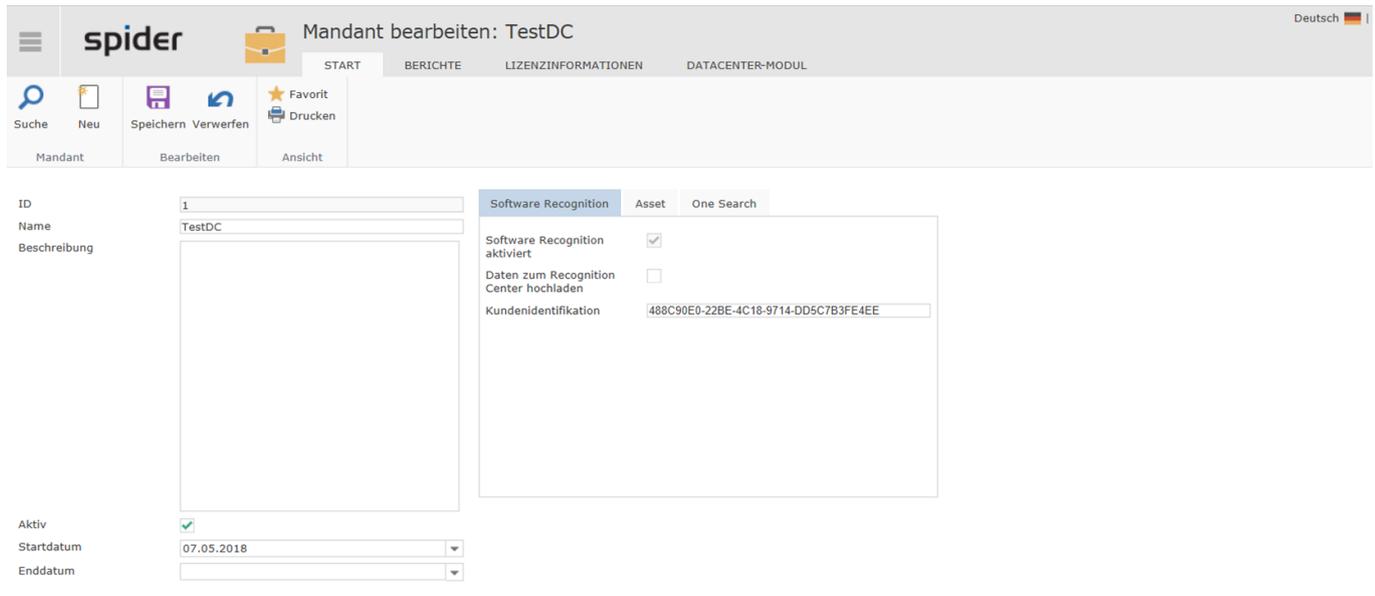
Datacenter-Modul-Details: "Test5"	
ID (intern)	7
Name	Test5
API URL	https://novaapphh.bwq.corp:9443/eRunbook/API/eAdmin
API Systembenutzer	eRunAdmin
Passwort API Systembenutzer	
Angelegt von	admin
Angelegt am	26.03.2018
Bearbeitet von	tf
Bearbeitet am	16.05.2018
Zugewiesene Mandanten	keine

### 6.3.3 Mandantenzuweisung

Ein Datacenter-Modul kann die Daten mehrerer Mandanten verarbeiten. Ein Mandant ist umgekehrt immer genau einem Datacenter-Module zugeordnet.

Demzufolge erfolgt die Zuweisung eines Datacenter-Modules im Kontext des betreffenden Mandanten.

Über den Register Datacenter-Modul wird die Einrichtungsseite geöffnet.



Ein zuvor eingerichteter Zugriff auf ein Datacenter-Modul steht über ein Auswahlfeld zur Verfügung. Die Auswahl ist mit einem Klick auf „Speichern“ zu bestätigen. Damit startet im Hintergrund die Einrichtung des Mandanten im Datacenter-Modul.



Die Mandanteneinrichtung ist abgeschlossen. Werden für den Mandaten Inventardaten fürs Datacenter-Modul angeliefert, werden diese automatisch zur Verarbeitung ans Datacenter-Modul weitergeleitet. In Spider werden nach einem Verarbeitungslauf die Ergebnisse angezeigt:

- Systeme
- Software aus der Datacenter Software Detection
- Oracle Daten

# DSGVO Anforderungen in Spider

---

## 7.1 Personenbezogene Daten

---

In der Begriffsbestimmung der Europäischen Datenschutz-Grundverordnung, im Artikel 4, Nr. 1 DSGVO ist folgende Definition zu finden:

... „**personenbezogene Daten** sind alle Informationen, die sich auf eine identifizierte oder identifizierbare natürliche Person (im Folgenden „betroffene Person“) beziehen; als identifizierbar wird eine natürliche Person angesehen, die direkt oder indirekt, insbesondere mittels Zuordnung zu einer Kennung wie einem Namen, zu einer Kennnummer, zu Standortdaten, zu einer Online-Kennung oder zu einem oder mehreren besonderen Merkmalen, die Ausdruck der physischen, physiologischen, genetischen, psychischen, wirtschaftlichen, kulturellen oder sozialen Identität dieser natürlichen Person sind, identifiziert werden kann.“

In Bezug auf die Spider-Produkte bedeutet dies, dass eine Vielzahl von Modulen, Prozessen und Berichten von der Schaffung einer DSGVO-Konformität betroffen sind. Daten, die hingegen keiner Anwendung der DSGVO bedürfen oder anderen Rechtsgrundlagen unterliegen, bleiben von den Änderungen unberührt. Hierzu zählen bspw. Daten juristischer Personen.

### 7.1.1 Welche Datenkategorien werden in Spider-Produkten verarbeitet?

---

Innerhalb der Spider-Produkte werden im Produktstandard folgende Datenkategorien verarbeitet:

- Personeninformationen (bspw. Name, Vorname),
- Kontaktinformationen (bspw. E-Mail-Adressen, Telefonnummern),
- Domain-Informationen (bspw. Domain-Login, Active Directory-Kennungen, Last-Login-Benutzer, Hauptbenutzer oder IP-Adressen)
- Und teilweise Betriebsangehörigkeitsinformationen (bspw. Unternehmenseintritt, -austritt oder Personalnummern)

Die konkret vorliegenden Datenfelder für personenbezogene Daten sind bei **individuellen Anpassungen kundenseitig** zu prüfen und zu dokumentieren. Dies gilt insbesondere für die Möglichkeit zusätzlicher Datenkategorien.

### 7.1.2 Besondere Kategorien personenbezogener Daten (Artikel 9 Abs. 1 DSGVO)

---

Ergänzend zu den oben genannten Daten spezifiziert die DSGVO im Artikel 9 Abs. 1 besondere Kategorien personenbezogener Daten, deren Verarbeitung im Grundsatz untersagt ist.

**Im Standard sind in Spider-Produkten diese besonderen Datenkategorien nicht enthalten.**

### Kundenindividuelle Erweiterungen prüfen!

---

Es ist möglich, dass im Rahmen von **kundenindividuellen Anpassungen** personenbezogene Daten besonderer Kategorien zum Gegenstand der Spider-Produkte geworden sind. Dies sind bspw. zusätzliche Felder oder die Einbindung von Drittsystemen mit personenbezogenen Daten besonderer Kategorien. Hier bedarf es einer **kundenindividuellen Betrachtung, Dokumentation und Definition erforderlicher Maßnahmen**, bspw. bei der Verarbeitung von personenbezogener Daten Minderjähriger oder das Verarbeiten zusätzlicher Informationen wie Geburtsdatum oder Konfes-

sion. Insbesondere Kunden, die spezifische genetische, biometrische oder anderweitige Gesundheitsdaten bzw. -merkmale mit unseren Produkten erheben, speichern oder verarbeiten, sind hiervon betroffen.

### 7.1.3 Verarbeitung personenbezogener Daten (Artikel 4 Nr. 2 DSGVO)

---

Im Rahmen von Risiko- und Compliance-Prüfungen ist ggf. Auskunft über die Verarbeitung von personenbezogenen Daten in Spider-Produkten zu erteilen. Eine Frage nach einer entsprechenden Verarbeitung ist zu bejahen. Die Begriffsbestimmung gemäß Artikel 4 Nr. 2 DSGVO besagt eindeutig:

*„**Verarbeitung**“ ist jeder mit oder ohne Hilfe automatisierter Verfahren ausgeführte Vorgang oder jede solche Vorgangsreihe im Zusammenhang mit personenbezogenen Daten wie das Erheben, das Erfassen, die Organisation, das Ordnen, die Speicherung, die Anpassung oder Veränderung, das Auslesen, das Abfragen, die Verwendung, die Offenlegung durch Übermittlung, Verbreitung oder eine andere Form der Bereitstellung, den Abgleich oder die Verknüpfung, die Einschränkung, das Löschen oder die Vernichtung.*

In Spider-Produkten werden personenbezogene Daten definierter Datenkategorien verarbeitet. Ohne eine Verarbeitung dieser Daten wären Spider-Produkte nicht funktionsfähig oder würden ihren Zweck erheblich einschränken.

## 7.2 Personenbezogene Datenfelder an ausgewählten Spider Datenobjekten

---

Die folgenden Kapitel zeigen die konkreten Felder der Datenobjekte an denen personenbezogene Daten gespeichert werden. Da das Spider Datenmodell über Parameterdefinition individuell ergänzt werden kann, handelt es sich um die Standardauslieferung, die nur bei Full-Update-Editionen nicht verändert werden kann. Bei Flex-Editionen ist die Prüfung und Dokumentation immer kundenindividuell zu erfolgen.

### 7.2.1 Personenbezogene Daten am Mitarbeiter (Employee)

---

Mitarbeiter Stammdaten werden in der Regel aus Drittsystemen in Spider importiert. Als häufigstes System ist Microsoft Active Directory genannt.

Folgende Datenkategorien werden am Mitarbeiter Datenobjekt gespeichert:

- Personeninformationen (bspw. Name, Vorname)
- Kontaktinformationen (bspw. E-Mail-Adresse, Telefonnummer)
- Domain-Informationen (bspw. Domain-Login, Active Directory-Kennungen, Hauptbenutzer)
- Teilweise Betriebsangehörigkeitsinformationen (bspw. Unternehmenseintritt, -austritt)

Folgende Felder werden beim Employee ausgeliefert. Der Umfang ist auf das notwendige Maß reduziert worden. Hauptsächlich geht es um die Anzeige des Names eines Mitarbeiters, sowie dessen Identifikation beim Import oder die Identifikation und die Verbindung beim technischen Inventar von Systemen über Domainkennzeichen. Da häufig Daten aus Microsoft Active Directory angebunden werden und die Schnittstelle und Datenintegration im Produktstandard enthalten sind, werden die Feldnamen aus Microsoft Active Directory mit angegeben. Über den Active Directory Connector können die Spalten, die exportiert werden sollen, eingestellt werden. Im Standard sind nur die folgenden Felder aktiviert:

- LastName (Active Directory: GivenName)
- Firstname (Active Directoy: Surname)
- DomainLogin (Active Directory: SAMAccountName)
- Fullname
- EmailAddress (Active Directory: EmailAddress)
- EmploymentTypeID – Kategorisierung bspw. intern/extern

- UserPrincipalName (Active Directory: UserPrincipalName)
- swrADUserObjectID – interne ID in Recognition DB
- ExternalReference – Identifikation in einem Drittsystem

Folgende Felder werden nicht länger in der Standardkonfiguration ausgeliefert und beim Update von Full-Update Editionen auf Spider 6.4 entfernt. Die Felder bleiben in der Datentabelle bestehen. Jegliche Berechtigung und Sichtbarkeit wurden entfernt. Sollte ein kundenindividueller Verwendungszweck bestehen, der das Verarbeiten der Daten rechtfertigt, können die Felder in der Flex-Edition eingeblendet werden:

- Title
- StaffNo (Active Directory: StaffNo)
- PhoneNo (Active Directory: PhoneNo)
- MobilePhoneNo (Active Directory: MobilePhoneNo)
- DateOfEntry
- DateOfEmission
- FaxNo
- PrivatePhoneNo
- Department
- Location
- Country
- CountryCode
- State
- StreetAddress
- PostalCode

Über OneSearch wird eine indexbasierte Suche abgebildet. Mitarbeiterdaten werden indiziert. OneSearch steht für alle Spider Benutzer zur Verfügung – eine Berechtigungseinschränkung auf Felder oder Datensätze ist nicht möglich. Daher werden zukünftig ausschließlich folgende Felder für das Suchen und Anzeigen über OneSearch indiziert:

- LastName (Active Directory: GivenName)
- Firstname (Active Directory: Surname)
- DomainLogin (Active Directory: SAMAccountName)

Darüber hinaus wird das folgende Feld indiziert. Dies aus Gründen der modulübergreifenden Integration.

- EmailAddress (Active Directory: EmailAddress)

## 7.2.2 Personenbezogene Daten an Assets

---

Assetdaten werden in der Regel aus Inventarsysteme in Spider verarbeitet und aktuell gehalten. Bestimmte Felder können direkt personenbezogene Daten enthalten (bspw. LastLoggedOnUser), andere Felder können technische Bezeichner enthalten, über die auf natürliche Personen rückgeschlossen werden könnte.

Folgende Datenkategorie werden am Asset Datenobjekt gespeichert:

- Domain-Informationen (bspw. LastLoggedOnUser, Hostname, IP-Adressen, UUID)

Bei den folgende Felder handelt es sich um technische Bezeichner oder Identifikationsfelder, die für die Verarbeitung und Identifikation benötigt werden. Sie werden benötigt und können nicht deaktiviert werden.

- Hostname
- AssetNo
- International Mobile Station Equipment Identity (IMEI)
- Rufnummer (PhoneNo)
- UUID
- URN
- ID

Folgende Felder enthalten personenbezogene Domaininformationen aus Benutzerkonten.

- LastLoggedOnUser
- PrimaryUser (ermittelter Hauptbenutzer)

Über den LastLoggedOnUser kann auf ein Microsoft Active Benutzerkonto rückgeschlossen werden, welches mit einer natürlichen Person verknüpft ist. Dies wird in Spider aufgelöst. Benötigt werden diese Felder aus mehreren Gründen. Wichtig zu nennen sind die Anforderungen zur Abbildung von userbasierten Lizenzmetriken, oder auch die Notwendigkeit zur Ermittlung von Hauptbenutzern auf einem Gerät. Sie werden benötigt und können in Spider nicht deaktiviert werden. Werden hingegen keine LastLoggedOnUser Daten durch ein anlieferndes System zur Verfügung gestellt, können keine userbasierten Lizenzabbildungen erfolgen.

Die folgenden Felder enthalten zusätzliche technische Angaben, für die im Standard kein konkreter Verwendungszweck vorliegt. Diese Felder werden nicht länger in der Standardkonfiguration ausgeliefert und beim Update von Full-Update Editionen auf Spider 6.4 entfernt. Die Felder bleiben in der Datentabelle bestehen. Jegliche Berechtigung und Sichtbarkeit wurden entfernt. Sollte ein kundenindividueller Verwendungszweck bestehen, der das Verarbeiten der Daten rechtfertigt, können die Felder in der Flex-Edition eingeblendet werden:

- IPAddressV4
- IPAddressV6
- MacAddress 1-4

Über OneSearch wird eine indexbasierte Suche abgebildet. Assetdaten werden indiziert. OneSearch steht für alle Spider Benutzer zur Verfügung – eine Berechtigungseinschränkung auf Felder oder Datensätze ist nicht möglich. Folgende Felder werden für das Suchen und Anzeigen über OneSearch indiziert:

- Hostname
- AssetNo
- UUID

## 7.2.3 Personenbezogene Daten am Spider Benutzer (User)

---

Für den Zugriff auf Daten in Spider ist ein Spider Benutzerkonto (User) erforderlich. Die Benutzerkonten können über ein Berechtigungskonzept an Funktionen und Daten in Spider berechtigt werden.

Folgende Datenkategorien werden am Mitarbeiter Datenobjekt gespeichert:

- Personeninformationen (bspw. Name)
- Kontaktinformationen (bspw. E-Mail-Adresse)
- Domain-Informationen (bspw. Login)

Folgende personenbezogene Datenfelder sind am User-Objekt vorhanden und werden für die Identifikation, Anzeige und Benutzung in Spider benötigt

- Login
- RealName

Zudem empfiehlt sich die Speicherung der E-Mail Adresse. Für die Verwendung der Passwort-vergessen-Funktion bei Spider internen Benutzerkonten ist die E-Mail Adresse zwingend erforderlich.

- EmailAddress

Es können Spider internen Benutzerkonten eingerichtet werden, oder ein Microsoft Active Directory angebunden werden, über das Benutzerkonten automatisch übernommen werden, die Mitglieder in dafür vorgesehenen ADGruppen sind. Bei Verwendung von Benutzerkonten aus Microsoft Active Directory wird als LoginName der SAMAccountName verwendet. Bei Spider internen Benutzerkonten empfiehlt sich die Verwendung von automatisch generierten Login-Kennungen, um einen Rückschluss auf eine natürliche Person zu erschweren (Pseudonym).

Der Login wird bei der in Spider an folgenden Stellen protokolliert:

- Login wird je Datenobjekt gespeichert, welches von einem Benutzer angelegt oder geändert worden ist.
- Login wird in den Änderungshistorien zu einer Änderung gespeichert, die der Benutzer initiiert hat.

Darüber hinaus kann zu einem Spider Benutzerkonto ein Mitarbeiterstammdatensatz verknüpft werden. Handelt es sich um ein Active Directory Konto, kann der Mitarbeiterstammdatensatz automatisch verknüpft werden.

## 7.2.4 Personenbezogene Daten an den Kontaktperson (Contact Person)

---

Im Vertragsmanagement können sog. Kontaktpersonen zu Vertragspartnern manuell erfasst werden. Bei den Kontaktpersonen handelt es sich um natürliche Personen. Die Datenverarbeitung dieser Daten unterliegt den Anforderungen der DSGVO. Eine Speicherung der Daten darf nur mit Einwilligung oder vertraglicher Grundlage erfolgen. Für die Einhaltung der Anforderungen ist kundenseitig bspw. in den begleitenden Prozessen Sorge zu tragen. Wenn dies nicht sichergestellt werden kann, sollte der Bereich vollständig deaktiviert werden.

Folgende Datenkategorien werden über eine Kontaktperson gespeichert:

- Personeninformationen (bspw. Name, Vorname)
- Kontaktinformationen (bspw. E-Mail-Adresse, Telefonnummer)
- Organisationsinformationen (bspw. Funktion)

Der vorgesehene Verwendungszweck entspricht der Vereinfachung der Kontaktaufnahme zuständiger Ansprechpartner bei einem Vertragspartner.

Da über das Spider Meta-Daten-Modell beliebige Felder ergänzt werden können, die personenbezogene Daten enthalten können, kann in diesem Dokument nur der Umfang der Felder dokumentiert werden, die im Standardumfang ausgeliefert werden. Bei den Full-Update Systemen, wo die vollständige Konfiguration aktualisiert wird und keine zusätzlichen Felder ergänzt werden können, entspricht der Stand dem Standardumfang. Bei allen anderen Systemen ist kundenindividuell die konkrete Parametrisierung zu dokumentieren.

Folgende Felder werden im Rahmen der Standardkonfiguration ausgeliefert:

- LastName
- FirstName
- Title
- Phone

- Fax
- Email
- MobilePhoneNo

Über OneSearch wird eine indexbasierte Suche abgebildet. Daten von Kontaktpersonen werden nicht indiziert.

## 7.3 Grundsätze für die Verarbeitung personenbezogener Daten (Artikel 5 DSGVO)

---

Die Spider-Produkte werden für unterschiedliche Geschäftsprozesse eingesetzt. Die Erfüllung der Anforderungen der DSGVO ist von mehreren Faktoren abhängig. Folgende häufige Faktoren seien beispielhaft genannt:

- Art und Ausprägung der begleitenden oder zugrundeliegenden Geschäftsprozesse
- Umfang der kundenindividuellen Erweiterungen oder Ausprägungen im Einsatz der Spider-Produkte
- Grad der Einbindung von personenbezogenen Daten aus Fremdsystemen und der Gestaltung der Schnittstellen

Eine Betrachtung, die nur den Produktstandard umfasst, reicht nicht aus. Eine allgemeine Aussage zur Erfüllung der verschiedenen DSGVO-Anforderungen des Spider-Produktes in einer Kundenumgebung ist aufgrund der flexiblen Anpassungsmöglichkeiten **immer kundenindividuell** zu prüfen. Die Einschätzungen in diesem Dokument beziehen sich auf die grundlegenden Standardfunktionen in den Spider-Produkten.

Die Spider-Produkte werden die verschiedenen DSGVO-Anforderungen gemäß Artikel 5 technisch unterstützen. Neben der reinen Erfüllung der Anforderungen geht es auch um Einfachheit und das Erzielen zusätzlicher Vorteile, die mit der Umsetzung in den Produkten erreicht wird. Aus den Erfahrungen im Einsatz bei unseren Kunden werden identifizierte Anforderungen und Verbesserungen aufgenommen und in zukünftigen Produktversionen realisiert.

### 7.3.1 Angemessenheit und Erheblichkeit von personenbezogenen Daten, sowie die Beschränkung derer Verarbeitung auf das notwendige Maß (Artikel 5, Nr. 1, lit. c DSGVO)

---

In den Produkten sollen nur personenbezogene Daten verarbeitet werden, die für die originären Geschäftsprozesse oder Funktionen erforderlich sind.

Bei kundenindividuellen Anpassungen oder Erweiterungen der Spider-Produkte sowie beim Anbinden von Drittsystemen (bspw. Microsoft Active Directory), die personenbezogene Daten liefern, sind diese Grundsätze individuell zu prüfen und sicherzustellen.

Die Standardkonfiguration wurde in Hinblick auf die Beschränkung auf das notwendige Maß überarbeitet. Speziell beim Mitarbeiter-Datenobjekt wurden mehrere Felder im Standard entfernt.

### 7.3.2 Angemessene Speicherung personenbezogener Daten, sowie deren mögliche Anonymisierung (Artikel 5, Nr. 1, lit. e DSGVO)

---

Für verwaltete personenbezogene Daten, die keine Abhängigkeit zu zuliefernden Systemen haben und die für den originären Geschäftsprozess nicht mehr als erforderlich gelten (bspw. Unternehmensaustritt eines Mitarbeiters), stehen Löschroutinen zur Verfügung. Funktionen zur Anonymisierung bspw. für statistische Zwecke stehen nicht zur Verfügung. Demnach sind die Daten physikalisch zu Löschen. Die Löschung der Daten ist kundenindividuell sicherzustellen.

## 7.3.3 Betroffenenrechte

---

### Auskunftsrecht (Artikel 15 DSGVO)

---

Betroffene Personen haben ein generelles Auskunftsrecht, wenn ihre personenbezogenen Daten verarbeitet werden. Das Auskunftsrecht kann mit den heutigen Produktversionen grundsätzlich erfüllt werden.

#### Auskunftsseite für Mitarbeiter (Employee)

Die personenbezogenen Daten von Betroffene (bspw. Beschäftigtendaten) können über eine Auskunftsseite eingesehen und ausgedruckt werden. Dazu wurde mit Spider 6.4 eine neue Seite (EmployeeEditGDPRInfo) ergänzt, die einzeln berechtigt werden kann. Über die Seite werden alle Felder des Mitarbeiterobjektes, sowie Assets des Mitarbeiters angezeigt.

#### Externe Kontaktpersonen (Contact Persons)

Erfasste Daten einer Kontaktperson können über das Detailformular eingesehen werden. Es ist sicherzustellen, dass der Benutzer, der die Felder für eine Auskunft zusammenstellt, an allen Feldern, die personenbezogenen Daten enthalten, berechtigt ist.

#### Auskunftsseite für Spider Benutzer

Spider Benutzer können eigenständig ihre verwalteten personenbezogenen Daten einsehen. Diese kann über die Benutzereinstellungen eingesehen werden.

### Berichtigung und Löschung personenbezogener Daten (Artikel 5, Nr. 1, lit. d DSGVO)

---

Grundsätzlich können personenbezogene Daten in Spider gepflegt, geändert und auch gelöscht werden.

Bei Daten, die aus zuliefernden Systemen verwendet werden, ist eine Änderung in Spider i.d.R. nicht möglich, da diese Daten regelmäßig überschrieben werden. Eine Änderung dieser Daten muss in den jeweiligen Quellsystemen erfolgen.

Das Löschen von personenbezogenen Daten, die aus Fremdsystemen angeliefert werden, kann in den Spider-Produkten nicht einheitlich gelöst werden. Bspw. können gesetzliche Löschfristen diese überlagern. Das automatische Löschen von Daten ist immer kundenseitig festzulegen und einzustellen.

Das Recht auf Berichtigung und das Recht auf Löschung werden in den folgenden Abschnitten separat behandelt.

### Recht auf Berichtigung (Artikel 16 DSGVO)

---

In Spider-Produkten bestehen grundsätzlich Berichtigungsmöglichkeiten für personenbezogene Daten. Diese können über das Berechtigungskonzept definierten Berechtigungsgruppen vorbehalten sein.

Bei Zulieferung von personenbezogenen Daten aus Drittsystemen sind die Berichtigungen im Drittsystem durchzuführen. Änderungen an personenbezogenen Daten gelten nur ab dem Zeitpunkt der Eingabe bzw. Übernahme. Änderungshistorien sind demnach ausgenommen.

### Recht auf Löschung (Artikel 17 DSGVO)

---

In Spider-Produkten bestehen grundsätzlich Löschmöglichkeiten für personenbezogene Daten. Diese können über das Berechtigungskonzept definierten Berechtigungsgruppen vorbehalten sein.

Das Löschen erfolgt in zwei Schritten: Beim Löschen eines Datenobjektes wird dieser aus den Datentabellen gelöscht und in einen Recycle Bin geschrieben. Der Recycle Bin ist ebenfalls als Tabelle in den Datenbanken vorhanden. Der Datensatz ist nicht mehr in den Datentabellen vorhanden, sehr wohl noch physikalisch in der Datenbank gespeichert. Die Bereinigung des Recycle Bin – dem physikalischen Löschen aus der Datenbank – erfolgt produktseitig. **Die Vorhaltezeit in Tagen kann eingestellt werden.**

Mit dem Bereinigen des Recycle Bins werden auch etwaige Änderungshistorien gelöscht. Datensätze mit personenbezogenen Daten gelten erst nach der Bereinigung des Recycle Bins als physikalisch gelöscht.

Kundenindividuell sind die Backup-Strategien und Vorhaltezeiten der Backups noch zu berücksichtigen.

### **Besonderheit bei personenbezogenen Daten aus angebundenen Drittsystemen**

Bei Zulieferung von personenbezogenen Daten aus Drittsystemen ist die Löschung im Drittsystem durchzuführen. Das Löschen von personenbezogenen Daten, die aus Fremdsystemen angeliefert werden, kann in den Spider-Produkten nicht einheitlich und damit nicht im Produktstandard gelöst werden. Bei der Löschung von personenbezogenen Daten sind u.a. gesetzliche Aufbewahrungsfristen zu berücksichtigen. Demnach ist abhängig von den eingebundenen Drittsystemen und der Art des Geschäftsprozesses die Umsetzung für das Löschen zu gestalten.

### **Recht auf Einschränkung der Verarbeitung (Artikel 18 DSGVO)**

---

Da eine Rechtmäßigkeit der Verarbeitung u.a. gemäß Artikel 6, Nr. 1, lit. f DSGVO vorausgesetzt wird und somit berechnete Gründe das Recht auf Einschränkung überwiegen dürften, sind keine dedizierten technischen Maßnahmen zu ergreifen.

### **Mitteilungspflicht (Artikel 19 DSGVO)**

---

Die Mitteilungspflicht bei Veränderung von personenbezogenen Daten sind in den begleitenden kundenseitigen Verarbeitungsprozessen (bspw. Account Management) zu berücksichtigen. Dies ist eine klare Empfehlung. Das jedes System eigenständig die Betroffenen (Beschäftigten) in einer Organisation über Änderungen an den personenbezogenen Daten informiert, insbesondere bei indirekter Nutzung dieser aus einem Quellsystem, dürfte eher störend für die Betroffenen und schlussendlich aufwendiger für die Organisation sein.

Sofern eine automatische Mitteilung aus dem Spider-System gewünscht wird, muss dies über eine kundenindividuelle Erweiterung und Einrichtung realisiert werden. Dazu kann das produktintegrierte Alert&Reporting System genutzt, was täglich dynamisch E-Mails verschickt, die auf vorbereiteten Datenbanksichten (Views) basieren. Die E-Mail-Empfänger können dabei dynamisch je Datensatz mitgegeben werden. Als Datengrundlage für die Auswertung von Änderungen können die Spider Änderungshistorien genutzt werden.

### **Recht auf Datenübertragbarkeit (Artikel 20 DSGVO)**

---

Eine Datenübertragbarkeit der personenbezogenen Daten ist nicht relevant, da Spider-Produkte ausschließlich für interne Unternehmensprozesse und -systeme eingesetzt werden. Sollten Daten übertragen werden, besteht ein programmtechnischer Zugriff auf die Daten über eine Rest API.

### **Widerspruchsrecht (Artikel 21 DSGVO)**

---

Da eine Rechtmäßigkeit der Verarbeitung u.a. gemäß Artikel 6, Nr. 1, lit. f DSGVO vorausgesetzt wird und somit berechnete Gründe das Recht auf Widerspruch überwiegen dürften, werden keine technischen Maßnahmen ergriffen.

## Geeignete Möglichkeiten zum Nachweis der Einhaltung der Anforderungen im Rechenschaftsfall (Artikel 5, Nr. 2 DSGVO)

---

Änderungen an personenbezogenen Daten können über Änderungsprotokolle eingesehen werden, für die separate Berechtigungen erteilt werden können.

## 7.4 Rechtmäßigkeit der Verarbeitung (Artikel 6 DSGVO)

---

Die Rechtmäßigkeit der Verarbeitung personenbezogener Daten in Spider-Produkten bedingt sich durch die Rechtsgrundlagen gemäß folgender Artikel:

### Artikel 6, Nr. 1, lit. c DSGVO

*"Die Verarbeitung ist zur Erfüllung einer rechtlichen Verpflichtung erforderlich, der der Verantwortliche unterliegt."*

### Artikel 6, Nr. 1, lit. f DSGVO

*„Die Verarbeitung ist zur Wahrung der berechtigten Interessen des Verantwortlichen oder eines Dritten erforderlich, sofern nicht die Interessen oder Grundrechte und Grundfreiheiten der betroffenen Person, die den Schutz personenbezogener Daten erfordern, überwiegen, insbesondere dann, wenn es sich bei der betroffenen Person um ein Kind handelt.“*

Beide Passagen beziehen sich im Schwerpunkt hierbei implizit auf Pflichten aus dem BGB, HGB, UWG und UrhG.

## 7.5 Datenschutz durch Technikgestaltung und durch datenschutzfreundliche Voreinstellungen (Artikel 25 DSGVO)

---

Spider-Produkte und eingesetzte Infrastrukturprodukte (bspw. Microsoft SQL Server) beinhalten verschiedene technische Funktionalitäten zur Unterstützung technischer und organisatorischer Maßnahmen. Die konkreten technischen und organisatorischen Maßnahmen sind kundenindividuell zu treffen.

### 7.5.1 Vertraulichkeit (Art 32 Abs. 1 lit. b DSGVO)

---

- **Zutrittskontrolle:** bspw. Kein unbefugter Zutritt zu Datenverarbeitungsanlagen.  
Kundenseitig sicherzustellen
- **Zugangskontrollen:** keine unbefugte Systembenutzung durch Kennwörter  
Kundenseitig sicherzustellen
  - Daten werden in Microsoft SQL Server Datenbanken verarbeitet
  - Bestimmte Daten werden auch im Index gespeichert (ESI)
- **Zugriffskontrolle:** Kein unbefugtes Lesen, Kopieren, Verändern oder Entfernen innerhalb des Systems
  - Der schreibende/ändernde Zugriff auf die Daten wird über ein rollenbasiertes Spider Berechtigungskonzept kontrolliert
  - Ein direkter Zugriff für Benutzer auf den SQL Server sollte daher unbedingt vermieden werden, da über diesen die Spider Berechtigungsprüfungen keine Anwendung finden
- **Trennungskontrolle:** getrennte Verarbeitung von Daten, die zu unterschiedlichen Zwecken erhoben wurden, z.B. Mandantenfähigkeit
  - Bei Multi-Mandantensystemen werden an jedem Datensatz eine Mandanten ID gespeichert. Beim Zugriff auf die Daten durch die Spider Businesslogik wird die Mandantentrennung durchgesetzt. Ein Benutzer kann nur auf die Daten aus einer aktuellen Mandantensitzung (Session) zugreifen, an der er berechtigt ist.

- Innerhalb eines Mandanten können Objekt und ihre Daten zu Geschäftseinheiten zugewiesen werden. Ein Benutzer kann nur auf die Daten von Objekten von berechtigten Geschäftseinheiten zugreifen.

## 7.5.2 Integrität (Art 32 Abs. 1 lit b DSGVO)

---

- **Weitergabekontrolle:** Kein unbefugtes Lesen, Kopieren, Verändern oder Entfernen bei elektronischer Übertragung oder Transport z.B. Verschlüsselung, Verbindungsverschlüsselung (HTTPS) kundenseitig sicherzustellen
  - Für den Zugriff auf das Spider Web User Interface (Web UI) und die Web API wird der Einsatz von https dringend empfohlen
  - Beim Versenden von Datenexporten aus dem Spider System muss kundenseitig die Weitergabekontrolle durch geeignete, technisch organisatorische Maßnahmen sichergestellt werden

## 7.5.3 Verfügbarkeit und Belastbarkeit (Art 32 Abs. 1 lit b DSGVO)

---

- **Verfügbarkeitskontrolle:** Schutz gegen zufällige oder mutmaßliche Zerstörung bzw. Verlust, z.B. Backup-Strategie, unterbrechungsfreie Stromversorgung (USV), Virenschutz, Firewall, Meldewege und Notfallpläne kundenseitig umzusetzen

## 7.6 Bedingungen für die Einwilligung (Artikel 7 DSGVO)

---

Erfolgt mittels Software beim Betroffenen direkt eine Erhebung von personenbezogenen Daten, so werden technische Möglichkeiten zur Erteilung von Einwilligungen, deren Protokollierung und deren Erläuterung erforderlich. Auch ein Widerruf der Einwilligung wird unter den genannten Voraussetzungen benötigt.

### 7.6.1 Self-Service-Funktionen

---

Self-Service-Funktionen, bei denen ein Benutzer (Betroffener im Sinne der DSGVO) direkt seine personenbezogenen Daten erfasst, müssen die Bedingungen zur Einwilligung berücksichtigen. Im Produktstandard findet in Spider keine direkte Erhebung der Daten durch den Betroffenen selbst statt. Der größte Teil der personenbezogenen Daten wird aus Fremdsystemen beigestellt und in Spider verarbeitet. Sollten durch **kundenindividuelle Erweiterungen** Self-Service Funktionen möglich sein, müssen kundenseitig geeignete Maßnahmen zur Erfüllung der Anforderungen getroffen werden.

Bei der Verarbeitung von Beschäftigungsdaten (bspw. Microsoft Active Directory) können Einwilligungen über die begleitenden kundenseitigen Verarbeitungsprozesse oder vorgelagerte Informationsprozesse abgebildet werden.

### 7.6.2 Einwilligung bei personenbezogenen Daten aus Fremdsystemen oder für die Erhebung zusätzlicher Daten

---

Bei der Zulieferung von personenbezogenen Daten aus Drittsystemen (bspw. Microsoft Active Directory) obliegt die Einholung einer Einwilligung der hierfür verantwortlichen Stelle.

Für zusätzliche personenbezogene Daten, die in Spider eingegeben und verarbeitet werden, ist eine Einwilligung der Betroffenen erforderlich. Dies ist in den begleitenden kundenseitigen Verarbeitungsprozessen sicherzustellen.

## 7.7 Informationspflicht (Artikel 13 und 14 DSGVO)

Bei der direkten und indirekten Erhebung personenbezogener Daten existiert eine Informationspflicht gegenüber dem Betroffenen.

Die Erfüllung einer aktiven Informationspflicht kann mit den aktuellen Produktversionen nur über kundenindividuelle Einstellungen und Erweiterungen erfolgen. Häufig werden zur Information der Betroffenen E-Mail-basierte Informationsverfahren eingesetzt. Grundsätzlich stellt sich die Frage, ob überhaupt eine aktive Benachrichtigung der Betroffenen aus Spider sinnvoll ist, da die Produkte i.d.R. für dedizierte Teilaufgaben eingesetzt werden und indirekt die prozessrelevanten personenbezogenen Daten aus Fremdsystemen nutzen.

### 7.7.1 Verarbeitung von Beschäftigungsdaten

Für die häufig anzutreffende Verarbeitung von Beschäftigungsdaten (bspw. aus Microsoft Active Directory) empfehlen wir die Erfüllung der Informationspflicht in den begleitenden kundenseitigen Verarbeitungsprozessen oder vorgelagerten Informationsprozessen sicherzustellen.

### 7.7.2 Direkte Benachrichtigung der Betroffenen

Sofern eine direkte Benachrichtigung der Betroffenen aus dem Spider-System dennoch erforderlich sein sollte, sind die konkreten Informationsprozesse kundenindividuell abzubilden.

## 7.8 Verzeichnis von Verarbeitungstätigkeiten (Artikel 30 DSGVO)

Zur Erfüllung der Dokumentationspflichten gemäß Artikel 30 DSGVO ist u.a. ein Verzeichnis aller Verarbeitungstätigkeiten mit personenbezogenen Daten zu führen. Der Aufsichtsbehörde muss das Verzeichnis der Verarbeitungstätigkeiten auf Anfrage zur Verfügung gestellt werden (Artikel 30 Abs. 4 DSGVO und Erwägungsgrund 82). Unternehmen, die der deutschen behördlichen Aufsicht unterstehen, müssen das Verzeichnissesverzeichnis in deutscher Sprache führen (§23 Abs. 1 und 2 Verwaltungsverfahrensgesetz VwVfG). Eine Ausnahme gemäß Artikel 30 Abs. 5 DSGVO ist auszuschließen, selbst wenn der quantitative Faktor von weniger als 250 Mitarbeiter gegeben ist. Dieses begründet sich in den erforderlichen IT-Protokollierungen, die wesentlich für die Funktionsweise der Spider-Produkte notwendig sind.

Das Verzeichnissesverzeichnis ist kundenindividuell zu erstellen.

## Logfiles

Für die Installation und das Update aller Setups sowie der Services werden Log Dateien geschrieben. Wo diese zu finden sind, entnehmen Sie bitte der nachstehenden Tabelle.

Produkt	Pfad zur Logdatei
Spider Installation / Update	%ProgramData%\brainwaregroup\Log
Recognition Installation / Update	
Software Catalog	

Produkt	Pfad zur Logdatei
Spider Dienste	%ProgramData%\brainwaregroup\CoreServer_xx\<<Name des Dienstes>
Spider Data Receiver Spider Data Collector	%Windir%\
EDC Monitor	%ProgramData%\brainwaregroup\EDCMonitor\Log

## 8.1 Brainware.log

---

Die Brainware.log Logdatei wird periodisch neu geschrieben. Alte Stände werden mit Brainware\_0.log, Brainware\_1.log, Brainware\_2.log, usw. umbenannt.

Die Datei ist eine Textdatei und kann mit jedem Editor geöffnet werden.

## 8.2 SmartInspect-Logdateien (\*.sil)

---

Die \*.sil Dateien liegen im SmartInspect Format vor. Diese sind ohne weiteres nicht lesbar. Es wird dafür ein (kostenloses) Tool der Firma Gurock Software GmbH benötigt.

Die SmartInspect Redistributable Console kann unter, [SmartInspect Redistributable Console](#) heruntergeladen werden.