

Columbus REST

Benutzerhandbuch

Produktversion 7.6.3

columbus

© brainwaregroup - 1997-2019 - Alle Rechte vorbehalten

Jedwede von der brainwaregroup zur Verfügung gestellte Dokumentation unterliegt dem Urheberrecht und ist Eigentum der brainwaregroup. Die brainwaregroup 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. brainwaregroup übernimmt jedoch keine Garantie hinsichtlich dieser Dokumentation und keine gesetzliche Gewährleistung für die marktgängige Qualität und Eignung für einen bestimmten Zweck. Des Weiteren übernimmt brainwaregroup keine Haftung für Fehler oder unbeabsichtigte Schäden bzw. Folgeschäden im Zusammenhang mit der Bereitstellung, Leistung oder Verwendung dieses Dokumentes oder der darin enthaltenen Beispiele. brainwaregroup 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.

Dokumententitel	Columbus REST - Benutzerhandbuch
Produktversion	7.6.3
Herstellung und Druck	Brainware Consulting & Development AG Sumpfstrasse 15 CH-6312 Steinhausen www.brainwaregroup.com
Veröffentlichungsdatum	12.2019

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 der brainwaregroup, weder vollständig noch auszugsweise kopiert, fotokopiert, reproduziert oder weiterverarbeitet werden.

Additional Copyright

Diese Informationen finden in der Dokumentation «Tec_UserManual_Col_7_ComponentLicenses.pdf»

Inhaltsverzeichnis

1	Allgemeines	5
1.1	Lizenzierung	5
1.2	Typografische Konventionen.....	6
1.3	Hilfestellungen	6
1.3.1	Support	6
1.3.2	Dokumentationen	7
1.3.3	Schulungsinformationen	7
2	Übersicht	8
2.1	Was ist REST?	8
2.2	Infrastrukturkomponenten	8
2.3	Konfiguration	8
2.4	Authentifizierung	9
2.5	Ressourcen	9
2.6	Methoden	10
2.7	Auf REST zugreifen	10
3	REST API	12
3.1	Allgemeines.....	12
3.1.1	Formatierung	12
3.1.2	Listen.....	12
3.1.3	Allgemeine Response Codes	13
3.2	devices.....	13
3.2.1	GET /oid	13
3.2.2	POST /	14
3.2.3	PUT /oid	15
3.2.4	DELETE /oid.....	16
3.2.5	GET /oid/actions	16
3.2.6	POST /oid/actions	16
3.2.7	DELETE /oid/actions/aoid	18
3.2.8	GET /oid/isActionPending/actionName	18
3.2.9	GET /oid/inventory[/hw /sw] (7.6.3)	19
3.2.10	POST /oid/operatingSystemTemplates.....	20
3.2.11	POST /oid/softwareAssignments (veraltet)	21
3.2.12	GET /oid/software[/packages /groups] (7.6.3).....	22
3.2.13	PUT /oid/software/packages/packageid (7.6.3)	22
3.2.14	PUT /oid/software/groups/groupid (7.6.3)	23
3.2.15	GET /oid/softwareDepot (7.6.3)	23
3.2.16	DELETE /oid/softwareAssignments/packages/packageid.....	24
3.2.17	DELETE /oid/softwareAssignments/groups/groupid	24
3.2.18	GET /oid/variables	25
3.2.19	POST/PUT /oid/variables	25
3.2.20	DELETE /oid/variables/variable.....	26
3.2.21	GET /oid/os (7.6.3).....	26
3.2.22	GET /oid/osDepot (7.6.3).....	26
3.3	users	27
3.3.1	GET /{oid}.....	27
3.3.2	POST /	28
3.3.3	PUT /oid	29
3.3.4	DELETE /oid.....	29
3.3.5	POST /oid/variables	34
3.3.6	DELETE /oid/variables	35
3.3.7	POST /oid/softwareAssignments/packages (veraltet)	30

3.3.8	GET /oid/software[/packages /groups] (7.6.3).....	31
3.3.9	PUT /oid/software/packages/packageid (7.6.3).....	31
3.3.10	PUT /oid/software/groups/groupid (7.6.3)	32
3.3.11	GET /oid/softwareDepot (7.6.3)	33
3.3.12	DELETE /oid/softwareAssignments/packages/packageid.....	33
3.3.13	DELETE /oid/softwareAssignments/groups/groupid	33
3.3.14	GET /oid/variables	34
3.3.15	PUT /actionid/reschedule	34
3.4	companies	36
3.4.1	GET /{oid}.....	36
3.4.2	GET /oid/[softwareInfo software].....	37
3.4.3	GET /oid/operatingSystemTemplates.....	37
3.5	treeNodes.....	39
3.5.1	GET /{oid}.....	39
3.5.2	POST.....	39
3.6	deviceClasses.....	40
3.6.1	GET /	40

1 Allgemeines

In diesem Kapitel

Lizenzierung.....	5
Typografische Konventionen.....	6
Hilfestellungen.....	6

Es freut uns, dass Sie sich für die brainwaregroup und Columbus entschieden haben.

Mit Columbus und Spider steht Ihnen ein mächtiges integriertes Software- und Lifecycle Management Suite Werkzeug zur Verfügung.

Dieses Handbuch unterstützt Sie dabei, die vielfältigen Möglichkeiten von Columbus gewinnbringend in Ihrem Unternehmen einzusetzen.

1.1 Lizenzierung

Massgeblich für die Lizenzierung ist die Anzahl der Managed-Assets gemäß der gültigen Preisliste zum Zeitpunkt des Lizenzierwerbs.

Als Managed-Asset gelten Computersysteme wie Laptops, Desktops, Server, virtuelle Server, Thin Clients, Mobile Devices, Tablets, jede Assetklasse, für die Software verwaltet werden kann oder für die eine anderweitige aktive Verwaltung technisch vorgesehen ist.

Eine aktive Verwaltung kann z.B. die Installation des Betriebssystems, die Inventarisierung oder die Verteilung von Software sein (keine abschliessende Auflistung). Die vom Kunden gewünschte Leistung bestimmt wiederum die zu lizenzierenden Suiten resp. Module.

Eine Lizenz ist grundsätzlich an einen Computer gebunden, unabhängig davon ob die Lizenz nur einmalig oder kontinuierlich genutzt wird.

Neben Computern ist Columbus in der Lage, Daten von zusätzlichen Netzwerkgeräten (Router, Switches, Printer usw.) aufzunehmen welche als so genannte *Not Managed Devices* verwaltet werden. Diese Geräte erfordern keine Columbus Lizenz, solange sie im Status *Not Managed* sind.

Columbus verfügt über einen aktiven Lizenzcheck für alle Columbus-Module. Überschreitungen werden sowohl an der Console als auch in der Protokolldatei gemeldet.

Zusammenfassung

Die brainwaregroup Lizenzregeln beinhalten folgende Punkte:

- Jedes mit Columbus verwaltete Asset benötigt eine Lizenz.
- Die Übertragung der Lizenz auf einen anderen Computer ist begrenzt auf Ersatzbeschaffungen.

Bei einer Lizenz-Überschreitung

- Erscheint beim Start oder während der Arbeit mit der Console ein Hinweis.
- Werden in den Protokolldateien Einträge generiert.

Wichtig	Sofern sich die Anzahl der Managed-Assets erhöht, verpflichtet sich der Kunde, die Mehranzahl innert 30 Tagen neu zu lizenzieren. Die Software kann den Betrieb nach 30 Tagen Unterlizenzierung einschränken oder einstellen.
----------------	---

Eine Lizenz umfasst zeitlich unbegrenzt das nicht exklusive Nutzungsrecht der Software für die aktuelle Version einschliesslich sämtlicher Hotfixes für ein Jahr ab Rechnungsdatum.

1.2 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
Fette Schrift	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. Datei > Öffnen
<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.

Hinweis Wird für Hinweise oder Tipps verwendet, welche die Arbeit erleichtern oder für zusätzliche Informationen, die das Verständnis für das Produkt fördern.

Wichtig Informationen, die der Benutzer beachten sollte, da sonst Probleme oder Mehraufwand im Betrieb entstehen können.

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

1.3 Hilfestellungen

Für zusätzliche Informationen und Unterstützung empfehlen wir unsere [Website](#) . Hier finden Sie sämtliche Dokumentationen, eine [Knowledgebase](#) und ein Supportformular für Fragen an unsere Spezialisten.

1.3.1 Support

Auf der Supportseite finden Sie den Link zur [KnowledgeBase](#) , das Supportformular und Informationen zu unserem Kundendienst.

- Navigieren Sie in ihrem Browser zu www.brainwaregroup.com und klicken Sie auf der rechten Seite auf **Support und KnowledgeBase**.
- Beginnen Sie ihre Suche in der **KnowledgeBase**, wo die häufigsten Supportfälle aufgeführt sind.

Falls Sie nicht fündig werden, können Sie das Supportformular für weiterführende Informationen benutzen. Beschreiben Sie ausführlich ihr Anliegen, damit ihnen unsere Spezialisten so rasch wie möglich helfen können.

- Klicken Sie auf Supportformular.
- Füllen Sie mindestens die mit * markierten Felder aus und klicken Sie auf **Anfrage absenden**.
 - ℳ Sie erhalten eine Bestätigung, dass ihre Anfrage im System aufgenommen wurde.

Für telefonischen Kontakt finden Sie auf der Supportseite die entsprechenden Telefonnummern.

1.3.2 Dokumentationen

Für die aktuellen Dokumentationen steht auf unserer Website ein eigener Downloadbereich zur Verfügung. Dieser enthält alle Handbücher im PDF und CHM Format in deutscher und englischer Sprache. Einige technische Handbücher sind derzeit nur in Englisch verfügbar.

- Navigieren Sie in ihrem Browser zu www.brainwaregroup.com und wählen Sie auf der rechten Seite **Downloads + Lizenzen**.
- Klicken Sie auf **Login**, melden Sie sich mit ihrem **Benutzernamen** und **Kennwort** an und klicken Sie auf **Anmelden**.
- Wählen Sie beim Downloadtyp **Dokumentation**, das gewünschte Produkt und klicken Sie dann auf **Suchen**.
 - ▮ Eine Übersicht aller zur Verfügung stehenden Dokumentationen wird angezeigt.
- Klicken Sie auf das gewünschte Format und Sprache, um die PDF/CHM-Datei in der jeweiligen Form zu downloaden.

Den Acrobat Reader finden Sie auf der Supportseite oder direkt unter www.adobe.com .

Hinweis Bei der Installation werden die Dokumente standardmässig in das Verzeichnis [C:\Program Files]\Columbus kopiert. Um die Konsolen-Hilfe in Deutsch anzuzeigen, muss im Unterverzeichnis \Console die Datei CMC_de.chm in CMC.chm umbenennet werden.

1.3.3 Schulungsinformationen

Die brainwaregroup hat grosses Interesse daran, dass unsere Produkte und Lösungen fachgerecht, qualifiziert und sachgemäss eingesetzt und betrieben werden. Aus diesem Grund bieten wir eine Reihe verschiedener Kurse an. Sämtliche Informationen über Kursarten und Daten finden Sie auf unserer Website www.brainwaregroup.com unter dem Menüpunkt **Support > Trainings**.

2 Übersicht

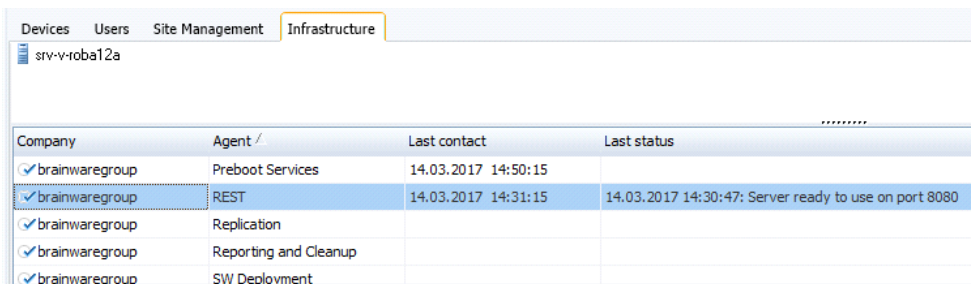
2.1 Was ist REST?

«REST» bedeutet «Representational State Transfer», beschreibt den verbindlichen Übergang von einem Zustand zum nächsten und wird für eine Maschine-zu-Maschine Kommunikation genutzt. Hierbei können über eine vordefinierte Schnittstelle Informationen Abgefragt, Hinzugefügt, Geändert oder Gelöscht werden. Dies geschieht über simple http Abfragen welche heutzutage von jedem Browser aufgerufen werden können.

In Columbus dient diese Schnittstelle dazu Informationen über die Computer oder Benutzer in der Datenbank abzufragen, zu erstellen oder löschen. Dies gilt auch für Softwarezuweisungen, Aktionen oder auch Variablen.

2.2 Infrastrukturkomponenten

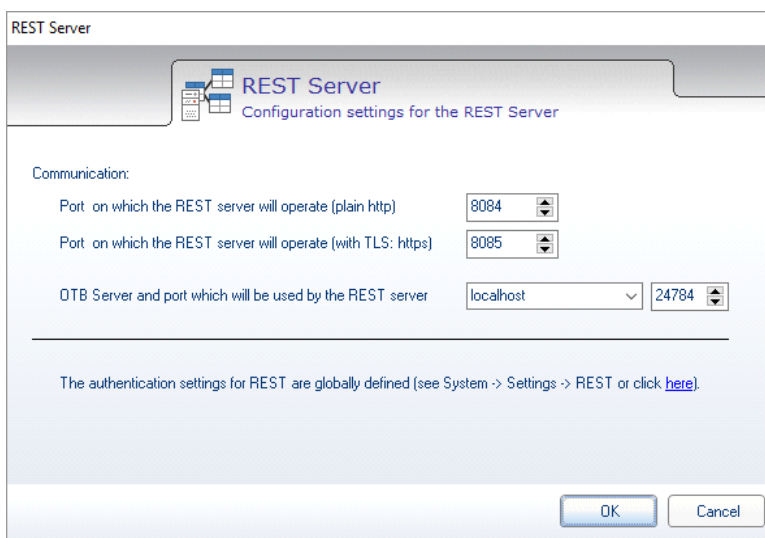
Die REST Schnittstelle wird in Columbus durch den Agenten «REST» abgebildet und benötigt als Informationslieferant lediglich einen OTB-Server. Dies kann in der Konfiguration des Agenten beliebig zugewiesen werden. Es empfiehlt sich hierzu den Master-Server zu nutzen um die bestmögliche Performance zu erreichen.



Company	Agent /	Last contact	Last status
✓ brainwaregroup	Preboot Services	14.03.2017 14:50:15	
✓ brainwaregroup	REST	14.03.2017 14:31:15	14.03.2017 14:30:47: Server ready to use on port 8080
✓ brainwaregroup	Replication		
✓ brainwaregroup	Reporting and Cleanup		
✓ brainwaregroup	SW Deployment		

2.3 Konfiguration

Die Konfiguration der REST Schnittstelle ist über den Reiter Infrastruktur zu erreichen. Wählen Sie hierzu in der «Infrastruktur»-Ansicht den gewünschten Server, welcher die Informationen zur Verfügung stellen soll. Wählen Sie anschließend im Ribbon «Configure» um den nachfolgenden Dialog zu öffnen.



REST Server

REST Server
Configuration settings for the REST Server

Communication:

Port on which the REST server will operate (plain http)

Port on which the REST server will operate (with TLS: https)

OTB Server and port which will be used by the REST server

The authentication settings for REST are globally defined (see System -> Settings -> REST or click [here](#)).

OK Cancel

Option	Beschreibung
--------	--------------

Port	Wählen Sie den Port auf welchem der Service zur Verfügung gestellt werden soll
OTB Server	Wählen Sie den zuständigen OTB Server, welcher die Daten für die Anfragen liefern soll.

2.4 Authentifizierung

Für die Authentifizierung wird eine HTTP Basic Authentication verwendet. Falls Sie ihre Schnittstelle von fremdem Zugriff Schützen möchten, können Sie hierzu einen Benutzer sowie Passwort definieren. Hierbei muss beachtet werden, dass diese Anmeldeinformationen bei jedem Aufruf mitgegeben werden müssen. Weitere Informationen finden Sie unter: <https://de.wikipedia.org/wiki/HTTP-Authentifizierung>

Die Schnittstelle ist zusätzlich wahlweise über HTTPS erreichbar, welches ein abhören erschwert.

Die Authentifizierung der REST Schnittstelle ist global definiert und wird aufgrund dessen in der «System Konfiguration» aufgerufen. Diese erreichen Sie über folgende Wege.

Klicken Sie in der Konfiguration des REST Agenten auf den Link in der unteren Hälfte des Dialoges oder Wählen Sie im «Ribbon» den Reiter «System» und anschliessend die Schaltfläche «Settings».

Im Reiter «Communication» können Sie anschliessend unter «REST Authentication» einen Benutzer sowie ein Passwort für die REST Schnittstelle definieren.

Wichtig Dies ist eine globale Einstellung und gilt für alle REST Agenten.

The screenshot shows a window titled 'System Settings' with a close button (X) in the top right corner. Below the title bar is a ribbon with tabs: 'Object Registration', 'Misc', 'UUID Management', 'Communication' (selected), 'Licensing', and 'Mail'. The 'Communication' tab is active, showing a section for 'Configure System Settings' with the subtitle 'Configure communication settings'. Underneath, there are two main sections: 'OTB Encryption' with checkboxes for 'Use SSL' and 'Force SSL', and 'REST Authentication' with the subtitle 'Credentials for Basic Authentication that will be used to verify contents from external systems'. The 'REST Authentication' section contains two input fields: 'User:' with the value 'RESTUser' and 'Password:' with the value '[leave unchanged]'. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.

2.5 Ressourcen

Eine REST Schnittstelle ist in sogenannte «Ressourcen» unterteilt und gibt je nach Anfrage die Entsprechenden Informationen zurück.

Derzeit sind folgende Ressourcen definiert:

Ressource	Beschreibung
devices	Liefert sämtliche Informationen über Computer sowie deren Inventarinformationen, Softwarezuweisungen, Aktionen, etc
users	Liefert sämtliche Informationen über Benutzer sowie deren Inventarinformationen, Softwarezuweisungen, Aktionen, etc
companies	Liefert die Software und OS welche dieser Company zugewiesen sind
treeNodes	Liefert Informationen Wie Software oder OS Zuweisungen auf der Entsprechenden Organisationseinheit.

Helper-Ressourcen

Folgende Ressourcen liefern Informationen welche nicht direkt mit Objekte oder Organisationalen Einheiten (z.B. TreeNodes) zusammenhängen

Ressource	Beschreibung
deviceClasses	Liefert die vorhandenen Geräteklassifizierungen zurück.

2.6 Methoden

Es gibt grundsätzlich 5 unterschiedliche Methoden um auf einen REST Ressource zuzugreifen wobei. Die wohl bekannteste ist GET. Am meisten wird diese verwendet, um eine Webseite aufgerufen wird (z.B. www.brainwaregroup.com). Bei HTTP/REST gibt's aber noch zusätzliche Methoden um auf dieselbe Ressource bzw. URI direkt Einfluss zu nehmen und diese entsprechend Anzusteuern.

Methode	Beschreibung
GET	Holt Informationen. Z.B. von einem Gerät http://columbus:8084/latest/device/H3EEGkzBM9
POST	Ruft dieselbe URI auf, gibt aber im unsichtbare Body noch Informationen mit. Wird verwendet um neuen Datensätze anzulegen (z.B: Computer).
PUT	Ruft dieselbe URI auf, gibt aber im unsichtbare Body noch Informationen mit. Wird verwendet um bestehenden Datensätze abzuändern oder ergänzen (z.B: Kostenstelle eines Objektes).
PATCH	Wie PUT aber beinhaltet mehrere Änderungen an unterschiedlichen Objekten.
DELETE	Ruft dieselbe URI auf, zeigt aber darauf hin, dass ein Datensatz gelöscht werden soll.

2.7 Auf REST zugreifen

Auf eine REST Schnittstelle kann Grundsätzlich jegliche Applikation zugreifen welche HTTPS sprechen kann. Dies geschieht mit dem Aufruf einer URI und der entsprechenden Ressource. Zusätzlich können zur Einschränkung noch Filterparameter definiert werden um nur das gewünschte Resultat zu erhalten.

Aufbau der URI

```
http://server:port/version/ressource
```

Die Version ist bei Columbus 7.5 / 7.6 den Wert «1» anzugeben. Alternativ kann auch «latest» verwendet werden. Dieser Platzhalter bezieht sich immer auf die letzte Version. Z.B.

```
http://server:port/latest/ressource
```

Beispiele

In folgendem Beispiel wird auf dem Server «columbus», Port 8084, die Ressource «devices» aufgerufen.

```
http://columbus:8084/1/devices
```

Die obige Anfrage liefert anschliessend alle Geräte zurück welche sich in der Datenbank befinden. Die Suche kann natürlich noch - z.B. auf Server – eingeschränkt werden:

```
http://columbus:8084/1/devices?filter=name:srv*
```

oder Domain

```
http://columbus:8084/1/devices?filter=fqdn:acme.inc
```

Beispiel mit CURL und aktiver Authentifizierung

```
curl.exe -u ColumbusRest:Brainware123 -H "Accept:application/json" -X GET "http://columbus:8084/1/devices"
```

Nachfolgen werden die URIs nur noch in Kurzform - ohne Angabe des Servers sowie der Version - dargestellt.

Aus

```
http://columbus:8084/latest/devices
```

wird

```
/devices
```

3 REST API

3.1 Allgemeines

3.1.1 Formatierung

Mit dem Parameter `pretty` kann die Ausgabe formatiert werden.

```
/devices?pretty=true
```

Die Ausgabe ...

```
{"total":0,"maxResults":0,"startAt":0,"items":[]}
```

... wird anschliessend leserlich dargestellt:

```
{
  "total": 0,
  "maxResults": 0,
  "startAt": 0,
  "items": []
}
```

3.1.2 Listen

Mit folgenden Parameter kann die Ausgabe eingegrenzt werden. Dies gilt derzeit für folgende Ressourcen:

- Devices
- Users

Name	Beschreibung	Typ	Standardwert
startAt	Start des ersten Elementes (0 ist erstes Element)	Integer	0
maxResults	Maximale Anzahl der zurückgegebenen Objekte	Integer	1000

Beispiel für die Elemente 100 bis 150

```
/devices?startAt=100&maxResults=50
```

3.1.3 Allgemeine Response Codes

Folgende Antwortcodes sind für alle all gültig.

Code	Beschreibung
200	Anfrage erfolgreich / Rückgabe des Objektes
404	Objekt nicht gefunden
50x	Server Error
501	Ressource nicht implementiert

3.2 devices

3.2.1 GET /oid

Gibt Informationen über die Geräte in der Datenbank aus. Diese kann mit **oid** auf ein einzelnes Gerät beschränkt werden.

URI Parameter

Name	Beschreibung
{oid}	(Optional) OID eines Gerätes. Z.b. <code>ExAmpL3d3v</code>
filter	Filter um die Resultate einzuschränken. Mehrere Filter mittels ";" (Komma) trennen. Benutze * als Platzhalter. Mehrere Informationen mit Komma trennen – Siehe Beispiele. <code>name</code> – Hostname <code>workgroup</code> – Arbeitsgruppe <code>fqdn</code> – Fully quallyfied domain name <code>uuid</code> – UUID des Gerätes <code>mac</code> – MAC Adresse <code>companyId</code> – OID der Company <code>homeId</code> – OID der Organisationseinheit <code>costCenter</code> – Konstenstelle
include	<code>softwareStatus</code> – Softwarezuweisungen <code>hardwareInventory</code> – Hardwareinventar <code>softwareInventory</code> – Softwareinventar (MS uninstall) <code>actions</code> – Aktionen <code>variables</code> – Variablen <code>osTemplate</code> – Zugewiwsenes OS template

Beispiele

Gibt alle Geräte zurück

```
GET /devices
```

Gibt ein bestimmtes Gerät anhand der OID zurück

```
GET /devices/ExAmpl3d3v
```

Gibt ein bestimmtes Gerät anhand der OID zurück und dessen Softwarezuweisungen sowie Hardwareinventar

```
GET /devices/ExAmpl3d3v?include=softwareStatus,hardwareInventory
```

Suchen nach Geräten der Domäne «bwg.corp» und mit dem präfix «srv-»

```
GET /devices?filter=fqdn:bwg.corp,name:srv-*
```

Response Codes

Code	Beschreibung
200	Rückgabe des Objektes
404	Objekt nicht gefunden

3.2.2 POST /

Erstellt ein Gerät. Bei der Erstellung wird geprüft, ob das gewünschte Objekt bereits besteht (Merkmale: name, fqdn, MAC, UUID).

Request Parameter

Parameter	Optional	Beschreibung
name	Nein	Hostname
workgroup	Nein *	Workgroup
fqdn	Nein *	Voll qualifizierter Name der Domäne
uuid	Ja	UUID vom BIOS
mac	Ja	Ethernet MAC Adresse
homeId	Ja **	OID der Organisationseinheit oder Firma (nur benötigt wenn mehrere Firmen vorhanden sind)
deviceClassId	Ja	Gerätekategorie – die Helperfunktion «deviceClass» gibt alle verfügbaren Klassen zurück
activationState	Ja	MANAGED LOCKED UNMANAGED UNKNOWN
costcenter	Ja	String – Max 30 Zeichen

* Entweder `fqdn` oder `workgroup` angeben

** Falls mehrere Firmen vorhanden sind, muss die `homeId` definiert werden.

Beispiel

```
POST /devices
```

```
{
  "name": "wrk-p-rest000",
  "workgroup": "restdom",
  "fqdn": "rest.dom",
  "uuid": "12345678-1234-1234-1234-1234567890AB",
  "mac": "1234567890AB",
  "homeId": "5qQExAMP13",
  "deviceClassId": "~TypPiVDI",
  "activationState": "LOCKED",
  "costCenter": "REST-DEMO"
}
```

Response Codes

Code	Beschreibung
200	Das Gerät existiert bereits in der Datenbank
201	Das Gerät wurde erfolgreich erstellt
400	Fehler bei der Erstellung des Objektes

3.2.3 PUT /oid

Aktualisiert die Eigenschaften eines Gerätes.

URI Parameter

Name	Beschreibung
oid	OID eines Gerätes

Body Parameter

Parameter	Optional	Beschreibung
name	Ja	Hostname
workgroup	Ja	Workgroup
fqdn	Ja	Fully qualified domain name
uuid	Ja	UUID vom BIOS
mac	Ja	Ethernet MAC Adresse
homeld	Ja	OID der Organisationseinheit
deviceClassId	Ja	Geräteklasse – die Helperfunktion «deviceClass» gibt alle verfügbaren Klassen zurück
activationState	Ja	ACTIVE LOCKED UNMANAGED
costcenter	Ja	String – Max 30 Zeichen

Beispiel

PUT /devices/ExAmpL3d3v

```
{
  "activationState": "UNMANAGED",
  "costCenter": "Lager"
}
```

Response Codes

Code	Beschreibung
200	Anfrage wurde erfolgreich abgesetzt
400	Ungültiger Body
404	Objekt nicht gefunden

3.2.4 DELETE /oid

Löscht das angegebene Gerät

Parameter

Name	Beschreibung
{oid}	OID eines Gerätes

Beispiel

```
DELETE /devices/ExAmpL3123
```

Response Codes

Code	Beschreibung
202	Das Gerät wird gelöscht
400	Keine Device ID angegeben

3.2.5 GET /oid/actions (7.6.3)

Listet alle pendenten Aktionen eines Gerätes auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

```
GET /devices/ExAmpL312D3V/actions
```

Response Codes

Code	Beschreibung
200	Aktion erstellt
400	Ungültiger Request
404	Objekt nicht gefunden

3.2.6 POST /oid/actions

Setzt eine Aktion auf ein Gerät. Die actions müssen in Grossbuchstaben angegeben werden.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Request Parameter

Name	Beschreibung
------	--------------

code	SWPROC – Software push POWERON – Wake On Lan REBOOT - Neustart SHUTDOWN – Herunterfahren CMBACKUP – Backup der Userdaten AUTOSMARTDEPLOY – Smart OSDeploy EPSCAN – UpdateManagement Patch scan EPSCANINSTALL – UpdateManagement Patch scan und Installation OCOMPBACK – Online Backup CLIUNINST – Deinstallation des Management Clients
------	--

Beispiel

POST /devices/ExAmpL312D3V/actions

```
{  
  "code": "SWPROC"  
}
```

Response Codes

Code	Beschreibung
201	Aktion erstellt
400	Unbekannte Aktion
404	Angegebenes Objekt (OID) nicht gefunden

3.2.7 DELETE /oid/actions/aid

Löscht eine Aktion auf dem

URI Parameter

Name	Beschreibung
Oid	OID des Gerätes
Aoid	OID der Aktion Eine Liste der vorhandenen Aktionen kann mit /oid/actions abgerufen werden.

Eine liste der vorhandenen Aktionen kann mit /oid/actions abgerufen werden

Beispiel

```
DELETE /devices/ExAmpL312D3V/actions/ExAmpL312Ac7
```

Response Codes

Code	Beschreibung
202	Aktion gelöscht
400	Unbekannte Aktion
404	Gerät nicht gefunden

3.2.8 GET /oid/isActionPending/actionName

Prüft ob die Aktion `actionName` auf einem Gerät pendent ist.

URI Parameter

Name	Beschreibung
oid	OID eines Gerätes. Z.b. ExAmpL3D3v

actionName	SWPROC – Software push POWERON – Wake On Lan REBOOT - Neustart SHUTDOWN – Herunterfahren CMBACKUP – Backup der Userdaten AUTOSMARTDEPLOY – Smart OSDeploy EPSCAN – UpdateManagement Patch scan EPSCANINSTALL – UpdateManagement Patch scan und Installation OCOMPBACK – Online Backup CLIUNINST – Deinstallation des Management Clients
------------	--

Beispiel

```
GET /devices/ExAmpL3D3v/isActionPending/REBOOT
```

Response Codes

Code	Beschreibung
201	Anfrage OK
400	Unbekannte Aktion
404	Angegebenes Objekt (OID) nicht gefunden
422	Gerät nicht aktiviert
501	Oid ist kein Gerät

3.2.9 GET /oid/inventory[/hw|/sw] (7.6.3)

Listet das Hardware- bzw. Softwareinventar auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

```
GET /devices/ExAmpL3123/inventory
GET /devices/ExAmpL3123/inventory/hw
GET /devices/ExAmpL3123/inventory/sw
```

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.2.10 POST /oid/operatingSystemTemplates

Weist ein OS template zu.

Request Parameter

Name	Beschreibung
oid	Geräte OID

Die mögliche OS können über die Helperfunktion «operatingSystems» abgefragt werden werden.

Beispiel

POST /devices/ExAmpL3d3v/operatingSystemTemplates

```
{
  "Id": 7
}
```

Response Codes

Code	Beschreibung
201	Release template zugewiesen
400	Template existiert nicht
403	OS deploy nicht lizenziert
404	Angegebenes Objekt existiert nicht.
422	Gerät nicht aktiviert

3.2.11 POST /oid/softwareAssignments (veraltet)

WICHTIG: Diese Funktion wurde überarbeitet. Bitte nutzen Sie «PUT /oid/software/packages/{packageId}» bzw. «PUT/oid/software/groups/{packageId}».

Weist ein Paket oder Paketgruppe zu.

URI Parameter

Name	Beschreibung
oid	OID des Objektes

Request Parameter

Name	Beschreibung
id	Name des Paketes oder Identifier des Paketes. Format: <Identifier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01
softwareType	PACKAGE GROUP

Beispiel

POST /devices/ExAmpL3123/softwareAssignments

```
{
  "id": "900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01",
  "softwareType": "PACKAGE"
}
```

Response Codes

Code	Beschreibung
201	Zuweisung erstellt
400	Fehlerhafter request body
403	Softwaredeploy nicht lizenziert

404	Gerät nicht gefunden
422	Gerät nicht aktiviert

3.2.12 GET /oid/software[/packages[/groups]] (7.6.3)

Listet die zugewiesenen Softwarepakete bzw. gruppen auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

```
GET /devices/ExAmpL3123/software
GET /devices/ExAmpL3123/software/packages
GET /devices/ExAmpL3123/software/groups
```

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.2.13 PUT /oid/software/packages/packageid(7.6.3)

Weist ein Paket zu.

URI Parameter

Name	Beschreibung
oid	OID des Objektes
packageid	Identifizier des Paketes. Format: <Identifizier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01

Beispiel

```
PUT /devices/ExAmpL3123/software/packages/903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01
```

Response Codes

Code	Beschreibung
201	Zuweisung erstellt
400	Fehlerhafter request
403	Softwaredeploy nicht lizenziert
404	Gerät nicht gefunden

422	Gerät nicht aktiviert
-----	-----------------------

3.2.14 PUT /oid/software/groups/groupid (7.6.3)

Weist eine Softwaregruppe zu.

URI Parameter

Name	Beschreibung
oid	OID des Objektes
groupid	Name der Gruppe z.b. «Client-Standard-2019» Wenn die Gruppe ein Leerzeichen enthält, muss dieses codiert werden "Client Standard 2019" -> "Client%20Standard%202019"

Beispiel

PUT /devices/ExAmpL3123/software/groups/Client-Standard-2019

Response Codes

Code	Beschreibung
201	Zuweisung erstellt
400	Fehlerhafter request
403	Softwaredeploy nicht lizenziert
404	Gerät nicht gefunden
422	Gerät nicht aktiviert

3.2.15 GET /oid/softwareDepot (7.6.3)

Listet die verfügbaren Softwarepakete und Gruppen auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

GET /devices/ExAmpL3123/softwareDepot

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.2.16 DELETE /oid/softwareAssignments/packages/packageId

Entfernt eine zugewiesene Software

URI Parameter

Name	Beschreibung
oid	OID des Gerätes
packageId	Identifier des Paketes. Format beachten: <Identifier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01

Beispiel

```
DELETE
/devices/ExAmpL3d3v/softwareAssignments/packages/900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01
```

Response Codes

Code	Beschreibung
201	Anfrage wird ausgeführt
400	Ungültiger request body
403	SoftwareDeploy nichth lizenziert
404	Objekt nicht gefunden
422	Gerät nicht aktiviert

3.2.17 DELETE /oid/softwareAssignments/groups/groupId

Entfernt eine zugewiesene Software

URI Parameter

Name	Beschreibung
oid	OID des Gerätes
groupId	Name der Gruppe z.b. «Client-Standard-2019» Wenn die Gruppe ein Leerzeichen enthält, muss dieses codiert werden "Client Standard 2019" -> "Client%20Standard%202019"

Beispiel

```
DELETE /devices/ExAmpL3123/softwareAssignments/groups/Client-Standard-2019
```

Code	Beschreibung
201	Anfrage wird ausgeführt
400	Ungültiger request body
403	SoftwareDeploy nichth lizenziert
404	Objekt nicht gefunden

422	Gerät nicht aktiviert
-----	-----------------------

3.2.18 GET /oid/variables (7.6.3)

Listet alle zugewiesenen Variablen auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

```
GET /devices/ExAmpL312D3V/variables
```

Response Codes

Code	Beschreibung
201	OK
400	Unbekannte Aktion
404	Angegebenes Objekt (OID) nicht gefunden

3.2.19 POST/PUT /oid/variables

Erstellt oder ändert eine direkt zugewiesenen Variable.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Request Parameter

Name	Beschreibung
name	Name der zu erstellenden Variable
value	Wert der zu erstellenden Variable

Beispiel

```
POST /devices/ExAmpL3123/variables
```

```
PUT /devices/ExAmpL3123/variables
```

```
{
  "name": "c_MachineLanguage",
  "value": "de-DE"
}
```

Response Codes

Code	Beschreibung
201	Variable erstellt
400	Keine Informationen im Request body

404	Angegebenes Objekt nicht gefunden
-----	-----------------------------------

3.2.20 DELETE /oid/variables/variable

Löscht eine zugewiesene Variable

URI Parameter

Name	Beschreibung
oid	OID des Gerätes
variable	Name der Variable

Beispiel

```
DELETE /devices/ExAmpL3123/variables/c_restvar
```

Response Codes

Code	Beschreibung
202	Variable wird gelöscht
400	Keine Informationen im request body
404	Angegebenes Device nicht gefunden

3.2.21 GET /oid/os (7.6.3)

Listet das zugewiesene OS template auf.

URI Parameter

Name	Beschreibung
oid	OID des Gerätes

Beispiel

```
GET /devices/ExAmpL3123/os
```

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.2.22 GET /oid/osDepot (7.6.3)

Listet die verfügbaren OS templates auf.

URI Parameter

Name	Beschreibung
------	--------------

oid	OID des Gerätes
-----	-----------------

Beispiel

```
GET /devices/ExAmpL3123/osDepot
```

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.3 users

3.3.1 GET /{oid}

Gibt Informationen über die Benutzer in der Datenbank aus. Diese kann mit `oid` auf einen einzelnen Benutzer beschränkt werden.

URI Parameter

Name	Beschreibung
{oid}	(Optional) OID eines Benutzers. Z.b. <code>ExAmpL3uSR</code>
filter	Filter um die Resultate einzuschränken. Mehrere Filter mittels "," (Komma) trennen. Benutze * als Platzhalter. <code>name</code> – Hostname <code>workgroup</code> – Arbeitsgruppe <code>fqdn</code> – Fully qualified domain name <code>companyId</code> – OID der Company <code>homeId</code> – OID der Organisationseinheit <code>costCenter</code> – Kostenstelle <code>eMail</code> – E-Mail Adresse <code>phone</code> – Telefonnummer (Wenn mit AD synchronisiert oder manuell geführt) <code>fullName</code> – Anzeigenname (Wenn mit AD synchronisiert oder manuell geführt)
include	<code>softwareStatus</code> – Zugewiesene Pakete <code>variables</code> - Variablen

Beispiele

Gibt alle Benutzer zurück

```
/users
```

Nur Benutzer der Domäne «bwg.training» und mit dem präfix «training»

```
/users?filter=fqdn:bwg.training,name:training*
```

3.3.2 POST /

Erstellt einen Benutzer. Bei der Erstellung wird geprüft, ob das gewünschte Objekt bereits besteht (Merkmale name, fqdn, workgroup).

Request Parameter

Parameter	Optional	Beschreibung
name	Nein	Hostname
workgroup	Nein*	Workgroup
fqdn	Nein*	Voll qualifizierter Name der Domäne
homeld	Ja**	OID der Organisationseinheit oder Firma
activationState	Ja	ACTIVE LOCKED UNMANAGED
costcenter	Ja	String – Max 30 Zeichen
eMail	Ja	E-Mail Adresse
phone	Ja	Telefonnummer

* Entweder "fqdn" oder "workgroup" angeben

** Falls mehrere Firmen vorhanden sind, muss die homeld definiert werden.

Beispiel

POST /users

```
{
  "name": "rest-User",
  "workgroup": "restdom",
  "fqdn": "rest.dom",
  "homeId": "5qQExAMP13",
  "activationState": "ACTIVE",
  "costCenter": "REST-DEMO"
}
```

Response Codes

Code	Beschreibung
200	Der Benutzer existiert bereits in der Datenbank
201	Der Benutzer wurde erfolgreich erstellt
400	Fehler bei der Erstellung des Objektes

3.3.3 PUT /oid

Aktualisiert die Eigenschaften eines Benutzers.

URI Parameter

Name	Beschreibung
oid	OID eines Benutzers

Body Parameter

Parameter	Optional	Beschreibung
name	Ja	Hostname
workgroup	Ja *	Workgroup
fqdn	Ja *	Fully qualified domain name
homeld	Ja	OID der Organisationseinheit
activationState	Ja	ACTIVE LOCKED UNMANAGED
costcenter	Ja	String – Max 30 Zeichen
eMail	Ja	E-Mail Adresse
phone	Ja	Telefonnummer

Beispiel

PUT /users/ExAmpL3uSR

```
{
  "activationState": "UNMANAGED",
  "costCenter": "Inaktiv - Austritt"
}
```

Response Codes

Code	Beschreibung
200	Update erfolgreich
400	Ungültiger Body
404	Objekt nicht gefunden

3.3.4 DELETE /oid

Löscht den angegebenen Benutzer

URI-Parameter

Name	Beschreibung
oid	OID des Benutzers

Beispiel

DELETE /users/ExAmpL3uSR

Response Codes

Code	Beschreibung
202	Benutzer wird gelöscht
404	Benutzer nicht gefunden

3.3.5 POST /oid/softwareAssignments/packages (veraltet)

WICHTIG: Diese Funktion wurde überarbeitet. Bitte nutzen Sie «PUT /oid/software/packages/{packageId}» bzw. «PUT /oid/software/groups/{groupid}».

Weist ein Paket oder Paketgruppe zu. Es können nur «Benutzer Pakete» (Bedingte Verwendung) an Benutzer zugewiesen werden.

URI Parameter

Name	Beschreibung
oid	OID des Users

Request Parameter

Name	Beschreibung
id	Name der Gruppe oder Identifier des Paketes. Format beachten: <Identifier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01
softwareType	PACKAGE GROUP

Beispiel

POST /users/ExAmpL3uSR/softwareAssignments

```
{
  "id": "900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01",
  "softwareType": "PACKAGE"
}
```

Response Codes

Code	Beschreibung
201	Zuweisung wird erstellt
400	Ungültiger request body
404	Objekt nicht gefunden

3.3.6 GET /oid/software[/packages]/groups (7.6.3)

Listet die zugewiesenen Softwarepakete bzw. gruppen auf.

URI Parameter

Name	Beschreibung
oid	OID des Users

Beispiel

```
GET /users/ExAmpL3uSR/software
GET /users/ExAmpL3uSR/software/packages
GET /users/ExAmpL3uSR/software/groups
```

Response Codes

Code	Beschreibung
201	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.3.7 PUT /oid/software/packages/packageid (7.6.3)

Weist ein Paket oder Paketgruppe zu. Es können nur «Benutzer Pakete» (Bedingte Verwendung) an Benutzer zugewiesen werden.

URI Parameter

Name	Beschreibung
oid	OID des Users
packageid	Identifizier des Paketes. Format beachten: <Identifizier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01

Beispiel

```
PUT
/users/ExAmpL3uSR/software/packages/900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01
```

Response Codes

Code	Beschreibung
201	Zuweisung wird erstellt
400	Ungültiger request
404	Objekt nicht gefunden

3.3.8 PUT /oid/software/groups/groupid (7.6.3)

Weist eine Paketgruppe zu.

URI Parameter

Name	Beschreibung
oid	OID des Users
groupid	

Request Parameter

Name	Beschreibung
id	Name der Gruppe oder Identifier des Paketes. Format beachten: <Identifier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01

Beispiel

```
PUT /users/ExAmpL3uSR/software/903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01
```

Response Codes

Code	Beschreibung
201	Zuweisung wird erstellt
400	Ungültiger request body
404	Objekt nicht gefunden

3.3.9 GET /oid/softwareDepot (7.6.3)

Listet die verfügbaren Softwarepakete und Gruppen auf.

URI Parameter

Name	Beschreibung
oid	OID des Benutzers

Beispiel

```
GET /users/ExAmpL3123/softwareDepot
```

Response Codes

Code	Beschreibung
200	OK
400	Ungültiger request
404	Objekt nicht gefunden

3.3.10 DELETE /oid/softwareAssignments/packages/packageId

Entfernt eine direkt zugewiesene Software

URI Parameter

Name	Beschreibung
oid	OID des Benutzers
packageId	Identifizier des Paketes. Format: <Identifizier>_<Sprache>_<Version> z.B: 900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01 903000_COLUMBUS_PACKAGINGTOOLS_7_UNI_01

Beispiel

```
DELETE /users/ExAmpL3uSR/softwareAssignments/packages/900000_REMOTELYANYWHERE_12_CONFIG_CLIENT_ENU_01
```

Response Codes

Code	Beschreibung
202	Anfrage wird ausgeführt
400	Ungültiger request body
404	Objekt nicht gefunden

3.3.11 DELETE /oid/softwareAssignments/groups/groupId

Entfernt eine direkt zugewiesene Softwaregruppe

URI Parameter

Name	Beschreibung
oid	Name der zu erstellenden Variable
groupId	Name der Softwaregruppe

Beispiel

```
DELETE /users/ExAmpL3uSR/softwareAssignments/groups/CH_Standard_Client_2016
```

Response Codes

Code	Beschreibung
202	Anfrage wird ausgeführt
400	Ungültiger request body
404	Objekt nicht gefunden

3.3.12 GET /oid/variables (7.6.3)

Listet alle zugewiesenen Variablen auf.

URI Parameter

Name	Beschreibung
oid	OID des Benutzers

Beispiel

```
GET /devices/ExAmpL312D3V/variables
```

Response Codes

Code	Beschreibung
201	OK
400	Unbekannte Aktion
404	Angegebenes Objekt (OID) nicht gefunden

3.3.13 POST /oid/variables

Erstellt oder ändert eine direkt zugewiesenen Variable.

URI Parameter

Name	Beschreibung
oid	OID des Benutzers

Request Parameter

Name	Beschreibung
name	Name der zu erstellenden Variable
value	Wert der zu erstellenden Variable

Beispiel

POST /devices/ExAmpL3uSR/variables

```
{
  "name": "c_UserLanguage",
  "value": "de-DE"
}
```

Response Codes

Code	Beschreibung
200	Variable erstellt
400	Keine oder ungültige Informationen im Request Body
404	Angegebenes Objekt nicht gefunden

3.3.14 DELETE /oid/variables

Löscht eine zugewiesene Variable

URI Parameter

Name	Beschreibung
oid	OID des Benutzers

Request Parameter

Name	Beschreibung
name	Name der zu löschenden Variable

Beispiel

```
DELETE /ExAmpL3uSR/variables
```

Response Codes

Code	Beschreibung
202	Variable gelöscht
400	Keine/ungültige Informationen im Request body
404	Angegebenes Device nicht gefunden

3.4 Actions

3.4.1 PUT /actionid/reschedule

Setzt den Zeitpunkt der Ausführung auf das nächste Intervall. Dieser Aufruf kann nur für repetitive Aktionen verwendet werden.

URI Parameter

Name	Beschreibung
actionid	OID einer Aktion. Z.b. ExAmpL3Ac7

Beispiele

Gibt alle Aktionen eines Gerätes zurück

```
GET /devices/ExAmpL3D3v/actions
```

Setzt den Timer auf das nächste Intervall

```
PUT /actions/ExAmpL3Ac7/reschedule
```

Response Codes

Code	Beschreibung
200	Anfrage erfolgreich ausgeführt
202	Ungültiger request

3.5 companies

3.5.1 GET /{oid}

Gibt Informationen über die Companies in der Datenbank aus. Diese kann mit `oid` auf einen einzelnen Benutzer beschränkt werden.

URI Parameter

Name	Beschreibung
oid	(Optional) OID einer Company. Z.b. ExAmpL30rG

Beispiele

Gibt alle Companies zurück

```
GET /companies
```

Gibt die Informationen einer bestimmten Company zurück

```
GET /companies/ExAmpL30rG
```

Response Codes

Code	Beschreibung
200	Anfrage erfolgreich ausgeführt
400	Keine Company {oid} angegeben
404	Objekt nicht gefunden

3.5.2 GET /oid/[softwareInfo|software]

Listet alle Softwarepakete und Softwaregruppen eine Company aus.

URI Parameter

Name	Beschreibung
oid	OID der Organisationseinheit. Z.b. <code>ExAmpL30rG</code>

Beispiel

```
GET /companies/ExAmpL30rG/softwareInfo
```

```
GET /companies/ExAmpL30rG/software
```

Response Codes

Code	Beschreibung
200	Anfrage erfolgreich ausgeführt
400	Keine Company oid angegeben
404	Objekt nicht gefunden

3.5.3 GET /oid/operatingSystemTemplates

Listet alle Softwarepakete und Softwaregruppen eine Company aus.

URI Parameter

Name	Beschreibung
oid	OID der Organisationseinheit. Z.b. <code>ExAmpL30rG</code>

Beispiel

```
GET /companies/ExAmpL30rG/operatingSystemTemplates
```

Response Codes

Code	Beschreibung
200	Anfrage erfolgreich ausgeführt
400	Keine Company oid angegeben

404	Objekt nicht gefunden
-----	-----------------------

3.6 treeNodes

3.6.1 GET /{oid}

Gibt Informationen über die Organisationseinheiten aus.

URI Parameter

Feld	Beschreibung
{oid}	(Optional) OID der Organisationseinheit. Z.b. ExAmpL30rG

Beispiel

Gibt den kompletten Organisationsbaum zurück

```
GET /treeNodes
```

Gibt eine bestimmte Organisationseinheit zurück

```
GET /treeNodes/ExAmpL30rG
```

Response Codes

Code	Beschreibung
200	Request OK
400	Keine OID angegeben
404	Organisationseinheit nicht gefunden

3.6.2 POST

Erstellt eine Organisationseinheit.

Beispiel

```
POST /treeNodes
```

```
{
  "fullpath": "Organisations.brainwaregroup.Neue OU"
}
```

Response Codes

Code	Beschreibung
201	OU erstellt
400	Company nicht gefunden

3.7 deviceClasses

3.7.1 GET /

Gibt alle Geräteklassen zurück

Beispiel

GET /deviceClasses

Response Codes

Code	Beschreibung
200	OK