Altova StyleVision Server 2025

Benutzer- und Referenzhandbuch

Altova StyleVision Server 2025
Benutzer- und Referenzhandbuch

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Published: 2025

© 2019-2025 Altova GmbH

Inhaltsverzeichnis

1 Einführung 7

2 Funktionalität 8

2.1 Im FlowForce Workflow 9

2.2 Als eigenständiger Server 10

3 Installation und Lizenzierung 11

3.1 Einrichten unter Windows 12

3.1.1 Installation unter Windows 12

3.1.2 Installation auf Windows Server Core 13

3.1.3 Installation von LicenseServer (Windows) 15

3.1.4 Starten von LicenseServer, StyleVision Server (Windows) 16

3.1.5 Registrieren von StyleVision Server (Windows) 17

3.1.6 Zuweisen einer Lizenz (Windows) 18

3.1.7 Zusätzliche Anmerkungen zur Konfiguration (Windows) 19

3.2 Einrichten unter Linux 20

3.2.1 Installation unter Linux 21

3.2.2 Installation von LicenseServer (Linux) 22

3.2.3 Starten von LicenseServer, StyleVision Server (LInux) 23

3.2.4 Registrieren von StyleVision Server (Linux) 23

3.2.5 Zuweisen einer Lizenz (LInux) 24

3.2.6 Anmerkungen zur Umgebung (Linux) 25

3.3 Einrichten auf macOS 27

3.3.1 Installation auf macOS 27

3.3.2 Installation von LicenseServer (macOS) 29

3.3.3 Starten von LicenseServer, StyleVision Server (macOS) 29

3.3.4 Registrieren von StyleVision Server (macOS) 30

3.3.5 Zuweisen einer Lizenz (macOS) 30

3.3.6 Anmerkungen zur Umgebung (macOS) 31
3.4 Upgraden von StyleVision Server 33
3.5 Migrieren von StyleVision Server auf einen neuen Rechner 34
3.6 FOP-Anforderungen 35
3.7 Sicherheitsaspekte 36

4 StyleVision Server-Befehlszeile 37
4.1 accepteula (nur Linux) 39
4.2 assignlicense 40
4.3 exportresourcestrings 42
4.4 generate 44
4.5 help 49
4.6 licenseserver 50
4.7 pdfdata 52
4.8 setdeflang 54
4.9 setfopath 55
4.10 verifylicense 57
4.11 version 59

5 StyleVision Server API 60
5.1 Informationen zur .NET-Schnittstelle 61
5.2 Informationen zur COM-Schnittstelle 62
5.3 Informationen zur Java-Schnittstelle 63
5.4 Codebeispiele 65
5.4.1 C# 65
5.4.2 C++ 66
5.4.3 Java 68
5.4.4 VBScript 69
5.4.5 Visual Basic 71
5.5 API-Referenz 74
5.5.1 COM und .NET 74
5.5.2 Java 81

6 Schema-Manager 86

6.1 Ausführen des Schema-Managers ... 89
6.2 Statuskategorien ... 92
6.3 Anwenden eines Patch oder Installation eines Schemas ... 94
6.4 Deinstallieren eines Schemas, Zurücksetzen ... 96
6.5 Befehlszeilenschnittstelle (CLI) ... 97
6.5.1 help ... 97
6.5.2 info ... 98
6.5.3 initialize ... 98
6.5.4 install ... 99
6.5.5 list ... 100
6.5.6 reset ... 100
6.5.7 uninstall ... 101
6.5.8 update ... 102
6.5.9 upgrade ... 103

7 Taxonomie-Manager 104

7.1 Ausführen des Taxonomie-Managers ... 108
7.2 Statuskategorien ... 111
7.3 Anwenden eines Patch oder Installation einer Taxonomie ... 113
7.4 Deinstallieren einer Taxonomie, Zurücksetzen ... 115
7.5 Befehlszeilenschnittstelle (CLI) ... 116
7.5.1 help ... 116
7.5.2 info ... 117
7.5.3 initialize ... 117
7.5.4 install ... 118
7.5.5 list ... 119
7.5.6 reset ... 119
7.5.7 uninstall ... 120
7.5.8 update ... 121
7.5.9 upgrade ... 122

Index **123**

1 Einführung

Altova StyleVision Server ist eine Implementierung des Built-in-Ausführungsprozessors von Altova StyleVision. Das Produkt kann als eigenständiges Server-Produkt[10] oder als Modul von FlowForce Server[9] von Altova verwendet werden.

StyleVision Server führt Transformationspakete aus, die auf einem Speichermedium gespeichert oder auf einem FlowForce Server[9] bereitgestellt wurden.

- StyleVision Server-Funktionalitäten können über die Befehlszeile[37] aufgerufen werden, um ein Transformationspaket auszuführen und die im Paket definierten Dateien zu generieren.
- StyleVision Server-Transformationen werden von FlowForce Server auf Basis von in FlowForce Server definierten Triggern, Datei-Triggern oder Remote Triggern initiiert.

StyleVision Server steht sowohl für 32-Bit als auch 64-Bit-Windows-Rechner zur Verfügung. Nähere Informationen zur Installation und Lizenzierung finden Sie in den Abschnitten "Einrichten von..." für Windows[12], Linux[20] und macOS[27].

Anmerkung: Wenn die ausfüllbaren Bereiche eines ausfüllbaren PDF-Dokuments fehlen, wenn das PDF-Dokument auf einem macOS-System geöffnet wird, ist die Ursache wahrscheinlich, dass Java 6 auf dem Rechner nicht installiert ist. Installieren Sie in diesem Fall Java 6 von https://support.apple.com/kb/dl1572?locale=en_US. Wenn auf Ihrem System bereits eine neuere Version als Java 6 installiert ist, hat die Installation der älteren Java 6-Version keine Auswirkung auf das Funktionieren der neuen Version, die weiterhin als die Standardversion des Systems verwendet wird.

Letzte Aktualisierung: 20.03.2025

2 Funktionalität

StyleVision Server transformiert XML-Dateien mit Hilfe von XSLT-Stylesheets in HTML-, PDF-, RTF- und DOCX-Ausgabedokumente. Diese XSLT-Stylesheets sind in PXF-Dateien enthalten, die mit Altova StyleVision, der Altova-Applikation zur Erstellung von Stylesheets, erstellt wurden.

StyleVision Server kann auf zwei Arten verwendet werden:

- als Teil des Altova Altova FlowForce Workflow. Nähere Informationen zu Altova FlowForce finden Sie auf der Altova Website.
- als eigenständiges Server-Produkt 10, das über die Befehlszeile (CLI) aufgerufen wird

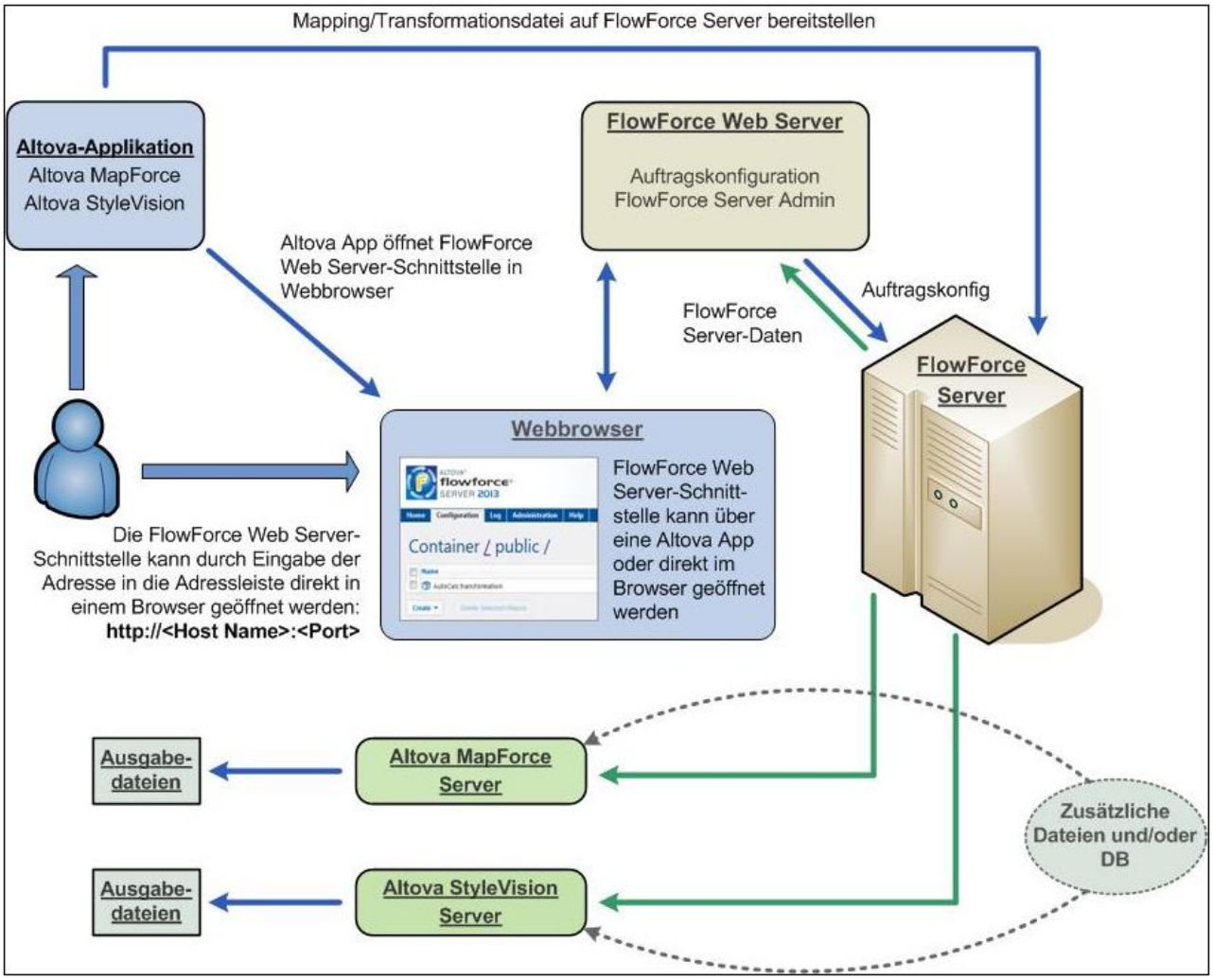
Eine XML-Input-Datei und eine PXF-Datei werden an StyleVision Server übergeben, der das/die erforderliche(n) Ausgabedokument(e) erzeugt.

© 2019-2025 Altova GmbH

2.1 Im FlowForce Workflow

Ein FlowForce Server-Auftrag wird in Altova FlowForce Server erstellt. Im FlowForce Server-Auftrag ist Folgendes definiert: (i) die Inputs und Outputs einer StyleVision Server Transformation; und (ii) die Trigger, wann der Auftrag ausgeführt werden soll, z.B. jeden Tag zu einer bestimmten Zeit. Bei der Ausführung übergibt Altova FlowForce Server die Transformationsanweisungen an StyleVision Server, der dann die Transformation ausführt.

Im unten gezeigten Diagramm sehen Sie die Rolle von StyleVision Server im FlowForce Server Workflow. (Im Diagramm sehen Sie auch die Rolle von MapForce Server im Workflow, da FlowForce Server-Aufträge erstellt werden können, die Altova MapForce Mappings zur Ausführung an den Altova MapForce Server senden.)



StyleVision Server kann nicht nur über einen FlowForce Server-Auftrag, sondern auch über die Befehlszeile aufgerufen werden. Eine Beschreibung dazu finden Sie im Abschnitt StyleVision Server-Befehlszeile [37].

2.2 Als eigenständiger Server

StyleVision Server kann auf Windows-, Linux- und macOS-Systemen als eigenständiges Produkt installiert werden. In dieser Version werden seine Funktionalitäten nur über die Befehlszeile aufgerufen. Eine Beschreibung dazu finden Sie im Abschnitt StyleVision Server-Befehlszeile 37.

© 2019-2025 Altova GmbH

3 Installation und Lizenzierung

In diesem Abschnitt finden Sie Anleitungen zum Installieren, Lizenzieren und Konfigurieren. Es ist in die folgenden Abschnitte gegliedert:

- Einrichten unter Windows 12
- Einrichten unter Linux 20
- Einrichten auf macOS 27
- Upgraden von StyleVision Server 33
- Migrieren von StyleVision Server auf einen neuen Rechner 34

3.1 Einrichten unter Windows

In diesem Abschnitt werden die Installation[12] und Lizenzierung von StyleVision Server auf Windows-Systemen beschrieben. Bei der Einrichtung müssen die folgenden Schritte durchgeführt werden:

1. Installation von StyleVision Server[12]
2. Installation von LicenseServer[15]
3. Starten von LicenseServer und StyleVision Server[16]
4. Registrieren von StyleVision Server auf LicenseServer[17]
5. Zuweisen einer Lizenz zu StyleVision Server[18]

Die einzelnen oben beschriebenen Schritte müssen nicht in genau derselben Reihenfolge wie aufgelistet erfolgen. Bevor Sie beginnen, muss jedoch zunächst die Installation durchgeführt werden. Außerdem muss StyleVision Server auf LicenseServer registriert werden, bevor Sie StyleVision Server über LicenseServer eine Lizenz zuweisen können.

Systemvoraussetzungen (Windows)

Beachten Sie die folgenden Systemvoraussetzungen:

- Windows 10, Windows 11
- Windows Server 2016 oder höher

Voraussetzungen

Beachten Sie dabei die folgenden Voraussetzungen:

- Führen Sie die Installation als Benutzer mit Admin-Rechten durch.
- Ab Version 2021 kann eine 32-Bit-Version von StyleVision Server nicht über eine 64-Bit-Version oder eine 64-Bit-Version nicht über eine 32-Bit-Version installiert werden. Sie müssen (i) die ältere Version entweder vor der Installation der neueren Version entfernen oder (ii) auf eine neuere Version, die dieselbe Bit-Version wie Ihre ältere Version hat, aktualisieren.

3.1.1 Installation unter Windows

Installation von StyleVision Server

StyleVision Server kann auf Windows-Systemen folgendermaßen installiert werden:

- Als separates eigenständiges Server-Produkt. Um StyleVision Server zu installieren, laden Sie das StyleVision Server-Installationsprogramm herunter und starten Sie es. Befolgen Sie die Anweisungen auf dem Bildschirm.
- Um StyleVision Server im Rahmen des FlowForce Server-Pakets zu installieren, laden Sie das FlowForce Server-Installationsprogramm herunter und starten Sie es. Befolgen Sie die Anweisungen auf dem Bildschirm und stellen Sie sicher, dass Sie die Option zur Installation von StyleVision Server aktiviert haben.

Sowohl das Installationsprogramm von StyleVision Server als auch das von FlowForce Server steht im Altova Download Center (https://www.altova.com/de/download.html) zur Verfügung. Sie können im Feld links unten im

© 2019-2025 Altova GmbH

Assistenten die Installationssprache auswählen. Beachten Sie, dass Sie damit auch die Standardsprache von StyleVision Server definieren. Sie können die Sprache später über die Befehlszeile ändern.

Nach der Installation befindet sich die ausführbare StyleVision Server-Datei standardmäßig unter dem folgenden Pfad:

```
<ProgramFilesFolder>\Altova\StyleVisionServer2025\bin\StyleVisionServer.exe
```

Alle erforderlichen Registrierungen zur Verwendung von StyleVision Server über eine COM-Schnittstelle, als Java-Schnittstelle und in der :NET-Umgebung werden vom Installationsprogramm vorgenommen.

Deinstallieren von StyleVision Server

Deinstallieren Sie StyleVision Server folgendermaßen:

1. Klicken Sie mit der rechten Maustaste auf die Windows-Schaltfläche **Start** und wählen Sie **Einstellungen**.
2. Öffnen Sie die Systemsteuerung (Geben Sie die ersten Buchstaben davon ein und klicken Sie auf den Vorschlag).
3. Klicken Sie unter *Programme* auf **Programm deinstallieren**.
4. Wählen Sie in der Systemsteuerung StyleVision Server aus und klicken Sie auf **Deinstallieren**.

Testlizenz

Sie haben während der Installation die Option, eine 30-Tage-Testlizenz für StyleVision Server anzufordern. Die Testlizenz wird nach Erhalt der Anforderung an die von Ihnen registrierte E-Mail-Adresse gesendet.

3.1.2 Installation auf Windows Server Core

Windows Server Core ist eine Windows Minimalinstallation, bei der eine Reihe von Funktionen der Benutzeroberfläche nicht verwendet wird. Sie können StyleVision Server folgendermaßen auf einem Windows Server Core-Rechner installieren:

1. Laden Sie die ausführbare StyleVision Server-Installationsdatei von der Altova Website herunter. Diese Datei hat den Namen **`StyleVisionServer<version>.exe`**. Stellen Sie sicher, dass die ausführbare Datei zu Ihrer Server-Plattform (32-Bit oder 64-Bit) passt.
2. Führen Sie auf einem Standard-Windows-Rechner (und nicht dem Windows Server Core-Rechner) den folgenden Befehl aus: **`StyleVisionServer<version>.exe /u`**. Dadurch wird die **`.msi`**-Datei im selben Ordner entpackt, in dem sich auch die ausführbare Installationsdatei befindet.
3. Kopieren Sie die entpackte **`.msi`**-Datei auf den Windows Server Core-Rechner.
4. Wenn Sie eine frühere Version von StyleVision Server aktualisieren, beenden Sie zuerst StyleVision Server, bevor Sie den nächsten Schritt durchführen.
5. Verwenden Sie die **`.msi`**-Datei für die Installation, indem Sie den Befehl **`msiexec /i StyleVisionServer.msi`** ausführen. Dadurch wird die Installation auf Windows Server Core gestartet.

> **Achtung: Bewahren Sie die .msi-Datei auf!**
>
> Beachten Sie die folgenden Punkte:

- Speichern Sie die extrahierte **.msi**-Datei an einem sicheren Ort. Sie benötigen diese später, um das Produkt zu deinstallieren, zu reparieren oder die Installation anzupassen.
- Wenn Sie die MSI-Datei umbenennen möchten, tun Sie das, bevor Sie StyleVision Server installieren.
- Der Name der MSI-Datei wird in der Registry gespeichert. Sie können den Dateinamen dort aktualisieren, falls er geändert wurde.

Registrieren von StyleVision Server auf LicenseServer

Wenn Sie StyleVision Server zum ersten Mal installieren oder ein Upgrade auf eine **Hauptversion** installieren, müssen Sie StyleVision Server auf einem Altova LicenseServer in Ihrem Netzwerk registrieren. Wenn Sie ein Upgrade auf eine Nicht-Hauptversion von StyleVision Server installieren, kennt das Installationsprogramm die vorherige LicenseServer-Registrierung, daher muss StyleVision Server nicht auf dem LicenseServer registriert werden. Wenn Sie den von StyleVision Server verwendeten LicenseServer jedoch zu irgendeinem Zeitpunkt wechseln möchten, müssen Sie StyleVision Server auf dem neuen LicenseServer registrieren.

Um StyleVision Server bei der Installation auf einem Altova LicenseServer zu registrieren, führen Sie den Installationsbefehl mit der Eigenschaft **REGISTER_WITH-LICENSE_SERVER** aus, wie unten aufgelistet, und geben Sie den Namen oder die Adresse des LicenseServer-Rechners als Wert der Eigenschaft an, z.B:

```
msiexec /i StyleVisionServer.msi REGISTER_WITH_LICENSE_SERVER="localhost"
```

Um StyleVision Server nach der Installation auf einem Altova LicenseServer zu registrieren, starten Sie den folgenden Befehl:

```
msiexec /r StyleVisionServer.msi REGISTER_WITH_LICENSE_SERVER="<MyLS-IPAddress>"
```

Nützliche Befehle

Im Folgenden finden Sie eine Reihe von Befehlen, die im Rahmen der Installation von Nutzen sind.

Den Rückgabewert der Installation können Sie mit einem Skript wie dem folgenden überprüfen. Der Rückgabecode befindet sich in der Umgebungsvariablen `%errorlevel%`. Der Rückgabecode **0** bedeutet, dass der Vorgang erfolgreich war.

```
start /wait msiexec /i StyleVisionServer.msi /q
echo %errorlevel%
```

Damit die Installation im Hintergrund, mit einem Rückgabecode und einem Log des Installationsvorgangs durchgeführt wird, führen Sie den folgenden Befehl aus:

```
start /wait msiexec /i StyleVisionServer.msi /q /L*v! <pathToInstallLogFile>
```

Um die Installation zu ändern, führen Sie den folgenden Befehl aus:

```
msiexec /m StyleVisionServer.msi
```

Um die Installation zu reparieren, führen Sie den folgenden Befehl aus:

```
msiexec /r StyleVisionServer.msi
```

So deinstallieren Sie StyleVision Server:

```
msiexec /x StyleVisionServer.msi
```

Um StyleVision Server im Hintergrund zu deinstallieren und das Resultat in ein detailliertes Log zu schreiben, verwenden Sie folgenden Befehl:

© 2019-2025 Altova GmbH

```
start /wait msiexec /x StyleVisionServer.msi /q /L*v! <pathToUninstallLogFile>
```

Um StyleVision Server in einer anderen Sprache zu installieren (die verfügbaren Sprachcodes sind: Deutsch=de; Spanisch=es; Französisch=fr**)**, verwenden Sie:

```
msiexec /i StyleVisionServer.msi INSTALLER_LANGUAGE=<languageCode>
```

Anmerkung: Auf Windows Server Core stehen die Diagramm- und Barcode-Funktionen von StyleVision Server nicht zur Verfügung.

Anmerkung: Um Taxonomien zu installieren, verwenden Sie den Taxonomiepaket-Manager über die Befehlszeile. Nähere Informationen dazu finden Sie im StyleVision Server-Benutzerhandbuch.

3.1.3 Installation von LicenseServer (Windows)

Damit StyleVision Server ausgeführt werden kann, muss das Produkt über einen Altova LicenseServer in Ihrem Netzwerk lizenziert sein. Wenn Sie StyleVision Server oder FlowForce Server auf Windows-Systemen installieren, können Sie LicenseServer zusammen mit StyleVision Server oder FlowForce Server installieren. Wenn in Ihrem Netzwerk bereits ein LicenseServer installiert ist, muss kein weiterer installiert werden, es sei denn, Sie benötigen eine neuere Version von LicenseServer. (*Siehe nächster Punkt, LicenseServer-Versionen*.)

Aktivieren Sie während der Installation von StyleVision Server oder FlowForce Server je nach Bedarf die Option zum Installieren von LicenseServer.

Beachten Sie die folgenden Punkte:

- Wenn Sie Altova LicenseServer noch nicht installiert haben, belassen Sie die Standardeinstellungen unverändert. Der Assistent installiert daraufhin auf dem Rechner, auf dem Sie den Assistenten gestartet haben, die neueste Version.
- Wenn Sie LicenseServer noch nicht installiert haben und Altova LicenseServer auf einem anderen Rechner installieren möchten und diesen verwenden möchten, deaktivieren Sie das Kontrollkästchen *Altova LicenseServer auf diesem Rechner installieren* und wählen Sie **Später registrieren**. In diesem Fall müssen Sie LicenseServer separat auf dem anderen Rechner installieren und StyleVision Server danach auf dem LicenseServer auf diesem Rechner registrieren.
- Wenn LicenseServer auf Ihrem Rechner bereits installiert wurde, aber eine niedrigere Versionsnummer als die vom Installationsassistenten angegebene hat, belassen Sie die Standardeinstellungen (zur Installation eines Upgrade auf die neuere Version) unverändert. In diesem Fall aktualisiert der Installationsassistent Ihre LicenseServer Version automatisch. Die vorhandenen Registrierungs- und Lizenzierungsdaten werden auf die neue Version von LicenseServer übertragen.
- Wenn LicenseServer bereits auf Ihrem Rechner oder in Ihrem Netzwerk installiert wurde und dieselbe Versionsnummer wie diejenige im Assistenten hat, gehen Sie folgendermaßen vor:
  - Deaktivieren Sie das Kontrollkästchen*LicenseServer auf diesem Rechner installieren.*
  - Wählen Sie unter *Dieses Produkt registrieren auf* den LicenseServer, auf dem Sie StyleVision Server registrieren möchten. Wählen Sie alternativ dazu **Später registrieren**. Beachten Sie, dass Sie immer die Möglichkeit haben, **Später registrieren** auszuwählen, wenn Sie die LicenseServer-Verknüpfungen ignorieren möchten und mit der Installation von StyleVision Server fortfahren möchten.

Lesen Sie nach, wie Sie StyleVision Server auf dem Altova LicenseServer registrieren[17] und lizenzieren[18]. Nähere Informationen dazu finden Sie außerdem in der Dokumentation zu LicenseServer.

LicenseServer-Versionen

- Altova-Produkte müssen entweder (i) mit einer Version von LicenseServer, die der installierten Version von StyleVision Server entspricht oder (ii) mit einer höheren Version von LicenseServer lizenziert werden.
- Die LicenseServer-Version, die der aktuellen Version von StyleVision Server entspricht, ist ***3.17***.
- Unter Windows Sie können die dazugehörige Version von LicenseServer zusammen mit StyleVision Server installieren oder Sie können LicenseServer separat installieren. Auf Linux- und macOS-Systemen muss LicenseServer separat installiert werden.
- Bevor Sie eine neuere Version von LicenseServer installieren, muss eine eventuell vorhandene ältere Version deinstalliert werden.
- Bei der Deinstallation werden alle Registrierungs- und Lizenzierungsinformationen aus der älteren LicenseServer-Version in einer Datenbank auf Ihrem Server gespeichert. Diese Daten werden bei der Installation der neueren Version automatisch in die neuere Version importiert.
- LicenseServer-Versionen sind rückwärts kompatibel. Sie funktionieren auch mit älteren Versionen von StyleVision Server.
- Die neueste Version von LicenseServer steht auf der Altova Website zur Verfügung. Diese Version funktioniert mit allen aktuellen oder älteren Versionen von StyleVision Server.
- Sie finden die Versionsnummer des aktuell installierten LicenseServer am unteren Rand der LicenseServer Konfigurationsseite (alle Register).

3.1.4 Starten von LicenseServer, StyleVision Server (Windows)

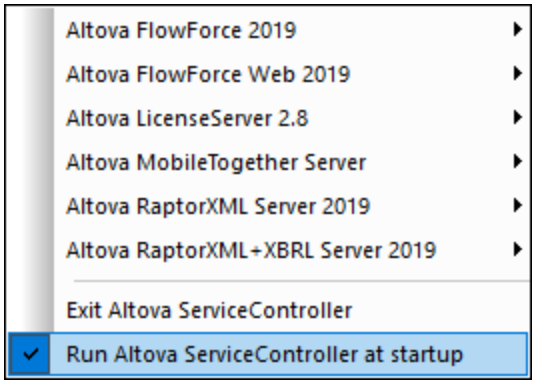
Altova LicenseServer (kurz LicenseServer) und StyleVision Server werden beide über Altova ServiceController gestartet.

Altova ServiceController

Der Altova ServiceController (in der Folge ServiceController genannt) ist eine Applikation, mit der Sie Altova-Dienste **auf Windows-Systemen** starten, beenden und konfigurieren können. ServiceController wird mit Altova LicenseServer und als Dienst installierten Altova Server-Produkten installiert (DiffDog Server, FlowForce Server, Mobile Together Server, and RaptorXML(+XBRL) Server) und kann über die Task-Leiste (*siehe Abbildung unten*) aufgerufen werden.

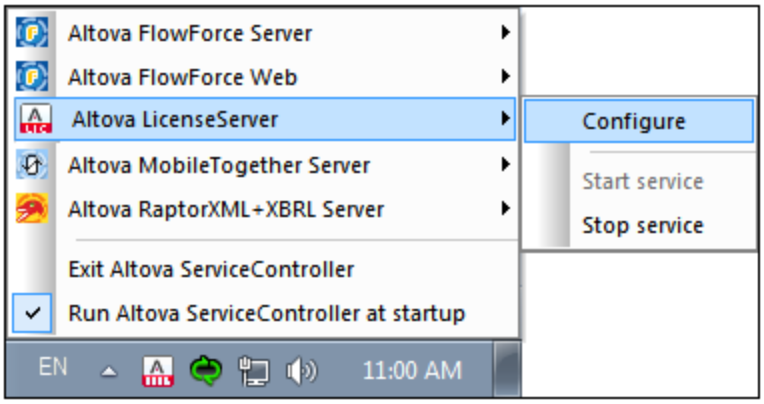
Damit der ServiceController automatisch ausgeführt wird, nachdem sich der Benutzer im System angemeldet hat, klicken Sie in der Task-Leiste auf das **ServiceController**-Symbol, um das **ServiceController**-Menü (*Abbildung unten*) aufzurufen und aktivieren Sie anschließend den Befehl **Run Altova ServiceController at Startup**. (Dieser Befehl ist standardmäßig aktiv). Um den ServiceController zu beenden, klicken Sie in der Task-Leiste auf das **ServiceController**-Symbol und klicken Sie im Menü, das daraufhin angezeigt wird (*Abbildung unten*) auf **Exit Altova ServiceController**.

© 2019-2025 Altova GmbH



Starten von LicenseServer

Um LicenseServer zu starten, klicken Sie in der Task-Leiste auf das **ServiceController**-Symbol, positionieren Sie den Mauszeiger im angezeigten Menü über **Altova LicenseServer** (*siehe Abbildung unten*) und wählen Sie anschließend im LicenseServer Untermenü den Befehl **Start Service**. Wenn LicenseServer bereits ausgeführt wird, ist die Option *Start Service* deaktiviert. Sie können den Dienst über Service Controller auch beenden.



3.1.5 Registrieren von StyleVision Server (Windows)

Damit StyleVision Server über Altova LicenseServer lizenziert werden kann, muss StyleVision Server über einen LicenseServer in Ihrem Netzwerk lizenziert sein. Um StyleVision Server über die Befehlszeilenschnittstelle zu registrieren, verwenden Sie den Befehl `licenseserver` und geben Sie die Adresse des LicenseServer-Rechners an (*siehe unten*).

```
StyleVisionServer licenseserver [options] ServerName-Or-IP-Address
```

Wenn z.B. `localhost` der Name des Servers ist, auf dem LicenseServer installiert ist, verwenden Sie den folgenden Befehl:

```
StyleVisionServer licenseserver localhost
```

Wenn StyleVision Server im Rahmen von FlowForce Server installiert wurde, wird auch StyleVision Server bei der Registrierung von FlowForce Server auf LicenseServer registriert. (i) Starten Sie Altova FlowForce Web als Dienst über den ServiceController (*siehe vorhergehender Punkt*); (ii) Geben Sie Ihr Passwort ein, um die Setup-Seite aufzurufen; (iii) Wählen Sie den Namen oder die Adresse des LicenseServers aus und klicken Sie auf **Auf LicenseServer registrieren**. Nähere Informationen dazu finden Sie unter Registrieren von FlowForce Server.

Nachdem Sie StyleVision Server erfolgreich registriert haben, gehen Sie zum Register "Client Management" der LicenseServer-Konfigurationsseite und weisen Sie dem Produkt eine Lizenz zu.

Nähere Informationen zum Registrieren von Altova-Produkten auf LicenseServer finden Sie im LicenseServer-Benutzerhandbuch.

3.1.6 Zuweisen einer Lizenz (Windows)

Nachdem Sie StyleVision Server erfolgreich registriert haben, wird das Produkt auf dem Register "Client Management" der LicenseServer-Konfigurationsseite aufgelistet. Gehen Sie zu diesem Register und weisen Sie StyleVision Server eine Lizenz zu.

Die Lizenzierung von Altova Server-Produkten basiert auf der Anzahl der auf dem Produktrechner verfügbaren Prozessorkerne. So hat z.B. ein Dual-Core-Prozessor zwei Prozessorkerne, ein Quad-Core-Prozessor vier Kerne, ein Hexa-Core-Prozessor sechs Kerne, usw. Die Anzahl der für ein Produkt lizenzierten Kerne muss größer oder gleich der Anzahl der auf diesem Serverrechner verfügbaren Kerne sein, unabhängig davon, ob es sich um einen physischen Rechner oder eine Virtual Machine handelt. Wenn ein Server z.B. acht Kerne hat (ein Octa-Core-Prozessor), müssen Sie mindestens eine Lizenz für acht Kerne erwerben. Sie können Lizenzen auch kombinieren, um das Produkt für die entsprechende Anzahl von Kernen zu lizenzieren. So können z.B. anstelle einer Lizenz für 8 Kerne auch zwei Lizenzen für 4 Kerne für einen 8-Kern-Prozessor verwendet werden.

Wenn Sie einen Server-Rechner mit einer großen Anzahl von CPU-Kernen verwenden, aber nur geringe Datenmengen verarbeiten müssen, können Sie auch eine Virtual Machine erstellen, der eine geringere Anzahl an Kernen zugewiesen ist und eine Lizenz für diese Anzahl an Kernen erwerben. In diesem Fall ist die Verarbeitungsgeschwindigkeit natürlich geringer als bei Verwendung aller Kerne des Rechners.

Anmerkung: Jede Altova Server-Produktlizenz kann immer nur für einen Client-Rechner gleichzeitig verwendet werden, selbst wenn die Lizenzkapazität dieser Lizenz noch nicht ausgeschöpft ist. (Ein Client-Rechner ist der Rechner, auf dem das Altova Server-Produkt installiert ist.) Wenn z.B. eine 10-Kern-Lizenz für einen Client-Rechner mit 6 CPU-Kernen verwendet wird, so können die verbleibenden Lizenzen für die restlichen 4 Kerne nicht gleichzeitig für einen anderen Client-Rechner verwendet werden.

Single-Thread-Ausführung
Wenn bei einem Altova-Server-Produkt eine Single-Thread-Ausführung möglich ist, so steht eine Option für die *Single-Thread-Ausführung* zur Verfügung. Wenn in solchen Fällen im Lizenzpool eine Altova Serverproduktlizenz für nur einen Prozessorkern verfügbar ist, können Sie einem Rechner mit mehreren Kernen diese Lizenz für einen Kern zuweisen. In diesem Fall führt der Rechner das Produkt an einem einzigen Kern aus. Dadurch verlangsamt sich die Verarbeitungsgeschwindigkeit, da kein Multi-Threading (welches bei mehreren Prozessorkernen möglich wäre) zur Verfügung steht. Das Produkt wird auf diesem Rechner im Single Thread-Modus ausgeführt.

Um einem Mehrkernrechner eine Lizenz für nur einen Kern zuzuweisen, aktivieren Sie in LicenseServer für das

 © 2019-2025 Altova GmbH

entsprechende Produkt das Kontrollkästchen *Limit to single thread execution*.

Schätzung der benötigten Prozessorkerne
Es gibt eine Reihe von externen Faktoren, die das Verarbeitungsvolumen und die Verarbeitungszeiten Ihres Servers beeinflussen (z.B. Hardware, CPU-Auslastung, Arbeitsspeicher für andere auf dem Server laufende Applikationen). Um die Leistung möglichst genau messen zu können, empfiehlt es sich, die Applikationen in Ihrer Umgebung mit möglichst realistischen Datenvolumina und unter möglichst realistischen Bedingungen zu testen.

3.1.7 Zusätzliche Anmerkungen zur Konfiguration (Windows)

Um die im Lieferumfang von StyleVision Server im Unterordner **etc\examples** des Applikationsordners inkludierten Beispiele ausführen zu können, muss die StyleVision Server DLL auf dem System korrekt registriert sein. Oft tritt dann ein Registrierungsfehler auf, wenn Sie im Laufe der Zeit auf demselben Rechner verschiedene Bit-Versionen von StyleVision Server (32-Bit und 64-Bit) installiert haben.

Um die StyleVision Server DLL (entweder die 32-Bit- oder die 64-Bit-Version) auf Windows-Rechnern korrekt zu registrieren, gehen Sie folgendermaßen vor:

1. Öffnen Sie im Administrator-Modus eine Eingabeaufforderung.
2. Wechseln Sie zu dem Ordner, in dem sich die DLL befindet. Der Befehl dazu lautet: **cd C:\Programme\Altova\StyleVisionServer2025\bin**
3. Führen Sie den folgenden Befehl aus, um die DLL (entweder 32-Bit oder 64-Bit) zu registrieren: **regsvr32 StyleVisionServer.dll**
4. Vergewissern Sie sich, dass eine Meldung angezeigt wird, dass die Registrierung erfolgreich war.
5. Öffnen Sie Visual Studio.
6. Laden Sie das Projekt mit Hilfe von **StyleVisionServerAPI_Sample.sln**.
7. Bestätigen Sie, dass Ihre **Program.cs**-Datei gültige Pfade enthält.
8. Starten Sie das Projekt mit Hilfe von **Strg+F5**

Anmerkung: Der Pfad zum Applikationsordner unter Windows ist normalerweise: **C:\Programme\Altova\StyleVisionServer2025**.

3.2 Einrichten unter Linux

In diesem Abschnitt werden die Installation[21] und Lizenzierung von StyleVision Server auf Linux-Systemen (Debian, Ubuntu, CentOS, RedHat) beschrieben. Bei der Einrichtung müssen die folgenden Schritte durchgeführt werden:

1. Installation von StyleVision Server[21]
2. Installation von LicenseServer[22]
3. Starten von LicenseServer[23]
4. Registrieren von StyleVision Server auf LicenseServer[23]
5. Zuweisen einer Lizenz zu StyleVision Server[24]

Die einzelnen oben beschriebenen Schritte müssen nicht in genau derselben Reihenfolge wie aufgelistet erfolgen. Bevor Sie beginnen, muss jedoch zunächst die Installation durchgeführt werden. Außerdem muss StyleVision Server auf LicenseServer registriert werden, bevor Sie StyleVision Server über LicenseServer eine Lizenz zuweisen können.

Systemvoraussetzungen (Linux)

- Red Hat Enterprise Linux 7 oder neuer
- CentOS 7, CentOS Stream 8
- Debian 10 oder neuer
- Ubuntu 20.04, 22.04, 24.04
- AlmaLinux 9.0
- Rocky Linux 9.0

Voraussetzungen

- Führen Sie die Installation entweder als **root**-Benutzer durch oder als Benutzer mit **sudo**-Rechten.
- Die vorherige Version von StyleVision Server muss deinstalliert werden, bevor Sie eine neue Version installieren.
- Wenn Sie beabsichtigen, die Diagrammfunktionalität von Altova zu verwenden, muss auf Ihrem System mindestens eine Schriftart installiert sein, damit die Diagramme korrekt dargestellt werden können. Installierte Schriftarten können Sie z.B. mit dem Befehl **`fc-list`** aus der Fontconfig-Bibliothek auflisten.
- Um die Applikation installieren und ausführen zu können, werden die folgenden Bibliotheken benötigt. Falls die unten angeführten Pakete auf Ihrem Linux-Rechner noch nicht zur Verfügung stehen, führen Sie die Befehl `yum` (oder ggf. `apt-get`) aus, um sie zu installieren.

CentOS, RedHat	Debian	Ubuntu
krb5-libs	libgssapi-krb5-2	libgssapi-krb5-2

© 2019-2025 Altova GmbH

3.2.1 Installation unter Linux

StyleVision Server steht für die Installation auf Linux-Systemen zur Verfügung. Führen Sie die Installation entweder als `root`-Benutzer durch oder als Benutzer mit `sudo`-Rechten.

Integration von FlowForce Server mit anderen Altova Server-Produkten

Wenn Sie StyleVision Server zusammen mit FlowForce Server installieren, sollten Sie zuerst FlowForce Server installieren. Wenn Sie StyleVision Server vor FlowForce Server installiert haben, führen Sie im Anschluss an die Installation von StyleVision Server und FlowForce Server den folgenden Befehl aus:

```
cp /opt/Altova/StyleVisionServer2025/etc/*.tool /opt/Altova/FlowForceServer2025/tool
s
```

Dieser Befehl kopiert die **`.tool`**-Datei aus dem Verzeichnis **`/etc`** von StyleVision Server in das FlowForce Server **`/tools`**-Verzeichnis. Die Datei **`.tool`** wird von FlowForce Server benötigt. Sie enthält den Pfad zur ausführbaren StyleVision Server-Datei. Sie müssen diesen Befehl nicht ausführen, wenn Sie FlowForce Server vor StyleVision Server installieren.

Deinstallieren von StyleVision Server

Bevor Sie StyleVision Server installieren, sollten Sie ältere Versionen deinstallieren.

So überprüfen Sie, welche Altova Server-Produkte aktuell installiert sind:

```
[Debian, Ubuntu]:   dpkg --list | grep Altova
[CentOS, RedHat]:   rpm -qa | grep server
```

So deinstallieren Sie eine alte Version von StyleVision Server:

```
[Debian, Ubuntu]:   sudo dpkg --remove stylevisionserver
[CentOS, RedHat]:   sudo rpm -e stylevisionserver
```

Auf Debian- und Ubuntu-Systemen kann es vorkommen, dass StyleVision Server auch nach seiner Deinstallation noch in der Liste der installierten Produkte angezeigt wird. Führen Sie in diesem Fall den **`purge`**-Befehl aus, um StyleVision Server aus der Liste zu entfernen. Anstelle des oben aufgelisteten Befehls **`remove`** *können Sie auch den* **`purge`**-Befehl verwenden.

```
[Debian, Ubuntu]:   sudo dpkg --purge stylevisionserver
```

Herunterladen des StyleVision Server Linux-Pakets

Auf der Altova Website stehen StyleVision Server-Installationspakete für die folgenden Linux-Systeme zur Verfügung.

Distribution	Paketerweiterung
Debian	`.deb`
Ubuntu	`.deb`

Distribution	Paketerweiterung
CentOS	.rpm
RedHat	.rpm

Nachdem Sie das Linux-Paket heruntergeladen haben, kopieren Sie das Paket in ein beliebiges Verzeichnis auf dem Linux-System. Da StyleVision Server auf einen Altova LicenseServer lizenziert werden muss, benötigen, sollten Sie gleichzeitig mit StyleVision Server auch LicenseServer von der Altova Website herunterladen.

Installieren von StyleVision Server

Wechseln Sie in einem Terminal-Fenster zu dem Verzeichnis, in das Sie das Linux-Paket kopiert haben. Wenn Sie es z.B in ein Benutzerverzeichnis namens `MyAltova` (z.B. im Verzeichnis `/home/User`) kopiert haben, dann wechseln Sie folgendermaßen zu diesem Verzeichnis:

```
cd /home/User/MyAltova
```

Installieren Sie StyleVision Server mit dem entsprechenden Befehl:

```
[Debian]:   sudo dpkg --install stylevisionserver-2025-debian.deb
[Ubuntu]:   sudo dpkg --install stylevisionserver-2025-ubuntu.deb
[CentOS]:   sudo rpm -ivh stylevisionserver-2025-1.x86_64.rpm
[RedHat]:   sudo rpm -ivh stylevisionserver-2025-1.x86_64.rpm
```

Sie müssen den Namen des obigen Pakets eventuell anpassen, damit er der aktuellen Release- oder Service Pack-Version entspricht.

Das StyleVision Server-Paket wird im folgenden Ordner installiert:

```
/opt/Altova/StyleVisionServer2025
```

3.2.2 Installation von LicenseServer (Linux)

Damit StyleVision Server ausgeführt werden kann, muss das Produkt über einen Altova LicenseServer in Ihrem Netzwerk lizenziert sein. Laden Sie Altova LicenseServer von der Altova Website herunter und kopieren Sie das Paket in ein beliebiges Verzeichnis. Installieren Sie es genau wie StyleVision Server (*siehe vorheriges Kapitel* 21).

```
[Debian]:   sudo dpkg --install licenseserver-3.17-debian.deb
[Ubuntu]:   sudo dpkg --install licenseserver-3.17-ubuntu.deb
[CentOS]:   sudo rpm -ivh licenseserver-3.17-1.x86_64.rpm
[RedHat]:   sudo rpm -ivh licenseserver-3.17-1.x86_64.rpm
```

Das LicenseServer-Paket wird im folgenden Ordner installiert:

```
/opt/Altova/LicenseServer
```


© 2019-2025 Altova GmbH

Lesen Sie nach, wie Sie StyleVision Server auf dem Altova LicenseServer registrieren[23] und lizenzieren[24]. Nähere Informationen dazu finden Sie außerdem in der Dokumentation zu LicenseServer.

LicenseServer-Versionen

- Altova-Produkte müssen entweder (i) mit einer Version von LicenseServer, die der installierten Version von StyleVision Server entspricht oder (ii) mit einer höheren Version von LicenseServer lizenziert werden.
- Die LicenseServer-Version, die der aktuellen Version von StyleVision Server entspricht, ist ***3.17***.
- Unter Windows Sie können die dazugehörige Version von LicenseServer zusammen mit StyleVision Server installieren oder Sie können LicenseServer separat installieren. Auf Linux- und macOS-Systemen muss LicenseServer separat installiert werden.
- Bevor Sie eine neuere Version von LicenseServer installieren, muss eine eventuell vorhandene ältere Version deinstalliert werden.
- Bei der Deinstallation werden alle Registrierungs- und Lizenzierungsinformationen aus der älteren LicenseServer-Version in einer Datenbank auf Ihrem Server gespeichert. Diese Daten werden bei der Installation der neueren Version automatisch in die neuere Version importiert.
- LicenseServer-Versionen sind rückwärts kompatibel. Sie funktionieren auch mit älteren Versionen von StyleVision Server.
- Die neueste Version von LicenseServer steht auf der Altova Website zur Verfügung. Diese Version funktioniert mit allen aktuellen oder älteren Versionen von StyleVision Server.
- Sie finden die Versionsnummer des aktuell installierten LicenseServer am unteren Rand der LicenseServer Konfigurationsseite (alle Register).

3.2.3 Starten von LicenseServer, StyleVision Server (LInux)

Starten Sie LicenseServer und StyleVision Server entweder als `root`-Benutzer oder als Benutzer mit `sudo`-Rechten.

Starten von LicenseServer

Um StyleVision Server korrekt auf LicenseServer zu registrieren und zu lizenzieren, muss LicenseServer als Daemon im Netzwerk ausgeführt werden. Starten Sie LicenseServer mit dem folgenden Befehl als Daemon:

```
sudo systemctl start licenseserver
```

(Um LicenseServer zu beenden, ersetzen Sie im obigen Befehl **`start`** durch **`stop`**.) Beispiel:

```
sudo systemctl stop licenseserver
```

3.2.4 Registrieren von StyleVision Server (Linux)

Damit StyleVision Server über Altova LicenseServer lizenziert werden kann, muss StyleVision Server über einen LicenseServer in Ihrem Netzwerk lizenziert sein.

Um StyleVision Server zu registrieren, rufen Sie seine CLI auf und verwenden Sie den Befehl `licenseserver`:

```
sudo /opt/Altova/StyleVisionServer2025/bin/stylevisionserver licenseserver [options]
ServerName-Or-IP-Address
```

Wenn z.B. `localhost` der Name des Servers ist, auf dem LicenseServer installiert ist:

```
sudo /opt/Altova/StyleVisionServer2025/bin/stylevisionserver licenseserver localhost
```

Im obigen Befehl ist `localhost` der Name des Servers, auf dem LicenseServer installiert ist. Beachten Sie, dass der Pfad der ausführbaren StyleVision Server-Datei der folgende ist:

```
/opt/Altova/StyleVisionServer2025/bin/
```

Nachdem Sie StyleVision Server erfolgreich registriert haben, gehen Sie zum Register "Client Management" der LicenseServer-Konfigurationsseite und weisen Sie dem Produkt eine Lizenz zu.

Nähere Informationen zum Registrieren von Altova-Produkten auf LicenseServer finden Sie im LicenseServer-Benutzerhandbuch.

3.2.5 Zuweisen einer Lizenz (LInux)

Nachdem Sie StyleVision Server erfolgreich registriert haben, wird das Produkt auf dem Register "Client Management" der LicenseServer-Konfigurationsseite aufgelistet. Gehen Sie zu diesem Register und weisen Sie StyleVision Server eine Lizenz zu.

Die Lizenzierung von Altova Server-Produkten basiert auf der Anzahl der auf dem Produktrechner verfügbaren Prozessorkerne. So hat z.B. ein Dual-Core-Prozessor zwei Prozessorkerne, ein Quad-Core-Prozessor vier Kerne, ein Hexa-Core-Prozessor sechs Kerne, usw. Die Anzahl der für ein Produkt lizenzierten Kerne muss größer oder gleich der Anzahl der auf diesem Serverrechner verfügbaren Kerne sein, unabhängig davon, ob es sich um einen physischen Rechner oder eine Virtual Machine handelt. Wenn ein Server z.B. acht Kerne hat (ein Octa-Core-Prozessor), müssen Sie mindestens eine Lizenz für acht Kerne erwerben. Sie können Lizenzen auch kombinieren, um das Produkt für die entsprechende Anzahl von Kernen zu lizenzieren. So können z.B. anstelle einer Lizenz für 8 Kerne auch zwei Lizenzen für 4 Kerne für einen 8-Kern-Prozessor verwendet werden.

Wenn Sie einen Server-Rechner mit einer großen Anzahl von CPU-Kernen verwenden, aber nur geringe Datenmengen verarbeiten müssen, können Sie auch eine Virtual Machine erstellen, der eine geringere Anzahl an Kernen zugewiesen ist und eine Lizenz für diese Anzahl an Kernen erwerben. In diesem Fall ist die Verarbeitungsgeschwindigkeit natürlich geringer als bei Verwendung aller Kerne des Rechners.

Anmerkung: Jede Altova Server-Produktlizenz kann immer nur für einen Client-Rechner gleichzeitig verwendet werden, selbst wenn die Lizenzkapazität dieser Lizenz noch nicht ausgeschöpft ist. (Ein Client-Rechner ist der Rechner, auf dem das Altova Server-Produkt installiert ist.) Wenn z.B. eine 10-Kern-Lizenz für einen Client-Rechner mit 6 CPU-Kernen verwendet wird, so können die verbleibenden Lizenzen für die restlichen 4 Kerne nicht gleichzeitig für einen anderen Client-Rechner verwendet werden.

Single-Thread-Ausführung
Wenn bei einem Altova-Server-Produkt eine Single-Thread-Ausführung möglich ist, so steht eine Option für die *Single-Thread-Ausführung* zur Verfügung. Wenn in solchen Fällen im Lizenzpool eine Altova Serverproduktlizenz für nur einen Prozessorkern verfügbar ist, können Sie einem Rechner mit mehreren Kernen diese Lizenz für einen Kern zuweisen. In diesem Fall führt der Rechner das Produkt an einem einzigen Kern aus. Dadurch verlangsamt sich die Verarbeitungsgeschwindigkeit, da kein Multi-Threading (welches bei

 © 2019-2025 Altova GmbH

mehreren Prozessorkernen möglich wäre) zur Verfügung steht. Das Produkt wird auf diesem Rechner im Single Thread-Modus ausgeführt.

Um einem Mehrkernrechner eine Lizenz für nur einen Kern zuzuweisen, aktivieren Sie in LicenseServer für das entsprechende Produkt das Kontrollkästchen *Limit to single thread execution*.

Schätzung der benötigten Prozessorkerne
Es gibt eine Reihe von externen Faktoren, die das Verarbeitungsvolumen und die Verarbeitungszeiten Ihres Servers beeinflussen (z.B. Hardware, CPU-Auslastung, Arbeitsspeicher für andere auf dem Server laufende Applikationen). Um die Leistung möglichst genau messen zu können, empfiehlt es sich, die Applikationen in Ihrer Umgebung mit möglichst realistischen Datenvolumina und unter möglichst realistischen Bedingungen zu testen.

3.2.6 Anmerkungen zur Umgebung (Linux)

Ordner

Im Folgenden finden Sie eine Liste wichtiger Ordner in Ihrer StyleVision Server-Installation.

- *Root-Ordner der Installation*

  /opt/Altova/StyleVisionServer2025/

- *Lizenzdateien*

  /var/opt/Altova/StyleVisionServer

- *Umgebungseinstellungen*

  /etc/profile.d/jdbc.sh
  Die Datei für die Umgebungseinstellungen (normalerweise mit dem Namen `jdbc.sh`) wird beim Systemstart ausgeführt. Die Definitionen darin müssen entsprechend Ihrer jeweiligen Umgebung konfiguriert sein. Der obige Pfad dient nur als allgemeines Beispiel.
  Anmerkung: Die Datei für die Umgebungseinstellungen definiert die Variablen für **alle Benutzer** im System, seien Sie daher vorsichtig, wenn Sie die Einstellungen ändern. Wenn Sie z.B. einen Klassenpfad in dieser Datei ändern, wird diese Änderung systemweit wirksam. Wenn Sie nur Änderungen für StyleVision Server vornehmen möchten, sollten Sie eventuell eine Unit-Datei verwenden (Eine Beschreibung dazu finden Sie weiter unten unter *JDBC-Verbindungen*).

Datenbankverbindungen

Unter Linux werden die folgenden Datenbankverbindungen unterstützt:

- JDBC - Sie können JDBC für alle unterstützten Datenbanken mit Ausnahme von Microsoft Access verwenden
- Native Verbindungen - stehen derzeit für SQLite- und PostgreSQL-Datenbanken zur Verfügung

Beachten Sie die folgenden Punkte, wenn Sie JDBC verwenden:

- Das Java Runtime Environment oder SDK muss installiert sein
- Die JDBC-Treiber für die Zieldatenbank müssen installiert sein

- Die folgenden Umgebungsvariablen müssen für Ihre Umgebung richtig konfiguriert werden:
  - `CLASSPATH`: Um die jar-Dateien für die Verbindung zur JDBC-Datenbank zu finden. Die jar-Dateien, können entweder (i) in ein ausführbares Skript (wie `jdbc.sh`), das bei Systemstart ausgeführt wird, eingegeben werden oder (ii) in eine Unit-Datei, die ausgeführt wird, wenn StyleVision Server als Dienst gestartet wird. Die Verwendung einer Unit-Datei zur Definition der jar-Dateien hat den Vorteil, dass die für die JDBC-Verbindungen von StyleVision Server benötigten jar-Dateien gefunden werden, ohne dass Sie die bestehende Systemkonfiguration ändern müssen. Unten sehen Sie den Inhalt einer Unit-Datei.
  - `PATH`: Um die JRE zu finden. Nicht unbedingt erforderlich, hängt von der Installation ab
  - `JAVA_HOME`: Falls nötig - abhängig von der Installation.

Liste wichtiger Dateien

Das folgende Shell-Skript (oder Unit-Datei) wird in den Ordner **`/opt/Altova/StyleVisionServer/etc`** kopiert, damit bereits vorhandene Konfigurationsdateien nicht überschrieben werden. Siehe auch Abschnitt *JDBC-Verbindungen* weiter oben. Nehmen Sie die nötigen Änderungen vor, wie erforderlich. Die blau markierten Teile sind umgebungsspezifisch und müssen an Ihre individuelle Umgebung angepasst werden.

Shell-Skript (Unit-Datei)

```
#- jdbc - environment -
export PATH=/usr/local/jdk1.7.0_17/bin:/usr/lib64/qt-
3.3/bin:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/qa/bin
export JAVA_HOME=/usr/local/jdk1.7.0_17
export
CLASSPATH=/usr/local/jdbc/oracle/ojdbc6.jar:/usr/local/jdbc/oracle/xdb.jar:/usr/local/j
dbc/oracle/xmlparserv2.jar:/usr/local/jdbc/postgre/postgresql-9.0-
801.jdbc4.jar:/usr/local/jdbc/mssql/sqljdbc4.jar:/usr/local/jdbc/iseries/lib/jt400.jar:
/usr/local/jdbc/mysql/mysql-connector-java-5.1.16-
bin.jar:/usr/local/jdbc/sqlite/sqlitejdbc-
v056.jar:/usr/local/jdbc/Informix_JDBC_Driver/lib/ifxjdbc.jar:/usr/local/jdbc/sybase/jc
onn7/jconn4.jar:/usr/local/jdbc/db2/db2jcc.jar:/usr/local/jdbc/db2/db2jcc_license_cu.ja
r:./:
```


© 2019-2025 Altova GmbH

3.3 Einrichten auf macOS

In diesem Abschnitt werden die Installation[27] und Lizenzierung von StyleVision Server auf macOS-Systemen beschrieben. Bei der Einrichtung müssen die folgenden Schritte durchgeführt werden:

1. Installation von StyleVision Server[27]
2. Installation von LicenseServer[29]
3. Starten von LicenseServer[29]
4. Registrieren von StyleVision Server auf LicenseServer[30]
5. Zuweisen einer Lizenz zu StyleVision Server[30]

Die einzelnen oben beschriebenen Schritte müssen nicht in genau derselben Reihenfolge wie aufgelistet erfolgen. Bevor Sie beginnen, muss jedoch zunächst die Installation durchgeführt werden. Außerdem muss StyleVision Server auf LicenseServer registriert werden, bevor Sie StyleVision Server über LicenseServer eine Lizenz zuweisen können.

Systemvoraussetzungen (macOS)

Beachten Sie die folgende Systemvoraussetzung:

- macOS 12 oder neuer

Voraussetzungen

Beachten Sie dabei die folgenden Voraussetzungen:

- Stellen Sie sicher, dass Altova LicenseServer installiert wurde und ausgeführt wird.
- Führen Sie die Installation entweder als `root`-Benutzer durch oder als Benutzer mit `sudo`-Rechten.
- Die vorherige Version von StyleVision Server muss deinstalliert werden, bevor Sie eine neue Version installieren.
- Wenn Sie beabsichtigen, die Diagrammfunktionalität von Altova zu verwenden, muss auf Ihrem System mindestens eine Schriftart installiert sein, damit die Diagramme korrekt dargestellt werden können. Installierte Schriftarten können Sie z.B. mit dem Befehl **`fc-list`** aus der Fontconfig-Bibliothek auflisten.
- Der macOS-Rechner muss so konfiguriert sein, dass sein Name zu einer IP-Adresse aufgelöst wird (d.h. der Host-Name muss vom Terminal aus mit dem Befehl **`ping <hostname>`** erfolgreich angepingt werden können).

3.3.1 Installation auf macOS

In diesem Kapitel werden die Installation und Konfiguration von StyleVision Server auf macOS-Systemen beschrieben.

Integration mit FlowForce

Wenn Sie StyleVision Server zusammen mit FlowForce Server installieren, sollten Sie zuerst FlowForce Server installieren. Wenn Sie StyleVision Server vor FlowForce Server installiert haben, führen Sie im Anschluss an die Installation der beiden Produkte den folgenden Befehl aus:

```
cp /usr/local/Altova/StyleVisionServer2025/etc/*.tool /usr/local/Altova/FlowForceServer2025
/tools
```

Dieser Befehl kopiert die **.tool**-Datei aus dem Verzeichnis **/etc** von StyleVision Server in das FlowForce Server **/tools**-Verzeichnis. Die Datei **.tool** wird von FlowForce Server benötigt. Sie enthält den Pfad zur ausführbaren StyleVision Server-Datei. Sie müssen diesen Befehl nicht ausführen, wenn Sie FlowForce Server vor StyleVision Server installieren.

Deinstallieren von StyleVision Server

Klicken Sie im Finder unter Applications mit der rechten Maustaste auf das StyleVision Server-Symbol und wählen Sie den Befehl **Move to Trash**. Die Applikation wird daraufhin in den Papierkorb verschoben. Sie müssen die Applikation jetzt noch aus dem Ordner `usr` entfernen. Führen Sie dazu den folgenden Befehl aus:

```
sudo rm -rf /usr/local/Altova/StyleVisionServer2025/
```

Wenn Sie eine ältere Version von Altova LicenseServer deinstallieren müssen, beenden Sie den Dienst zuerst mit dem folgenden Befehl:

```
sudo launchctl unload /Library/LaunchDaemons/com.altova.LicenseServer.plist
```

Um zu überprüfen, ob ein Dienst beendet wurde, öffnen Sie den Activity Monitor im Finder und stellen Sie sicher, dass sich LicenseServer nicht in der Liste befindet. Fahren Sie anschließend mit der Deinstallation fort, wie oben für StyleVision Server beschrieben.

Installieren von StyleVision Server

Um StyleVision Server zu installieren, gehen Sie folgendermaßen vor:

1. Laden Sie die **.dmg** (Disk Image)-Datei von StyleVision Server von der Altova Website (https://www.altova.com/de/download.html) in ein lokales Verzeichnis herunter.
2. Klicken Sie auf die heruntergeladene Disk Image-Datei (**.dmg**), um sie zu öffnen. Dadurch wird der StyleVision Server Installer als neues virtuelles Laufwerk auf Ihrem Computer angezeigt.
3. Doppelklicken Sie in diesem neuen virtuellen Laufwerk auf das Installer-Paket (**.pkg**).
4. Befolgen Sie die selbsterklärenden Anweisungen des Installationsassistenten, in dem Sie auch die Lizenzvereinbarung akzeptieren müssen, bevor Sie mit der Installation fortfahren können.
5. Um das Laufwerk nach der Installation auszuwerfen, klicken Sie mit der rechten Maustaste darauf und wählen Sie **Eject**.

Das StyleVision Server-Paket wird im folgenden Ordner installiert:

/usr/local/Altova/StyleVisionServer2025 (Applikationsbinärdateien)
/var/Altova/StyleVisionServer (Datendateien: Datenbank und Logs)

Der StyleVision Server Server-Daemon wird nach einer Installation und einem Neustart des Rechners automatisch gestartet. Sie können StyleVision Server jederzeit mit dem folgenden Befehl als Daemon starten:

```
sudo launchctl load /Library/LaunchDaemons/com.altova.StyleVisionServer2025.plist
```


© 2019-2025 Altova GmbH

3.3.2 Installation von LicenseServer (macOS)

Altova LicenseServer kann von der Altova Website (https://www.altova.com/de/download.html) heruntergeladen werden. Führen Sie die Installation durch, wie hier 27 beschrieben.

Das LicenseServer-Paket wird im folgenden Ordner installiert:

```
/usr/local/Altova/LicenseServer
```

Lesen Sie nach, wie Sie StyleVision Server auf dem Altova LicenseServer registrieren 30 und lizenzieren 30. Nähere Informationen dazu finden Sie außerdem in der Dokumentation zu LicenseServer.

LicenseServer-Versionen

- Altova-Produkte müssen entweder (i) mit einer Version von LicenseServer, die der installierten Version von StyleVision Server entspricht oder (ii) mit einer höheren Version von LicenseServer lizenziert werden.
- Die LicenseServer-Version, die der aktuellen Version von StyleVision Server entspricht, ist ***3.17***.
- Unter Windows Sie können die dazugehörige Version von LicenseServer zusammen mit StyleVision Server installieren oder Sie können LicenseServer separat installieren. Auf Linux- und macOS-Systemen muss LicenseServer separat installiert werden.
- Bevor Sie eine neuere Version von LicenseServer installieren, muss eine eventuell vorhandene ältere Version deinstalliert werden.
- Bei der Deinstallation werden alle Registrierungs- und Lizenzierungsinformationen aus der älteren LicenseServer-Version in einer Datenbank auf Ihrem Server gespeichert. Diese Daten werden bei der Installation der neueren Version automatisch in die neuere Version importiert.
- LicenseServer-Versionen sind rückwärts kompatibel. Sie funktionieren auch mit älteren Versionen von StyleVision Server.
- Die neueste Version von LicenseServer steht auf der Altova Website zur Verfügung. Diese Version funktioniert mit allen aktuellen oder älteren Versionen von StyleVision Server.
- Sie finden die Versionsnummer des aktuell installierten LicenseServer am unteren Rand der LicenseServer Konfigurationsseite (alle Register).

3.3.3 Starten von LicenseServer, StyleVision Server (macOS)

Starten Sie LicenseServer und StyleVision Server entweder als `root`-Benutzer oder als Benutzer mit `sudo`-Rechten.

Starten von LicenseServer

Um StyleVision Server korrekt auf LicenseServer zu registrieren und zu lizenzieren, muss LicenseServer als Daemon ausgeführt werden. Starten Sie LicenseServer mit dem folgenden Befehl als Daemon:

```
sudo launchctl load /Library/LaunchDaemons/com.altova.LicenseServer.plist
```

Um LicenseServer zu beenden, ersetzen Sie im obigen Befehl **`load`** durch **`unload`**.)

3.3.4 Registrieren von StyleVision Server (macOS)

Damit StyleVision Server über Altova LicenseServer lizenziert werden kann, muss StyleVision Server über einen LicenseServer in Ihrem Netzwerk lizenziert sein.

Um StyleVision Server über die Befehlszeile zu registrieren, verwenden Sie den Befehl `licenseserver`:

```
sudo /usr/local/Altova/StyleVisionServer2025/bin/StyleVisionServer licenseserver
[options] ServerName-Or-IP-Address
```

Wenn z.B. `localhost` der Name des Servers ist, auf dem LicenseServer installiert ist:

```
sudo /usr/local/Altova/StyleVisionServer2025/bin/StyleVisionServer licenseserver
localhost
```

Im obigen Befehl ist `localhost` der Name des Servers, auf dem LicenseServer installiert ist. Beachten Sie, dass der Pfad der ausführbaren StyleVision Server-Datei der folgende ist:

```
/usr/local/Altova/StyleVisionServer2025/bin/
```

Nachdem Sie StyleVision Server erfolgreich registriert haben, gehen Sie zum Register "Client Management" der LicenseServer-Konfigurationsseite und weisen Sie dem Produkt eine Lizenz zu.

Nähere Informationen zum Registrieren von Altova-Produkten auf LicenseServer finden Sie im LicenseServer-Benutzerhandbuch.

3.3.5 Zuweisen einer Lizenz (macOS)

Nachdem Sie StyleVision Server erfolgreich registriert haben, wird das Produkt auf dem Register "Client Management" der LicenseServer-Konfigurationsseite aufgelistet. Gehen Sie zu diesem Register und weisen Sie StyleVision Server eine Lizenz zu.

Die Lizenzierung von Altova Server-Produkten basiert auf der Anzahl der auf dem Produktrechner verfügbaren Prozessorkerne. So hat z.B. ein Dual-Core-Prozessor zwei Prozessorkerne, ein Quad-Core-Prozessor vier Kerne, ein Hexa-Core-Prozessor sechs Kerne, usw. Die Anzahl der für ein Produkt lizenzierten Kerne muss größer oder gleich der Anzahl der auf diesem Serverrechner verfügbaren Kerne sein, unabhängig davon, ob es sich um einen physischen Rechner oder eine Virtual Machine handelt. Wenn ein Server z.B. acht Kerne hat (ein Octa-Core-Prozessor), müssen Sie mindestens eine Lizenz für acht Kerne erwerben. Sie können Lizenzen auch kombinieren, um das Produkt für die entsprechende Anzahl von Kernen zu lizenzieren. So können z.B. anstelle einer Lizenz für 8 Kerne auch zwei Lizenzen für 4 Kerne für einen 8-Kern-Prozessor verwendet werden.

Wenn Sie einen Server-Rechner mit einer großen Anzahl von CPU-Kernen verwenden, aber nur geringe Datenmengen verarbeiten müssen, können Sie auch eine Virtual Machine erstellen, der eine geringere Anzahl an Kernen zugewiesen ist und eine Lizenz für diese Anzahl an Kernen erwerben. In diesem Fall ist die Verarbeitungsgeschwindigkeit natürlich geringer als bei Verwendung aller Kerne des Rechners.

Anmerkung: Jede Altova Server-Produktlizenz kann immer nur für einen Client-Rechner gleichzeitig verwendet werden, selbst wenn die Lizenzkapazität dieser Lizenz noch nicht ausgeschöpft ist. (Ein Client-Rechner ist der Rechner, auf dem das Altova Server-Produkt installiert ist.) Wenn z.B. eine 10-Kern-Lizenz für einen Client-

© 2019-2025 Altova GmbH

Rechner mit 6 CPU-Kernen verwendet wird, so können die verbleibenden Lizenzen für die restlichen 4 Kerne nicht gleichzeitig für einen anderen Client-Rechner verwendet werden.

Single-Thread-Ausführung
Wenn bei einem Altova-Server-Produkt eine Single-Thread-Ausführung möglich ist, so steht eine Option für die *Single-Thread-Ausführung* zur Verfügung. Wenn in solchen Fällen im Lizenzpool eine Altova Serverproduktlizenz für nur einen Prozessorkern verfügbar ist, können Sie einem Rechner mit mehreren Kernen diese Lizenz für einen Kern zuweisen. In diesem Fall führt der Rechner das Produkt an einem einzigen Kern aus. Dadurch verlangsamt sich die Verarbeitungsgeschwindigkeit, da kein Multi-Threading (welches bei mehreren Prozessorkernen möglich wäre) zur Verfügung steht. Das Produkt wird auf diesem Rechner im Single Thread-Modus ausgeführt.

Um einem Mehrkernrechner eine Lizenz für nur einen Kern zuzuweisen, aktivieren Sie in LicenseServer für das entsprechende Produkt das Kontrollkästchen *Limit to single thread execution*.

Schätzung der benötigten Prozessorkerne
Es gibt eine Reihe von externen Faktoren, die das Verarbeitungsvolumen und die Verarbeitungszeiten Ihres Servers beeinflussen (z.B. Hardware, CPU-Auslastung, Arbeitsspeicher für andere auf dem Server laufende Applikationen). Um die Leistung möglichst genau messen zu können, empfiehlt es sich, die Applikationen in Ihrer Umgebung mit möglichst realistischen Datenvolumina und unter möglichst realistischen Bedingungen zu testen.

3.3.6 Anmerkungen zur Umgebung (macOS)

Ordner

Im Folgenden finden Sie eine Liste wichtiger Ordner in Ihrer StyleVision Server-Installation.

- *Root-Ordner der Installation*

  `/usr/local/Altova/StyleVisionServer2025/`

- *Lizenzdateien*

  `/var/Altova/StyleVisionServer`

- *Umgebungseinstellungen*

  `/Library/LaunchDaemons/com.altova.StyleVisionServer.plist`
  Die Datei für die Umgebungseinstellungen muss entsprechend Ihrer jeweiligen Umgebung konfiguriert sein. Der obige Pfad dient nur als allgemeines Beispiel.
  Anmerkung: Diese Umgebungsvariablen sind nur für den StyleVision Server Prozess definiert und haben keine Auswirkung auf andere Benutzer.

Datenbankverbindungen

Auf MacOS werden die folgenden Datenbankverbindungen unterstützt:

- JDBC - Sie können JDBC für alle unterstützten Datenbanken mit Ausnahme von Microsoft Access verwenden.
- Native Verbindungen - stehen derzeit für SQLite- und PostgreSQL-Datenbanken zur Verfügung.

Beachten Sie die folgenden Punkte, wenn Sie JDBC verwenden:

- Das Java Runtime Environment oder SDK muss installiert sein.
- Die JDBC-Treiber für die Zieldatenbank müssen installiert sein.
- Die folgenden Umgebungsvariablen müssen für Ihre Umgebung richtig konfiguriert werden:
  - **CLASSPATH:** Um die jar-Dateien zu finden. Der Class Path wird in der Datei `Plist` definiert.
  - **PATH:** Um die JRE zu finden. Nicht unbedingt erforderlich, hängt von der Installation ab.
  - **JAVA_HOME:** Falls nötig - abhängig von der Installation.

Java 6 für ausfüllbare PDF-Formulare

Wenn die ausfüllbaren Bereiche eines ausfüllbaren PDF-Dokuments fehlen, wenn das PDF-Dokument auf einem macOS-System geöffnet wird, ist die Ursache wahrscheinlich, dass Java 6 auf dem Rechner nicht installiert ist. Installieren Sie in diesem Fall Java 6 von https://support.apple.com/kb/dl1572?locale=en_US. Wenn bereits eine neuere Version als Java 6 installiert ist, hat die Installation der älteren Java 6-Version keine Auswirkung auf das Funktionieren der neuen Version, die weiterhin als die Standardversion des Systems verwendet wird.

© 2019-2025 Altova GmbH

3.4 Upgraden von StyleVision Server

Am einfachsten lässt sich die Lizenz aus der vorherigen Version von StyleVision Server bei der Installation auf die neuere Version übertragen: Die wichtigsten Schritte bei der Installation sind die folgenden:

1. Registrieren Sie die neue Version von StyleVision Server auf dem LicenseServer, auf dem sich die Lizenz der älteren Version von StyleVision Server befindet.
2. Akzeptieren Sie die Lizenzvereinbarung von StyleVision Server. (Wenn Sie der Vereinbarung nicht zustimmen, wird die neue Version nicht installiert.)

Anmerkung: Wenn Sie StyleVision Server nicht während der Installation auf LicenseServer registrieren, können Sie dies später nachholen und die Lizenzierung erst dann abschließen.

3.5 Migrieren von StyleVision Server auf einen neuen Rechner

Wenn Sie StyleVision Server von einem Rechner auf einen anderen (eventuell auch auf eine andere Plattform) migrieren möchten, befolgen Sie die Richtlinien weiter unten.

Bei der Migration von StyleVision Server auf einen neuen Rechner wird die Lizenz vom alten Rechner einem neuen Rechner zugewiesen. Gehen Sie dazu folgendermaßen vor:

1. Installieren Sie StyleVision Server auf dem neuen Rechner. Wenn das Produkt bereits im Rahmen der FlowForce Server-Installation installiert wurde, ignorieren Sie diesen Schritt.
2. Registrieren Sie StyleVision Server auf dem neuen Rechner auf Altova LicenseServer.
3. Stellen Sie sicher, dass der Server auf dem alten Rechner von keinen Clients verwendet wird.
4. Öffnen Sie die Altova LicenseServer-Verwaltungsseite. Deaktivieren Sie die Lizenz des alten StyleVision Server Rechners und weisen Sie sie dem neuen Rechner zu.

Anmerkung: Wenn Sie auf dem alten Rechner XML-Kataloge verwendet haben, migrieren Sie diese auf den neuen Rechner.

© 2019-2025 Altova GmbH

3.6 FOP-Anforderungen

StyleVision Server verwendet bei der Generierung von PDF-Dokumenten mit seinem **generate**-Befehl standardmäßig den FO-Prozessor des Apache FOP-Projekts. (Sie können mit dem Befehl **setfopath** einen anderen FO-Prozessor auswählen).

FOP wird zusammen mit StyleVision Server installiert. Auf Windows-Systemen wird FOP im Ordner **C:\ProgramData\Altova\SharedBetweenVersions** installiert. Auf Linux- und macOS-Systemen wird der Prozessor in einem Unterordner von **StyleVisionServer2025** installiert.

Informationen zur Ausführung von FOP 2.9 finden Sie auf der Apache.org-Website im diesbezüglichen Kapitel. (Wenn Sie eine andere Version von FOP verwenden, rufen Sie die entsprechende Seite für diese FOP-Version auf).

Java Runtime Environment

Um PDFs erstellen zu können, muss Java Runtime Environment (JRE) auf Ihrem Rechner installiert sein. Überprüfen Sie auf der Apache.org-Website, welche JRE-Version für Ihre Version von FOP mindestens benötigt wird. So ist etwa für Version FOP 2.8 eine JRE-Version ab 1.8 erforderlich. Wenn JRE nicht installiert ist und Sie versuchen, ein PDF-Dokument mit Hilfe von FOP zu generieren, wird eine Meldung angezeigt, dass beim Aufruf von FOP ein Fehler aufgetreten ist.

Einrichten von OpenJDK
StyleVision Server wird regelmäßig mit der neuesten verfügbaren OpenJDK-Version getestet. Zum Zeitpunkt der Verfassung ist die neueste Version 19.0.2. Nachdem Sie Ihr OpenJDK-Paket heruntergeladen und entpackt haben, (i) müssen Sie den **java\bin**-Pfad zu den Systempfaden hinzufügen und (ii) die **JAVA_HOME**-Umgebungsvariable konfigurieren.

Auf Windows-Systemen konfigurieren Sie die **JAVA_HOME**-Umgebungsvariable folgendermaßen:

1. Klicken Sie in der Windows-Taskleiste mit der rechten Maustaste auf das Windows-Symbol und wählen Sie *System*.
2. Klicken Sie im Fenster "Einstellungen" unter *Verwandte Einstellungen* auf *Erweiterte Systemeinstellungen*.
3. Klicken Sie auf dem Register *Erweitert* auf *Umgebungsvariablen.*
4. Klicken Sie auf "Neu", um eine neue Umgebungsvariable zu erstellen und geben Sie ihr den Namen **JAVA_HOME**.
5. Definieren Sie den Wert der Umgebungsvariablen. Setzen Sie den Wert für eine**C:\Programme\openjdk\bin\java.exe** z.B. auf **C:\Programme\openjdk**.

3.7 Sicherheitsaspekte

XSLT, XPath und XQuery sind Turing-vollständige funktionale Programmiersprachen mit Zugriff auf lokale und entfernte Dateien und der Möglichkeit zur dynamischen Ausführung. Es wird daher empfohlen, den Zugriff auf diese Dateien für Transformationen und/oder die Dateiverarbeitung nur in einer sicheren und geregelten Umgebung, in der die Kontrolle über die Input-Dateien gewährleistet ist und sichergestellt werden kann, dass nur zuvor überprüfte Scripts ausgeführt werden, zu gestatten. Sollte die Notwendigkeit bestehen, diese Dateien von einem externen/öffentlichen Netzwerk (oder einem nicht sicheren Sub-Netzwerk) aufzurufen, wird empfohlen, den Zugriff mit Hilfe eines Reverse Proxy Servers, der eine Benutzerauthentifizierung und -autorisierung implementiert, einzuschränken. Außerdem wird empfohlen, den Vorgang über ein separates Benutzerkonto auszuführen, bei dem der Zugriff auf Betriebssystemebene konfiguriert ist, um den Zugriff auf die autorisierten Bereiche des Dateisystems einzuschränken.

© 2019-2025 Altova GmbH

4 StyleVision Server-Befehlszeile

Standardpfad der ausführbaren StyleVision Server-Datei

Im Folgenden finden Sie die jeweiligen Standardpfade der ausführbaren StyleVision Server-Datei:

Linux	`/opt/Altova/StyleVisionServer2025/bin/`**`stylevisionserver`**
Mac	`/usr/local/Altova/StyleVisionServer2025/bin/`**`stylevisionserver`**
Windows	`<`*`ProgramFilesFolder`*`>\Altova\StyleVisionServer2025\bin\`**`StyleVisionServer.exe`**

Verwendung und Liste der CLI-Befehle

Die Befehlszeilensyntax lautet:

```
stylevisionserver --h | --help | --version | <command> [options] [arguments]
```

- `--help` (Kurzform `--h`) Zeigt den Hilfetext zum jeweiligen Befehl an. Wenn kein Befehl angegeben ist, werden alle Befehle der ausführbaren Datei mit jeweils einer kurzen Beschreibung des Befehls aufgelistet.
- `--version` Zeigt die Versionsnummer von StyleVision Server an.
- `<command>` ist der auszuführende Befehl. Die Befehle sind in den Unterabschnitten dieses Abschnitts beschrieben (*siehe Liste unten*).
- `[options]` sind die Optionen eines Befehls. Diese werden mit ihren jeweiligen Befehlen aufgelistet und beschrieben.
- `[arguments]` sind die Argumente eines Befehls. Diese werden mit ihren jeweiligen Befehlen aufgelistet und beschrieben.

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

`StyleVisionServer` *unter Windows*
`stylevisionserver` *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

CLI-Befehle

Im Folgenden sind die gültigen CLI-Befehle von StyleVision Server aufgelistet, die in den Unterabschnitten dieses Abschnitts näher beschrieben werden.

- accepteula[39]: (nur Linux) Akzeptiert die Endbenutzer-Lizenzvereinbarung für die aktuelle Installation von StyleVision Server.
- assignlicense[40]: Lädt eine Lizenz in LicenseServer hoch und weist sie StyleVision Server auf diesem Rechner zu.

- exportresourcestrings 42: Exportiert alle Applikationsressourcenstrings in eine XML-Datei.
- generate 44: Generiert ein oder mehrere Dokumente anhand einer XML-Input-Datei und eines XSLT-Stylesheet in der PXF-Input-Datei.
- help 49: Zeigt die Syntax und andere Informationen zu dem im Argument angegebenen Befehl - oder wenn kein Befehl angegeben ist, zu allen Befehlen - an.
- licenseserver 50: Registriert StyleVision Server auf einem LicenseServer im lokalen Netzwerk.
- pdfdata 52: Generiert in einer FDF- oder XML-Datei anhand einer PDF-Datei Formulardaten.
- setdeflang 54: Definiert die Standardsprache von StyleVision Server.
- setfopath 55: Wählt einen alternativen FO-Prozessor für die anschließende PDF-Generierung aus.
- verifylicense 57: Überprüft, ob die aktuelle Version von StyleVision Server lizenziert ist. Überprüft optional, ob sie mit einem bestimmten Lizenzschlüssel lizenziert ist.
- version 59: Zeigt die Versionsnummer von StyleVision Server an.

© 2019-2025 Altova GmbH

4.1 accepteula (nur Linux)

Syntax und Beschreibung

Um StyleVision Server ausführen zu können, muss die Endbenutzer-Lizenzvereinbarung (EULA) akzeptiert werden. Sie können die EULA der Applikation mit dem Befehl **accepteula** akzeptieren.

Dieser Befehl ist z.B. nützlich, wenn Sie StyleVision Server direkt über automatisierte Prozesse unter Verwendung von Skripts lizenzieren und ausführen möchten.

```
stylevisionserver accepteula [options]
```

- Der Befehl funktioniert nur bei Altova Server-Produkten, die auf Linux-Rechnern installiert wurden.
- Sie müssen StyleVision Server zuerst auf dem LicenseServer registrieren, bevor Sie den Befehl **accepteula** ausführen.
- Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.
- Schreiben Sie den Befehl `stylevisionserver` in Kleinbuchstaben.
- Verwenden Sie auf Linux-Systemen Vorwärts-Schrägstriche.

Beispiele

Beispiele für den Befehl **accepteula**:

```
stylevisionserver accepteula
```

Optionen

Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

4.2 assignlicense

Syntax und Beschreibung

Mit dem Befehl **`assignlicense`** wird eine Lizenzdatei auf den Altova LicenseServer, auf dem StyleVision Server registriert ist (siehe Befehl `licenseserver`) hochgeladen und die Lizenz wird StyleVision Server zugewiesen. Der Befehl erhält den Pfad einer Lizenzdatei als Argument. Außerdem können Sie mit dem Befehl die Gültigkeit einer Lizenz überprüfen.

```
stylevisionserver assignlicense [options] FILE
```

- Das Argument *`FILE`* erhält den Pfad der Lizenzdatei.
- Mit der Option `--test-only` wird die Lizenzdatei auf LicenseServer hochgeladen und auf ihre Gültigkeit überprüft. Sie wird jedoch StyleVision Server nicht zugewiesen.

Nähere Informationen zur Lizenzierung finden Sie in der Dokumentation zu LicenseServer (https://www.altova.com/manual/de/licenseserver/3.17/).

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

`StyleVisionServer` *unter Windows*
`stylevisionserver` *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

▼ Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **`"Meine Datei"`**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **`"C:\Mein Verzeichnis\"`**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **`\"`** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **`\\"`**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **`"C:\Mein Verzeichnis\\"`**.

Beispiele

Beispiele für den Befehl `assignlicense`:

```
stylevisionserver assignlicense C:\licensepool\mylicensekey.altova_licenses
stylevisionserver assignlicense --test-only=true C:
\licensepool\mylicensekey.altova_licenses
```

- Mit dem ersten Befehl oben wird die angegebene Lizenzdatei auf LicenseServer hochgeladen und

© 2019-2025 Altova GmbH

StyleVision Server zugewiesen.
- Mit dem letzten Befehl wird die angegebene Lizenz auf LicenseServer hochgeladen und auf ihre Gültigkeit überprüft, ohne sie StyleVision Server zuzuweisen.

Optionen

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **`--option=wert`**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option `TRUE`. Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

▼ test-only [t]

`--t, --test-only =` `true|false`

Die Werte sind `true|false`. Bei `true` wird die Lizenzdatei auf LicenseServer hochgeladen und auf ihre Gültigkeit überprüft, aber nicht zugewiesen.

4.3 exportresourcestrings

Syntax und Beschreibung

Der Befehl **exportresourcestrings** gibt eine XML-Datei aus, die die Ressourcenstrings der StyleVision Server-Applikation in der definierten Sprache enthält. Als Exportsprachen stehen Englisch (en), Deutsch (de), Spanisch (es), Französisch (fr) und Japanisch (ja) zur Verfügung.

```
stylevisionserver exportresourcestrings [options] LanguageCode XMLOutputFile
```

- Das Argument ***LanguageCode*** gibt die Sprache der Ressourcenstrings in der XML-Ausgabedatei an; dies ist die *Exportsprache*. Derzeit unterstützte Exportsprachen sind (mit den Sprachcodes in Klammern): Englisch (en), Deutsch, (de), Spanisch (es), Französisch (fr) und Japanisch (ja).
- Das Argument ***XMLOutputFile*** definiert den Namen und Pfad der XML-Ausgabedatei.

Eine Anleitung zum Erstellen von Lokalisierungen finden Sie weiter unten.

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

StyleVisionServer *unter Windows*
stylevisionserver *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (stylevisionserver) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (StyleVisionServer) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

▼ Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **"Meine Datei"**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **"C:\Mein Verzeichnis\"**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **\"** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **\\"**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **"C:\Mein Verzeichnis\\"**.

Beispiele

Beispiele für den Befehl **exportresourcestrings**:

```
stylevisionserver exportresourcestrings en c:\Strings.xml
```

- Mit dem Befehl oben wird unter c:\ eine Datei namens Strings.xml erstellt, die alle Ressourcenstrings der StyleVision Server Applikation in englischer Sprache enthält.

© 2019-2025 Altova GmbH

Erstellen lokalisierter Versionen von StyleVision Server

Sie können für jede Sprache Ihrer Wahl eine lokalisierte Version von StyleVision Server erstellen. Im Ordner `C:\Programme (x86)\Altova\StyleVisionServer2025\bin` stehen fünf lokalisierte Versionen (Englisch, Deutsch, Spanisch, Französisch und Japanisch) bereits zur Verfügung. Eine Lokalisierung für diese Sprache ist daher nicht mehr notwendig.

Folgendermaßen können Sie eine lokalisierte Version erstellen:

1. Generieren Sie mit Hilfe des Befehls `exportresourcestrings` (*siehe Befehlssyntax oben*) eine XML-Datei, die die Ressourcenstrings enthält. Die Ressourcenstrings in dieser XML-Datei sind in einer der fünf unterstützten Sprachen: je nachdem, welches ***LanguageCode***-Argument mit dem Befehl verwendet wird, in Englisch (`en`), Deutsch (`de`), Spanisch (`es`), Französisch (`fr`) oder Japanisch (`ja`).
2. Übersetzen Sie die Ressourcenstrings aus einer der fünf unterstützten Sprachen in die Zielsprache. Die Ressourcenstrings bilden den Inhalt der `<string>` Elemente in der XML-Datei. Übersetzen Sie keine Variablen in geschweiften Klammern wie z.B. `{option}` oder `{product}`.
3. Wenden Sie sich an den Altova Support, um anhand Ihrer übersetzten XML-Datei eine lokalisierte StyleVision Server DLL-Datei zu generieren.
4. Nachdem Sie Ihre lokalisierte DLL-Datei vom Altova Support erhalten haben, speichern Sie diese unter `C:\Programme (x86)\Altova\StyleVisionServer2025\bin`. Ihre DLL-Datei wird einen Namen in der Form `StyleVisionServer2025_lc.dll` haben. Der `lc` Teil des Namens enthält den Sprachencode. So steht z.B. in `StyleVisionServer2025_de.dll` der Teil `de` für Deutsch.
5. Führen Sie den Befehl `setdeflang` aus, um Ihre lokalisierte DLL als die zu verwendende StyleVision Server Applikation zu definieren. Verwenden Sie den Sprachencode, der Teil des DLL-Namens ist, als Argument des Befehls `setdeflang`.

Anmerkung: Altova StyleVision Server ist mit Unterstützung für fünf Sprachen erhältlich: Englisch, Deutsch, Spanisch, Französisch und Japanisch. Sie müssen daher keine lokalisierte Version dieser Sprachen erstellen. Um eine dieser Sprachen als Standardsprache festzulegen, verwenden Sie den StyleVision Server Befehl `setdeflang`.

4.4 generate

Syntax und Beschreibung

Der Befehl **generate** (Kurzform **gen**) generiert durch Transformieren der XML-Input-Datei mit Hilfe des/der in der PXF-Input-Datei enthaltenen XSLT-Dokuments/Dokumente eine oder mehrere Ausgabedateien (HTML, PDF, RTF und/oder DOCX).

```
stylevisionserver generate | gen --inputxml=Filename [additional options] InputPXF
```

- Die Option **--inputxml** ist obligatorisch und gibt den Pfad der XML-Datei an.
- Das Argument ***InputPXF*** definiert den Pfad der PXF-Datei, die das/die zur Generierung des/der Ausgabedokuments/Ausgabedokumente verwendete(n) XSLT-Dokument(e) enthält. PXF-Dateien wird mit der Altova-Applikation StyleVision generiert.
- Jedes Ausgabeformat wird durch Definition einer Option für diese Ausgabe (*siehe Liste der Optionen unten*) generiert. Der Wert jeder Option ist ein Pfad, der definiert, wo die Ausgabe generiert werden soll.

Anmerkung: StyleVision Server verwendet Apache FOP, den FO-Prozessor des Apache Project zur Generierung von PDF-Dateien anhand von FO. Apache FOP wird zusammen mit StyleVision Server unter dem folgenden Pfad installiert: Auf Windows-Systemen, `ProgramData\Altova\SharedBetweenVersions`; auf Linux- und macOS-Systemen in einem Ordner unterhalb des Ordners `StyleVisionServer2025`. Beachten Sie, dass für Apache FOP **Java Runtime Environment 1.8 oder höher** auf dem StyleVision Server-Rechner installiert sein muss. Installieren Sie für die 32-Bit-Version von StyleVision Server die Java-32-Bit-Version und für die 64-Bit-Version von StyleVision Server die Java-64-Bit-Version. Nähere Informationen zur Konfiguration von FOP finden Sie im Kapitel FOP-Anforderungen 35.

▼ AltovaFOPServer für eine schnellere PDF-Generierung

Bei Ausführung mehrerer Transformationen ins PDF-Format empfiehlt sich vor allem bei großen Dokumenten die Verwendung von AltovaFOPServer, um die Transformation schneller durchführen zu können. Die höhere Geschwindigkeit ist darauf zurückzuführen, dass bei AltovaFOPServer die Java VM- und FOP jar-Dateien im Speicher geladen bleiben, wodurch diese Dateien nicht für jede Transformation erneut geladen werden müssen. AltovaFOPServer wird mit Ihrem StyleVision-Paket installiert. Um AltovaFOPServer für Transformationen verwenden zu können, müssen Sie ihn zuerst über die Befehlszeile starten.

Starten von AltovaFOPServer
Die Befehlssyntax zum Starten von AltovaFOPServer lautet:

```
java -cp "{classPath}" {className} --pid "{pidFile} --port {portNumbers}
```

Ein Beispiel für einen CLI-Befehl zum Starten von AltovaFOPServer:

```
java -cp "C:\Programme\Altova\Common2024\jar\gson\gson.jar;C:
\Programme\Altova\Common2024\jar\AltovaFOPServer.jar"
com.altova.stylevision.fopserver.Main --pid "C:
\Users\user\AppData\Local\Temp\AltovaFopServer.pid" --port 9090-9103,9999
```

Das Argument **classPath** muss die Pfade zu den Dateien **gson.jar** und **AltovaFOPServer.jar** enthalten. Geben Sie die korrekten Pfade dieser Dateien ein.

© 2019-2025 Altova GmbH

Aufrufen von AltovaFOPServer in StyleVision Server
Nachdem der AltovaFOPServer gestartet wurde, können Sie ihn von StyleVision Server mit Hilfe des **generate**-Befehls entweder mit der Option **altova-fopserver-pid-file** oder **altova-fopserver-port** aufrufen (*siehe Options unten*).

- Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

  StyleVisionServer *unter Windows*
  stylevisionserver *unter Windows und Unix (Linux, Mac)*

  * Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
  *Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

- Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

  Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **"Meine Datei"**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **"C:\Mein Verzeichnis\"**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **\"** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **\\"**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **"C:\Mein Verzeichnis\\"**.

Beispiele

Beispiele für den Befehl **generate**:

```
stylevisionserver generate --inputxml=C:\MyFiles\ExpReport.xml --html=Test.html
ExpReport.pxf
stylevisionserver generate --inputxml=C:\ExpReport.pxf|zip\ExpReport.xml --
html=Test.html ExpReport.pxf
stylevisionserver generate --inputxml=altova://packagedfile/ExpReport.xml --
html=Test.html ExpReport.pxf
stylevisionserver generate --inputxml=ExternalXML.xml --html=Test.html Test.pxf
```

- Die obigen Befehle enthalten die obligatorische Option **--inputxml**, das Argument `InputPXF` (`Test.pxf`) sowie mindestens eine Option zur Ausgabeerstellung (`--html` in allen Beispielen oben).
- Die verwendete XML-Input-Datei kann sich innerhalb der PXF-Datei (*siehe erstes und drittes Beispiel oben*) befinden oder eine externe XML-Datei sein (die sich außerhalb der PXF-Datei befindet; *siehe erstes und viertes Beispiel oben*).
- Der `--inputxml` Switch wird ignoriert, wenn die Hauptschemaquelle eine Datenbank oder DB-XML ist, muss aber aus syntaktischen Gründen vorhanden sein und Sie sollten z.B. `--inputxml=database` verwenden.

- Wenn die Option `--html` zur Erstellung des Output wie im Beispiel oben einen relativen Pfad erhält, so wird die Output-Datei unter einem Pfad relativ zum Ordner, in dem sich die PXF-Datei befindet, erstellt.

Optionen

▼ inputxml [xml]

`--xml, --inputxml =` *`PathToXMLFile`*
Diese Option ist obligatorisch. Sie definiert den Pfad zu der zu verarbeitenden XML-Datei. Die XML-Datei kann sich innerhalb oder außerhalb der PXF-Datei befinden. Um als Ziel XML-Dateien innerhalb einer PXF-Datei zu definieren, verwenden Sie den Lokator **`|zip`** (siehe markierter Teil in den Beispielen oben). Die Option `--inputxml` wird ignoriert, wenn es sich bei der Hauptschemaquelle der Input-PXF-Datei um eine Datenbank oder eine DB-XML-Quelldatei handelt.

▼ dbwhere [dbw]

`--dbw, --dbwhere =` *`WHEREClause`*
Eine SQL-`WHERE`-Klausel, die festlegt, welche Zeilen einer DB-XML-Quelldatei verarbeitet werden sollen.

▼ param [p]

`--p, --param =` `$`*`ParamName:ParamValue`*
Weist einem in der PXF-Datei definierten Parameter einen Wert zu. Vor jedem Parameter muss die Option `--param` verwendet werden. Verwenden Sie Anführungszeichen, wenn *`ParamName`* oder *`ParamValue`* ein Leerzeichen enthält. Beispiel: `--p=$company:"Nanonull Inc"`

▼ prohibit-output-outside-target-folder

`--prohibit-output-outside-target-folder =` `true|false`
Die Werte sind `true|false`. Bei Auswahl von `true` können Ausgabedokumente nur in dem Ordner erstellt werden, in dem die Hauptausgabedatei (HTML, PDF, RTF, DOCX, FO) erstellt wird. Dadurch werden andere Ordner, falls nötig, geschützt. Der Standardwert ist `false`.

▼ outhtml [html]

`--html, --outhtml =` *`FilePath`*
Pfad zur zu generierenden HTML-Datei

▼ outpdf [pdf]

`--pdf, --outpdf =` *`FilePath`*
Pfad zur zu generierenden PDF-Datei

▼ outrtf [rtf]

`--rtf, --outrtf =` *`FilePath`*
Pfad zur zu generierenden RTF-Datei

▼ outdocx [docx]

`--docx, --outdocx =` *`FilePath`*
Pfad zur zu generierenden DOCX-Datei

▼ outtext [text]

 © 2019-2025 Altova GmbH

`--text, --outtext =` *`FilePath`*
Pfad zur zu generierenden Textdatei.

- outfo [fo]

  `--fo, --outfo =` *`FilePath`*
  Pfad zur zu generierenden FO-Datei

- generate-html-output-as-mime

  `--generate-html-output-as-mime =` `true|false`
  Die Werte sind `true|false`. Wenn die Option nicht definiert ist, ist der Standardwert `false`. Wenn die Option ohne Wert definiert ist, so ist er `true`. Bei `true` wird die HTML-Ausgabe als Mime-Stream generiert.

- altova-fopserver-pid-file

  `--altova-fopserver-pid-file =` *`PathToFile`*
  Der Pfad zum PID-Datei, die für die Verbindung mit einem laufenden AltovaFOPServer verwendet werden soll. Sie können die Verbindung zu AltovaFOPServer entweder über die PID-Datei oder einen Port des Servers herstellen. Diese PID-Option ist eine Alternative zur Port-Option (*siehe nächste Option*). Nähere Informationen zur Verwendung von AltovaFOPServer finden Sie unter der allgemeinen Beschreibung des Befehls **`generate`** oben.

- altova-fopserver-port

  `--altova-fopserver-port =` *`Value`*
  Eine Port-Nummer oder eine Liste von Port-Nummern, die für die Verbindung zu einem laufenden AltovaFOPServer verwendet werden soll. Die Port-Nummern in einer Liste werden durch Kommas getrennt. Anstelle einer einzigen Nummer kann in der Liste ein Bereich verwendet werden. Daher können Sie als Wert dieser Option **`9090-9100,9999`** angeben. Beachten Sie, dass Sie die Verbindung zu AltovaFOPServer entweder über die PID-Datei des AltovaFOPServers oder einen Port herstellen können. Diese Port-Option ist eine Alternative zur PID-Option (*siehe vorherige Option*). Nähere Informationen zur Verwendung von AltovaFOPServer finden Sie unter der allgemeinen Beschreibung des Befehls **`generate`** oben.

- taxonomy-package

  `--taxonomy-package =` *`FilePath`*
  Pfad zu einem zusätzlichen Taxonomiepaket. Fügen Sie die Option mehrmals hinzu, um mehr als ein Taxonomiepaket zu definieren.

- taxonomy-packages-config-file

  `--taxonomy-packages-config-file =` *`FilePath`*
  Pfad zur **`TaxonomyPackagesConfig.json`**-Datei, einer gemeinsam von den Altova-Produkten XMLSpy, MapForce und StyleVision verwendeten Datei, zum Konfigurieren der XBRL-Taxonomien, die allen diesen Produkten zur Verfügung gestellt werden.

- verbose [v]

  `--v, --verbose =` `true|false`
  Die Werte sind `true|false`. Aktiviert bzw. deaktiviert Anzeige aller Meldungen. Der Standardwert ist `false`, wenn die Option nicht angegeben wird, `true`, wenn die Option ohne einen Wert angegeben wird.

- lang [l]

  --l, --lang = en|de|es|fr|ja
  Die zur Anzeige der Meldung verwendete Sprache.

Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **--option=wert**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option TRUE. Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

Kataloge

Wenn Sie den Altova-Katalog-Mechanismus verwenden, finden Sie die entsprechenden Katalogdateien im Ordner **etc** des StyleVision Server-Applikationsordners. Ausführliche Informationen dazu finden Sie im Abschnitt "Kataloge" des Altova StyleVision-Handbuchs.

CustomCatalog.xml kann anhand der Vorlagendatei **CustomCatalog_template.xml** erstellt werden. Benennen Sie die Vorlagendatei in **CustomCatalog.xml** um, da dies die Datei ist, die im Katalogmechanismus verwendet wird (und nicht die Vorlagendatei).

Beachten Sie dazu Folgendes:

- Bei einer Neuinstallation derselben Hauptversion (selbe oder unterschiedliche Nebenversion) wird die Vorlagendatei durch eine neue Vorlagendatei ersetzt, wobei **CustomCatalog.xml** jedoch unverändert bleibt.
- Wenn Sie jedoch eine neue Hauptversion über eine vorherige Hauptversion installieren, wird der Ordner der vorherigen Hauptversion zusammen mit seiner **CustomCatalog.xml**-Datei gelöscht. Wenn Sie also **CustomCatalog.xml** weiterhin verwenden möchten, müssen Sie **CustomCatalog.xml** aus dem Ordner der vorherigen Hauptversion in einem anderen Ordner sichern. Nach Installation der neuen Hauptversion können Sie die zuvor gespeicherte Datei **CustomCatalog.xml** in den Ordner **etc** der neuen Hauptversion kopieren und dort je nach Bedarf bearbeiten.

© 2019-2025 Altova GmbH

4.5 help

Syntax und Beschreibung

Der Befehl **help** hat ein einziges Argument (`Command`): den Namen des Befehls, zu dem die Hilfe benötigt wird. Er zeigt die korrekte Syntax des Befehls, seine Optionen sowie andere relevante Informationen an. Wenn das Argument `Command` nicht angegeben wird, werden alle Befehle der ausführbaren Datei aufgelistet, wobei zu jedem eine kurze Textbeschreibung angezeigt wird.

```
stylevisionserver help Command
```

- Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

  StyleVisionServer *unter Windows*
  stylevisionserver *unter Windows und Unix (Linux, Mac)*

  * Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
  *Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

Beispiel

Beispiel für den Befehl **help,** um Informationen über den Befehl **licenserver** anzuzeigen:

```
stylevisionserver help licenseserver
```

Die Option --help

Die Hilfe zu einem Befehl kann auch über die Option `--help` im Anschluss an diesen Befehl aufgerufen werden. Mit den beiden unten stehenden Befehlen erhalten Sie dasselbe Ergebnis:

```
stylevisionserver licenseserver --help
```

Im obigen Befehl wird die Option `--help` des Befehls **licenseserver** verwendet.

```
stylevisionserver help licenseserver
```

Der Befehl **help** erhält `licenseserver` als Argument.

In beiden Fällen wird die Hilfe zum Befehl **licenseserver** angezeigt.

4.6 licenseseserver

Syntax und Beschreibung

Bei Ausführung des Befehls **licenseserver** wird StyleVision Server auf dem durch das Argument *Server-Or-IP-Address* definierten LicenseServer registriert. Damit der Befehl `licenseserver` erfolgreich ausgeführt werden kann, müssen sich die beiden Server (StyleVision Server und LicenseServer) im selben Netzwerk befinden und LicenseServer muss ausgeführt werden. Außerdem benötigen Sie zum Registrieren von StyleVision Server auf dem LicenseServer Administratorrechte.

stylevisionserver licenseserver [options] *Server-Or-IP-Address*

- Das Argument *Server-Or-IP-Address* erhält den Namen oder die IP-Adresse des LicenseServer-Rechners.

Sobald StyleVision Server erfolgreich auf dem LicenseServer registriert wurde, erhalten Sie eine entsprechende Meldung. Darin wird auch die URL des LicenseServer angezeigt. Sie können nun zu LicenseServer wechseln und StyleVision Server eine Lizenz zuweisen. Nähere Informationen zur Lizenzierung finden Sie in der Dokumentation zu LicenseServer (https://www.altova.com/manual/de/licenseserver/3.17/).

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

StyleVisionServer *unter Windows*
stylevisionserver *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

▼ Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **"Meine Datei"**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **"C:\Mein Verzeichnis\"**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **\"** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **\\"**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **"C:\Mein Verzeichnis\\"**.

Beispiele

Beispiele für den Befehl **licenseserver**:

```
stylevisionserver licenseserver DOC.altova.com
stylevisionserver licenseserver localhost
stylevisionserver licenseserver 127.0.0.1
```


© 2019-2025 Altova GmbH

Die Befehle oben definieren den Rechner namens `DOC.altova.com` und den Rechner des Benutzers (`localhost` bzw. `127.0.0.1`) als den Rechner, auf dem Altova LicenseServer ausgeführt wird. In jedem dieser Fälle wird StyleVision Server auf dem LicenseServer auf dem angegebenen Rechner registriert. Mit dem letzten Befehl wird zum Ausführen des Befehls die ausführbare Server-Datei aufgerufen.

Optionen

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **`--option=wert`**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option `TRUE`. Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

▼ json [j]

`--j, --json =` `true|false`

Die Werte sind `true|false`. Bei `true` wird das Ergebnis des Registrierungsversuchs als JSON-Objekt ausgegeben, das von Rechnern geparst werden kann.

4.7 pdfdata

Syntax und Beschreibung

Der Befehl **pdfdata** generiert anhand der als ***InputPDF-***Argument bereitgestellten PDF-Datei eine FDF- oder XML-Datei.

```
stylevisionserver pdfdata [options] InputPDF
```

- Das Argument ***InputPDF*** definiert den Pfad zu der PDF-Datei, anhand welcher die FDF- oder XML-Ausgabedatei generiert wird. Wenn die PDF-Datei keine Formulardaten enthält, so enthält auch die generierte XML-Datei keine Formulardaten.
- Mit Hilfe der Option **--outfdf** können Sie den Pfad der generierten FDF-Datei bzw. mit der Option **--outxml** den Pfad der generierten XML-Datei definieren

Nähere Informationen zur Konfiguration von FOP, mit Hilfe dessen StyleVision Server standardmäßig PDF-Dateien generiert, finden Sie im Kapitel FOP-Anforderungen 35.

Nähere Informationen zu FDF-Dateien und der Erstellung von ausfüllbaren PDF-Formularen finden Sie in der Dokumentation zu Altova StyleVision (Enterprise Edition).

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

StyleVisionServer *unter Windows*
stylevisionserver *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

▼ Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **"Meine Datei"**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **"C:\Mein Verzeichnis\"**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **\"** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **\\"**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **"C:\Mein Verzeichnis\\"**.

Beispiele

Beispiele für den Befehl **pdfdata**:

```
stylevisionserver pdfdata --outfdf=C:\test\forms\FDFData.fdf C:\test\forms\TestForm.pdf
stylevisionserver pdfdata --outxml=C:\test\forms\XMLData.xml C:\test\forms\TestForm.pdf
```

 © 2019-2025 Altova GmbH

Mit den obigen Beispielen wird anhand derselben PDF-Input-Datei eine FDF-Datei bzw. eine XML-Datei erstellt.

Optionen

- outfdf

  `--outfdf =` *`FilePath`*
  Der Pfad zur generierten FDF-Datei.

- outxml

  `--outxml =` *`FilePath`*
  Der Pfad zur generierten XML-Datei.

- verbose [v]

  `--v, --verbose =` `true|false`
  Die Werte sind `true|false`. Aktiviert bzw. deaktiviert Anzeige aller Meldungen. Der Standardwert ist `false`, wenn die Option nicht angegeben wird, `true`, wenn die Option ohne einen Wert angegeben wird.

Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **`--option=wert`**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option `TRUE`. Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

4.8 setdeflang

Syntax und Beschreibung

Der Befehl **setdeflang** (Kurzform ist **sdl**) definiert die Standardsprache von StyleVision Server. Verfügbare Sprachen sind Englisch (en), Deutsch (de), Spanisch (es), Französisch (fr) und Japanisch (ja). Der Befehl erhält ein obligatorisches Argument *LanguageCode*.

```
stylevisionserver setdeflang [options] LanguageCode
```

- Das Argument *LanguageCode* definiert die Standardsprache von StyleVision Server. Die entsprechenden Werte sind: en, de, es, fr, ja.
- Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

StyleVisionServer *unter Windows*
stylevisionserver *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (stylevisionserver) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (StyleVisionServer) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

Beispiele

Beispiele für den Befehl **setdeflang** (**sdl**):

```
stylevisionserver sdl en
stylevisionserver setdeflang es
```

- Mit dem ersten Befehl wird als Standardsprache von StyleVision Server Englisch definiert.
- Mit dem zweiten Befehl wird als Standardsprache von StyleVision Server Spanisch definiert.

Optionen

Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

© 2019-2025 Altova GmbH

4.9 setfopath

Syntax und Beschreibung

Der Befehl **setfopath** (Kurzform ist **sfp**) definiert den Pfad zu einem anderen Apache-FOP-Prozessor als dem im StyleVision Server-Paket inkludierten.

stylevisionserver setfopath | sfp [options] ***Path***

- Standardmäßig wird zum Verarbeiten von FO-Dokumenten und Generieren von PDF-Ausgabedokumenten der in StyleVision Server inkludierte FOP-Prozessor verwendet. Wenn eine andere Apache FOP-Prozessorinstanz als die mit StyleVision Server installierte verwendet werden soll, verwenden Sie den Befehl `setfopath`, wobei mit dem Argument ***Path*** der Pfad zum gewünschten FO-Prozessor angegeben wird.
- Nachdem mit dem Befehl `setfopath` ein anderer FO-Prozessor definiert wurde, wird bei Verwendung nachfolgender **`generate`** [44] Befehle dieser Prozessor für die PDF-Generierung verwendet. Um den Prozessor wieder zu wechseln, verwenden Sie wieder den Befehl `setfopath`. Um zurück zum FOP-Prozessor von StyleVision Server zu wechseln, navigieren Sie zum FOP-Ordner auf Ihrem System und verwenden Sie diesen Pfad als Argument von `setfopath`.
- Auf Windows-Systemen befindet sich der mit StyleVision Server installierte Ordner FOP unter `ProgramData\Altova\SharedBetweenVersions`; auf Linux- und macOS-Systemen befindet er sich in einem untergeordnete Ordner des Ordners `StyleVisionServer2025`.

Nähere Informationen zur Konfiguration von FOP, mit Hilfe dessen StyleVision Server standardmäßig PDF-Dateien generiert, finden Sie im Kapitel FOP-Anforderungen [35].

Nähere Informationen zu FDF-Dateien und der Erstellung von ausfüllbaren PDF-Formularen finden Sie in der Dokumentation zu Altova StyleVision (Enterprise Edition).

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

`StyleVisionServer` *unter Windows*
`stylevisionserver` *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

▼ Umgekehrte Schrägstriche, Leerzeichen und Sonderzeichen auf Windows-Systemen

Auf Windows-Systemen: Verwenden Sie bei Vorhandensein von Leerzeichen oder Sonerzeichen in Strings (z.B. in Datei- oder Ordnernamen oder Firmen-, Personen- oder Produktnamen) Anführungszeichen: z.B., **`"Meine Datei"`**. Beachten Sie jedoch, dass ein von einem doppelten Anführungszeichen gefolgter umgekehrter Schrägstrich (z.B: **`"C:\Mein Verzeichnis\"`**) eventuell nicht korrekt gelesen wird, da der umgekehrte Schrägstrich auch den Beginn einer Escape-Sequenz markiert und die Escape-Sequenz **`\"`** für ein doppeltes Anführungszeichen steht. Wenn Sie diese Zeichensequenz mit einem Escape versehen wollen, verwenden Sie einen vorangestellten umgekehrten Schrägstrich, wie den folgenden: **`\\"`**. Einfacher ausgedrückt: Wenn Sie einen Dateipfad, der Leerzeichen oder einen umgekehrten Schrägstrich am Ende enthält, schreiben müssen, so schreiben Sie diesen folgendermaßen: **`"C:\Mein Verzeichnis\\"`**.

Beispiele

Nach Ausführung des Befehls `setfopath` können Sie mit Hilfe des Befehls **`generate`** [44] und unter Verwendung des soeben definierten FO-Prozessors eine PDF-Datei generieren.

```
StyleVisionServer setfopath C:\FOP\FOP.bat
StyleVisionServer generate --inputxml=Test.xml --pdf=Test.pdf Test.pxf
```

Die oben aufgelisteten Befehle führen folgende Schritte aus:

1. Der Befehl `setfopath` definiert, dass der FOP-Prozessor unter dem Pfad C:\FOP\FOP.bat verwendet werden soll, um die PDF-Datei zu generieren, die mit dem nachfolgenden Befehl generiert wird.
2. Der Befehl `generate` generiert anhand der definierten XML-Input-Datei eine PDF-Datei und verwendet in der PXF-Datei enthaltene Transformationsdateien. Zum Generieren der PDF-Datei wird der im vorherigen Befehl definierte FOP-Prozessor verwendet.

Optionen

Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **`--option=wert`**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option `TRUE`. Mit Hilfe der Option **`--h, --help`** können Sie Informationen über den Befehl anzeigen.

© 2019-2025 Altova GmbH

4.10 verifylicense

Syntax und Beschreibung

Mit dem Befehl **verifylicense** wird überprüft, ob das aktuelle Produkt lizenziert ist. Zusätzlich können Sie mit der Option `--license-key` überprüfen, ob dem Produkt bereits ein bestimmter Lizenzschlüssel zugewiesen wurde.

```
stylevisionserver verifylicense [options]
```

- Um zu überprüfen, ob StyleVision Server eine bestimmte Lizenz zugewiesen ist, geben Sie den Lizenzschlüssel als Wert der Option `--license-key` an.

Nähere Informationen zur Lizenzierung finden Sie in der Dokumentation zu LicenseServer (https://www.altova.com/manual/de/licenseserver/3.17/).

▼ Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

StyleVisionServer *unter Windows*
stylevisionserver *unter Windows und Unix (Linux, Mac)*

* Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
*Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

Beispiele

Beispiele für den Befehl **verifylicense**:

```
stylevisionserver verifylicense
stylevisionserver verifylicense --license-key=ABCD123-ABCD123-ABCD123-ABCD123-ABCD123-ABCD123
```

- Mit dem ersten Befehl wird überprüft, ob StyleVision Server lizenziert ist.
- Mit dem zweiten Befehl wird überprüft, ob StyleVision Server mit dem in der Option `--license-key` definierten Lizenzschlüssel lizenziert ist.

Optionen

Optionen werden in ihrer kurzen Form (falls verfügbar) und in ihrer langen Form aufgelistet. Für die kurze und die lange Form können ein oder zwei Bindestriche verwendet werden. Eine Option kann, muss aber keinen Wert erhalten. Eine Option, die einen Wert erhält, wird folgendermaßen geschrieben: **--option=wert**. Werte können außer in zwei Fällen ohne Anführungszeichen definiert werden: (i) wenn der Wertestring Leerzeichen enthält oder (ii) wenn in der Beschreibung der Option explizit erwähnt ist, dass Anführungszeichen zwingend erforderlich sind. Wenn eine Option einen Booleschen Wert erhält und kein Wert definiert ist, so ist der Standardwert der Option `TRUE`. Mit Hilfe der Option **--h, --help** können Sie Informationen über den Befehl anzeigen.

▼ license-key [l]

--l, --license-key = *Value*

Überprüft, ob StyleVision Server mit dem als Wert dieser Option definierten Lizenzschlüssel lizenziert ist.

© 2019-2025 Altova GmbH

4.11 version

Syntax und Beschreibung

Mit dem Befehl **version** wird die Versionsnummer von StyleVision Server angezeigt.

```
stylevisionserver version
```

- Groß- und Kleinschreibung und Schrägstriche in der Befehlszeile

  StyleVisionServer *unter Windows*
  stylevisionserver *unter Windows und Unix (Linux, Mac)*

  * Beachten Sie, dass klein geschriebene Befehle (`stylevisionserver`) auf allen Plattformen (Windows, Linux und Mac) funktionieren, während großgeschriebene Befehle (`StyleVisionServer`) nur unter Windows und Mac ausgeführt werden.
  *Verwenden Sie auf Linux und Mac-Systemen Schrägstriche und auf Windows-Systemen umgekehrte Schrägstriche.

Beispiel

Beispiel für den Befehl `version`:

```
stylevisionserver version
```

5 StyleVision Server API

StyleVision Server enthält eine API (Application Programming Interface), die Sie von Ihrem .NET-, COM- oder Java-basierten Code aus programmatisch aufrufen können.

Dieser Abschnitt ist in die folgenden Unterabschnitte gegliedert:

- Informationen zur .NET-Schnittstelle 61
- Informationen zur COM-Schnittstelle 62
- Informationen zur Java-Schnittstelle 63
- Codebeispiele 65
- API-Referenz 74

© 2019-2025 Altova GmbH

5.1 Informationen zur .NET-Schnittstelle

Die .NET-Schnittstelle bildet einen Wrapper rund um die COM-Schnittstelle. Sie dient als primäre, von Altova signierte Interop Assembly und verwendet den Namespace `Altova.StyleVisionServer`.

StyleVision Server wird bei der Installation automatisch als COM-Serverobjekt registriert und muss daher nicht manuell registriert werden. Wenn Sie einen Zugriffsfehler erhalten, öffnen Sie die Komponentendienste und räumen Sie dem Konto, auf dem der Applikationspool, der StyleVision Server enthält, ausgeführt wird, Zugriffsrechte ein.

Um StyleVision Server in Ihrem .NET-Projekt verwenden zu können, fügen Sie eine Referenz zur `Altova.StyleVisionServer.dll` Datei hinzu (siehe Anleitung unten). Die Datei `Altova.StyleVisionServer.dll` befindet sich im Ordner `bin` des StyleVision Server Installationsordners. Diese `.dll`-Datei wird bei der Installation von StyleVision Server automatisch zum GAC (Global Assembly Cache (GAC) hinzugefügt (der GAC befindet sich normalerweise im Ordner `C:\WINDOWS\assembly`).

Sobald StyleVision Server als COM-Serverobjekt registriert wurde und die Datei `Altova.StyleVisionServer.dll` der .NET-Schnittstelle zur Verfügung steht, stehen die StyleVision Server AP-Funktionen in Ihrem .NET-Projekt zur Verfügung.

Anmerkung: Wenn Sie eine 64-Bit-Version von StyleVision Server installiert haben, so befinden sich die Dateien der 32-Bit-Version von `Altova.StyleVisionServer.dll` im Ordner `bin\API_32bit`. Ebenso, wenn Sie eine 32-Bit-Version von StyleVision Server installiert haben, so befinden sich die Dateien der 64-Bit-Version von `Altova.StyleVisionServer.dll` im Ordner `bin\API_64bit`.

So fügen Sie in einem Visual Studio .NET-Projekt eine Referenz zur StyleVision Server-DLL hinzu

1. Klicken Sie, während das .NET-Projekt in Visual Studio geöffnet ist, auf **Project | Add Reference**. Daraufhin wird das Dialogfeld "Add Reference" geöffnet.
2. Suchen Sie auf dem Register "Browse" nach dem Ordner *<StyleVisionServer application folder>*`/bin`, wählen Sie `Altova.StyleVisionServer.dll` aus und klicken Sie auf **OK**.

Sie können die Struktur der `Altova.StyleVisionServer` Assembly mit dem Visual Studio Object Browser anzeigen (um den Object Browser aufzurufen, klicken Sie im Menü **View** auf **Object Browser**).

5.2 Informationen zur COM-Schnittstelle

StyleVision Server wird bei der Installation automatisch als COM-Serverobjekt registriert. Um zu überprüfen, ob die Registrierung erfolgreich war, öffnen Sie den Registry Editor (z.B. durch Eingabe des Befehls `regedit.exe` in der Befehlszeile). Wenn die Registrierung erfolgreich war, enthält die Registry die Klasse `StyleVision.Server`. Diese Klasse befindet sich normalerweise unter `HKEY_LOCAL_MACHINE\SOFTWARE\Classes`.

Sobald das COM-Serverobjekt registriert ist, können Sie es über Applikationen und Programmiersprachen mit Programmierunterstützung für COM-Aufrufe aufrufen. Wenn Sie den Pfad des StyleVision Server Installationspakets ändern möchten, sollten Sie StyleVision Server am besten deinstallieren und anschließend im gewünschten Ordner installieren. Auf diese Art wird die nötige Deregistrierung und Registrierung bei der Installation durchgeführt.

© 2019-2025 Altova GmbH

5.3 Informationen zur Java-Schnittstelle

Die API besteht aus einer JAR-Datei (**StyleVisionServer.jar**) und einer JNI-Datei (**StyleVisionServer.dll**). Beide Dateien sowie andere damit in Zusammenhang stehende API-Dateien befinden sich im Ordner `bin` des StyleVision Server Installationsordners. Sie können diese Dateien entweder über ihren Originalordner referenzieren oder sie in einen anderen Ordner kopieren, falls Ihr Projekt dies erfordert. (Auf Windows-Systemen benötigen Sie Administratorrechte, um das Programm von seinem Originalordner aus zu starten).

Anmerkung: Wenn Sie eine 64-Bit-Version von StyleVision Server installiert haben, so befinden sich die Dateien der 32-Bit-Version von **StyleVisionServer.jar** und **StyleVisionServer.dll** im Ordner `bin\API_32bit` des StyleVision Server Installationsverzeichnisses. Sie benötigen diese Dateien, wenn Sie eine 32-Bit-Java-Version verwenden. Ebenso, wenn Sie eine 32-Bit-Version von StyleVision Server installiert haben, so befinden sich die Dateien der 64-Bit-Version von **StyleVisionServer.jar** und **StyleVisionServer.dll** im Ordner `bin\API_64bit`. Sie benötigen diese Dateien, wenn Sie eine 64-Bit-Java-Version verwenden.

Um die StyleVision Server API über Java-Code aufrufen zu können, fügen Sie die folgenden Referenzen zur `.classpath`-Datei Ihres Java-Projekts hinzu.

`StyleVisionServer.jar`	Die Bibliothek, die mit StyleVision Server kommuniziert
`StyleVisionServer_JavaDoc.zip`	Die Dokumentation zur StyleVision Server API

Des Weiteren muss der **java.library.path** den Ordner enthalten, in dem sich die JNI-Bibliotheksdatei (**StyleVisionServer.dll**) befindet.

Wenn Sie Ihr Projekt auf einem Applikationsserver bereitstellen, stellen Sie sicher, dass **StyleVisionServer.jar** und **StyleVisionServer.dll** auf dem Server korrekt mit Java konfiguriert sind.

Ein Beispiel dafür, wie die API-Bibliotheksdateien verwendet werden, sehen Sie in der Beispiel-Batch-Datei **buildAndRun.bat** (*siehe unten*), im Ordner **etc\Examples\Java** Ihres StyleVision Server-Installationsverzeichnisses.

Erstellen und Ausführen eines Java-Programms über die API

Ein Beispiel dafür, wie Sie ein Java-Programm, das die StyleVision Server-API verwendet, erstellen und ausführen können, finden Sie in der Beispiel-Batch-Datei **buildAndRun.bat**. Sie können diese Datei für Ihre eigenen Java-Programme anpassen und wiederverwenden.

Starten Sie die Batch-Datei mit dem folgenden Befehl über eine Befehlszeilenschnittstelle:

```
buildAndRun "path_to_Java_bin_folder"
```

Anmerkung: Um zu überprüfen, ob sich Java in Ihrem Classpath befindet, führen Sie den Befehl **java --version** aus. Wenn sich Java nicht in Ihrem Classpath befindet, müssen Sie den Pfad als Parameter des **buildAndRun**-Befehls angeben. Wenn der Pfad Leerzeichen enthält, setzen Sie ihn in Anführungszeichen.

Codefragment von buildAndRun.bat

```
@echo off
```

```
if %1.==. goto error

REM The location of the JAVA API binaries, the JAR file and the JNI library.
REM Adapt to your needs.
SETLOCAL
Set JavaAPIBinPath=%PROGRAMFILES%\Altova\StyleVisionServer2025\bin

REM Compile sample java
REM The -cp option (classpath) needs to point to the installed jar file (here, in its
original location)
REM "Program.java" is the Java program you want to compile
%1\javac.exe -cp "%JavaAPIBinPath%\StyleVisionServer.jar" Program.java

REM Run sample java
REM The -cp option (classpath) needs to point to the StyleVisionServer.jar file
REM The java.library.path needs to include the folder where the JNI library
StyleVisionServer.dll is located.
%1\java.exe -cp "%JavaAPIBinPath%\StyleVisionServer.jar;." -Djava.library.path="%
JavaAPIBinPath%" Program

@echo off
goto end

:error
echo Usage: buildAndRun "<path_to_java_bin_folder>"

:end
```

Hinzufügen der Bibliotheksreferenzen in Eclipse

Sie können die classpath-Referenzen in Eclipse durch Bearbeiten der Eigenschaften des Java-Projekts hinzufügen. Die Beispielanleitung unten gilt für Eclipse 4.4.

1. Klicken Sie, während das Projekt in Eclipse geöffnet ist, auf das Menü **Project**, anschließend auf **Properties** und wählen Sie anschließend den Java Build Path aus.
2. Klicken Sie auf dem Register "Libraries" auf **Add External JARs** und navigieren Sie zur Datei `StyleVisionServer.jar` im StyleVision Server Installationsordner.
3. Erweitern Sie unter *JARs and class folders on the build path* den Eintrag `StyleVisionServer.jar` und doppelklicken Sie anschließend auf den Eintrag `Javadoc location: (None)`.
4. Stellen Sie sicher, dass die Optionen *Javadoc in archive* und *External file* aktiviert sind und navigieren Sie anschließend zur Datei `StyleVisionServer_JavaDoc.zip` im StyleVision Server Installationsordner.
5. Klicken Sie auf **OK**. Die Referenz zur StyleVision Server Bibliothek und zum Javadoc-Archiv wird zur `.classpath`-Datei des Projekts hinzugefügt.

© 2019-2025 Altova GmbH

5.4 Codebeispiele

Dieser Abschnitt enthält Beispiele für die folgenden Programmiersprachen:

- C# 65
- C++ 66
- Java 68
- VBScript 69
- Visual Basic 71

Nachdem Sie StyleVision Server installiert haben, kopieren Sie den Ordner **etc/Examples** in Ihr Startverzeichnis oder einen anderen geeigneten Ordner.

5.4.1 C#

Im Beispiel unten wird gezeigt, wie Sie über C#-Code anhand einer PXF-Datei und einer XML-Input-Datei eine RTF-Output-Datei generieren. Stellen Sie sicher, dass StyleVision Server installiert und lizenziert ist und als COM-Servereobjekt zur Verfügung steht. Die Registrierung als COM-Serverobjekt erfolgt normalerweise während der Installation von StyleVision Server. Eine Anleitung, wie Sie überprüfen, ob die Registrierung erfolgreich war, finden Sie unter Informationen zur COM-Schnittstelle 62. Siehe auch Informationen zur .NET-Schnittstelle 61.

```
namespace StyleVisionServerAPI_sample
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
               //Create a StyleVision Server object
               Altova.StyleVisionServer.Server objSVS = new
Altova.StyleVisionServer.Server();

               //Set a working directory - used for output and for intermediate files
               objSVS.WorkingDirectory = "..\\..\\..";

               //Default path to the StyleVision Server executable is the installation path
(same dir with the StyleVisionServer.dll)
               //In case you moved the binaries on the disk, you need to explicitly set the
path to the .exe file
               //The ServerPath should be set before any other properties because it re-
initializes the server and resets all properties.
               //objSVS.ServerPath = "C:\\Program Files (x86)\\Altova\
\StyleVisionServer2025\\bin\\StyleVisionServer.exe";

               //Prepare the name of the working XML
```

```
            //    This can be an absolute/relative path if the file is stored externally
(not inside PXF)
            // objSVS.InputXML = "ExpReport.xml";
            //    Or it can contain the path INSIDE the PXF
            // objSVS.InputXML = "ExpReport.pxf|zip\\ExpReport.xml";
            //    Easiest way is to refer to the file as being embedded in the
transformation file
            objSVS.InputXML = "altova://packagedfile/ExpReport.xml";

            //Add output paths (absolute or relative to WorkingDirectory) for all
formats that should be generated
            objSVS.OutputRTF = "C:\\tmp\\ExpReport.rtf";

            //Prepare the parameters, if your design uses parameters
            //objSVS.AddParameter( "testparam1", "value 1" );

            //Run the transformation; the output will be stored at C:\temp\ExpReport.rtf
            // NOTE Please adapt the path to the input file in order to run the sample
            if (objSVS.Generate("ExpReport.pxf"))
                System.Console.WriteLine("Success - finished execution");
            else
                System.Console.WriteLine(objSVS.LastExecutionMessage);
        }
        catch (System.Runtime.InteropServices.COMException ex)
        {
            // some general error like an invalid license happened
            System.Console.WriteLine("Internal Error - " + ex.Message);
        }
    }
  }
}
```

5.4.2 C++

Im Beispiel unten wird gezeigt, wie Sie über C++-Code anhand einer PXF-Datei und einer XML-Input-Datei eine RTF-Output-Datei generieren. Stellen Sie sicher, dass StyleVision Server installiert und lizenziert ist und als COM-Servereobjekt zur Verfügung steht. Die Registrierung als COM-Serverobjekt erfolgt normalerweise während der Installation von StyleVision Server. Eine Anleitung, wie Sie überprüfen, ob die Registrierung erfolgreich war, finden Sie unter Informationen zur COM-Schnittstelle 62. Siehe auch Informationen zur .NET-Schnittstelle 61.

```
// StyleVisionServerAPI_Sample.cpp : Defines the entry point for the console application.
//
#include <iostream>
#include "atlbase.h"


// The following import statements require the corresponding C++ tool-chain to be selected
in the project configuration file.
#ifndef _WIN64
```

 © 2019-2025 Altova GmbH

```
// 32-bit StyleVisionServer
#import "progid:StyleVision.Server"
#else
// 64-bit StyleVisionServer
#import "progid:StyleVision_x64.Server"
#endif


int _tmain(int argc, _TCHAR* argv[])
{
   CoInitialize( NULL );

   try
   {
      //Create a StyleVision Server object
      StyleVisionServerLib::IServerPtr pSVS;
      CoCreateInstance( __uuidof( StyleVisionServerLib::Server ), NULL, CLSCTX_ALL,
__uuidof( StyleVisionServerLib::IServer ), reinterpret_cast< void** >( &pSVS ) );


      //Set a working directory - used for output and for intermediate files
      pSVS->WorkingDirectory = ".."; // this is relative to this applications' working
directory (the project folder)

      //Default path to the StyleVision Server executable is the installation path (same dir
with the StyleVisionServer.dll)
      //In case you moved the binaries on the disk, you need to explicitly set the path to
the .exe file
      //The ServerPath should be set before any other properties because it re-initializes
the server and resets all properties.
      //pSVS->ServerPath = "C:\\Program Files (x86)\\Altova\\StyleVisionServer2025\\bin\
\StyleVisionServer.exe";
      //pSVS->ServerPath = "C:\\Program Files\\Altova\\StyleVisionServer2025\\bin\
\StyleVisionServer.exe";

      //Prepare the name of the working XML
      //    This can be an absolute/relative path if the file is stored externally (not
inside PXF)
      // pSVS->InputXML = "ExpReport.xml";
      //    Or it can contain the path INSIDE the PXF
      // pSVS->InputXML = "ExpReport.pxf|zip\\ExpReport.xml";
      //    Easiest way is to refer to the file as being embedded in the transformation file
      pSVS->InputXML = "altova://packagedfile/ExpReport.xml";

      //Add output paths (absolute or relative to WorkingDirectory) for all formats that
should be generated
      pSVS->OutputRTF = "ExpReport.rtf";
      pSVS->OutputPDF = "ExpReport.pdfrtf";
      pSVS->OutputHTML = "ExpReport.html";

      //Prepare the parameters, if your design uses parameters
      //pSVS->AddParameter( "testparam1", "value 1" );
```

```
      //Run the transformation; the output will be stored at C:\temp\ExpReport.rtf
      // NOTE Please adapt the path to the input file in order to run the sample
      if (pSVS->Generate("ExpReport.pxf"))
               {
         std::cout << pSVS->LastExecutionMessage << std::endl;
                     std::cout << "Success - finished execution" << std::endl;
               }
      else
         std::cout << pSVS->LastExecutionMessage << std::endl;
   }
   catch (_com_error& err )
   {
      BSTR  bstrMessage;
      (err).ErrorInfo()->GetDescription( &bstrMessage );
      std::cout << "Exception occurred: " << _com_util::ConvertBSTRToString( bstrMessage )
<< std::endl;
   }

   CoUninitialize();
   return 0;
}
```

5.4.3 Java

Im Beispiel unten wird gezeigt, wie Sie über Java-Code anhand einer PXF-Datei und einer XML-Input-Datei eine RTF-Output-Datei generieren. Stellen Sie sicher, dass StyleVision Server installiert und lizenziert ist und als Server-Objekt zur Verfügung steht. Die Registrierung als Serverobjekt erfolgt normalerweise während der Installation von StyleVision Server. Das unten stehende Beispielprogramm kann mit Hilfe der Batch-Datei **buildAndRun.bat** aus dem Ordner **etc\Examples\Java** Ihres StyleVision Server-Installationsverzeichnisses erstellt und ausgeführt werden.

Informationen zur Schnittstelle finden Sie unter Informationen zur Java-Schnittstelle 63.

Anmerkung: Es wird empfohlen, den Ordner "Examples" in Ihr Startverzeichnis oder einen andere geeignete Ordner zu kopieren und anschließend zum Java-Ordner zu navigieren, um **Program.java** und **buildAndRun.bat** aufzurufen.

Program.java

```
public class Program
{

   public static void main(String[] args)
   {
      com.altova.stylevisionserver.StyleVisionServer objSVS
      try
      {
          //Create a StyleVision Server object
          objSVS = new com.altova.stylevisionserver.StyleVisionServer();
```

 © 2019-2025 Altova GmbH

```
          //The default location of server binary is the folder containing the Java native
library
          //Select a different server binary with the following line:
          //obj.SVS.setServerPath(strServerPath);
          //The ServerPath should be set before any other properties because it re-
initializes the server and resets all properties.

          //The sample data is located in the parent folder of the Java sample code
          //Set this parent folder to be the working directory:
          objSVS.setWorkingDirectory( ".." );

          System.out.println("Running " + objSVS.getProductNameAndVersion());

          //Prepare the name of the working XML
          //This can be an absolute/relative path if the file is external (not inside PXF)
          //   objSVS.setInputXML( "ExpReport.xml" );
          //Or it can contain the path INSIDE the PXF
          //   objSVS.setInputXML( "C:\\Program Files (x86)\\Altova\\StyleVisionServer" +
majorVersionYear + "\\etc\\Examples\\ExpReport.pxf|zip\\ExpReport.xml" );
          //The easiest way is to refer to the file as being embedded in the transformation
file
          objSVS.setInputXML( "altova://packagedfile/ExpReport.xml" );

          //Add output paths (absolute or relative to WorkingDirectory) for all formats
that should be generated
          objSVS.setOutputRTF( "ExpReport.rtf" );

          //Prepare the parameters, if your design uses parameters
          //objSVS.AddParameter( "testparam1", "value 1" );

          //Run the transformation; the output will be stored at C:\temp\ExpReport.rtf
          // NOTE Please adapt the path to the input file in order to run the sample
          if ( objSVS.generate( "ExpReport.pxf" ) )
               System.out.println( "Success"  +  objSVS.getLastExecutionMessage() );
          else
               System.out.println( objSVS.getLastExecutionMessage() );
       }
       catch ( Exception e)
       {
           System.out.println(e.getMessage());
       }
   }

}
```

5.4.4 VBScript

Im Beispiel unten wird gezeigt, wie Sie über VB Script-Code anhand einer PXF-Datei und einer XML-Input-Datei eine RTF-Output-Datei generieren. Stellen Sie sicher, dass StyleVision Server installiert und lizenziert ist und als COM-Servereobjekt zur Verfügung steht. Die Registrierung als COM-Serverobjekt erfolgt normalerweise

während der Installation von StyleVision Server. Eine Anleitung, wie Sie überprüfen, ob die Registrierung erfolgreich war, finden Sie unter Informationen zur COM-Schnittstelle 62.

```
Option Explicit

'Create a StyleVision Server object; use "StyleVision_x64.Server" if you want to use the
64-bit installation
Dim objSVS
' Since we load a COM-DLL we need care about the process architecture
On Error Resume Next ' ignore any COM errors avoiding uncontrolled script termination
Dim WshShell
Dim WshProcEnv
Set WshShell =  CreateObject("WScript.Shell")
Set WshProcEnv = WshShell.Environment("Process")
Dim process_architecture
process_architecture= WshProcEnv("PROCESSOR_ARCHITECTURE")
If process_architecture = "x86" Then
   Set objSVS = WScript.GetObject( "", "StyleVision.Server" )
   If Err.Number <> 0 then
      WScript.Echo("You are running in a 32-bit process but StyleVision Server COM-API 32-
bit seems not to be installed on your system.")
      WScript.Quit -1
   End If
Else
   Set objSVS = WScript.GetObject( "", "StyleVision_x64.Server" )
   If Err.Number <> 0 then
      WScript.Echo("You are running in a 64-bit process but StyleVision Server COM-API 64-
bit seems not to be installed on your system.")
      WScript.Echo("If you have installed 32-bit StyleVision Server consider calling your
script from the 32-bit console 'C:\Windows\SysWOW64\cmd.exe.'")
      WScript.Quit -1
   End If
End If
On Error Goto 0    ' re-enble default error promotion

'Set a working directory - used for input, output and for intermediate files
'objSVS.WorkingDirectory = "C:\Program Files (x86)
\Altova\StyleVisionServer2020\etc\examples"
objSVS.WorkingDirectory = ".."

'Default path to the StyleVision Server executable is the installation path (same dir with
the StyleVisionServer.dll)
'In case you moved the binaries on the disk, you need to explicitly set the path to the
.exe file
'The ServerPath should be set before any other properties because it re-initializes the
server and resets all properties.
'objSVS.ServerPath = "C:\Program Files (x86)
\Altova\StyleVisionServer2020\bin\StyleVisionServer_DebugDLL.exe"

' The Generate method will return 'True' if generation was successful otherwise 'False'.
' In the case of fundamental errors like termination of the server process a COM error will
be raised which
```

 © 2019-2025 Altova GmbH

```
' can be handled using the VBScript Err object.
On Error Resume Next ' ignore any COM errors avoiding uncontrolled script termination
Err.Clear

WScript.Echo("Running " & objSVS.ProductNameAndVersion & vbCrlf)

'Prepare the name of the working XML
'    This can be an absolute/relative path if the file is stored externally (not inside
PXF)
' objSVS.InputXML = "ExpReport.xml"
'    or it can contain the path INSIDE the PXF
objSVS.InputXML = "ExpReport.pxf|zip\ExpReport.xml"
'    or refer to the file as being embedded in the transformation file
'objSVS.InputXML = "altova://packagedfile/ExpReport.xml"

'Add output paths (absolute or relative to WorkingDirectory) for all formats that should be
generated
' make sure you have write permissions
'objSVS.OutputRTF = "C:\tmp\ExpReport.rtf"
objSVS.OutputPDF = "C:\tmp\ExpReport.pdf"
'objSVS.OutputHTML = "C:\tmp\ExpReport.html"

'Prepare the parameters, if your design uses parameters
'Call objSVS.AddParameter( "testparam1", "value_1" )

' Run the transformation. The PXF file path can be relative to the working folder or
absolute.
WScript.Echo("Generating output from ExpReport.pxf...")
If ( objSVS.Generate( "ExpReport.pxf" ) ) Then
   WScript.Echo( objSVS.LastExecutionMessage )
   WScript.Echo( "Success - finished execution" )
Else
   WScript.Echo( objSVS.LastExecutionMessage )
End If

' handle COM errors
If Err.Number <> 0 Then
   WScript.Echo("Internal error - " & Err.Description )
   WScript.Quit -1
End If

On Error Goto 0    ' re-enble default error promotion
```

5.4.5 Visual Basic

Im Beispiel unten wird gezeigt, wie Sie über C++-Code anhand einer PXF-Datei und einer XML-Input-Datei eine RTF-Output-Datei generieren. Stellen Sie sicher, dass StyleVision Server installiert und lizenziert ist und als COM-Servereobjekt zur Verfügung steht. Die Registrierung als COM-Serverobjekt erfolgt normalerweise während der Installation von StyleVision Server. Eine Anleitung, wie Sie überprüfen, ob die Registrierung erfolgreich war, finden Sie unter Informationen zur COM-Schnittstelle 62.

```
Option Explicit On

Module Program

    Sub Main()

        Try
            'Create a StyleVision Server object
            Dim objSVS As Altova.StyleVisionServer.Server = New
Altova.StyleVisionServer.Server

            'Set a working directory - used for output and for intermediate files
            'objSVS.WorkingDirectory = "C:\Program Files (x86)
\Altova\MapForceServer2020\etc\Examples"
            objSVS.WorkingDirectory = "..\..\.."

            'Default path to the StyleVision Server executable is the installation path
(same dir with the StyleVisionServer.dll)
            'In case you moved the binaries on the disk, you need to explicitly set the
path to the .exe file
            'The ServerPath should be set before any other properties because it re-
initializes the server and resets all properties.
            'objSVS.ServerPath = "C:\Program Files (x86)
\Altova\StyleVisionServer2020\bin\StyleVisionServer.exe"
            'objSVS.ServerPath = "C:\Program
Files\Altova\StyleVisionServer2020\bin\StyleVisionServer.exe"

            'Prepare the name of the working XML
            '    This can be an absolute/relative path if the file is stored externally
(not inside PXF)
            ' objSVS.InputXML = "ExpReport.xml"
            '    Or it can contain the path INSIDE the PXF
            objSVS.InputXML = "ExpReport.pxf|zip\ExpReport.xml"
            '    Easiest way is to refer to the file as being embedded in the
transformation file
            ' objSVS.InputXML = "altova://packagedfile/ExpReport.xml"

            'Add output paths (absolute or relative to WorkingDirectory) for all formats
that should be generated
            objSVS.OutputRTF = "C:\tmp\ExpReport.rtf"
            objSVS.OutputPDF = "C:\tmp\ExpReport.pdf"
            objSVS.OutputHTML = "C:\tmp\ExpReport.html"

            'Prepare the parameters, if your design uses parameters
            'objSVS.AddParameter( "testparam1", "value 1" )

            ' Run the transformation; the output will be stored at C:\temp
            If (objSVS.Generate("ExpReport.pxf")) Then
                System.Console.WriteLine(objSVS.LastExecutionMessage)
                System.Console.WriteLine("Success - finished execution")
            Else
```

 © 2019-2025 Altova GmbH

```
                System.Console.WriteLine(objSVS.LastExecutionMessage)
            End If

        Catch ex As Exception
            System.Console.WriteLine("Internal Error - " & ex.Message())
        End Try

    End Sub

End Module
```

5.5 API-Referenz

Dieser Abschnitt ist eine Benutzerreferenz für die StyleVision Server API.

- COM und .NET 74
- Java 81

5.5.1 COM und .NET

Die StyleVisionServer API stellt die **Schnittstelle `IServer`**, zur Verfügung, über die Sie eine neue StyleVision Server-Objektinstanz erstellen und Zugriff auf StyleVision Server erhalten.

Die Schnittstelle `IServer` hat die folgenden Methoden und Parameter.

Methoden

© 2019-2025 Altova GmbH

AddParameter

Weist einem in der PXF-Datei definierten Parameter einen Wert zu.

C#

```
void AddParameter(string bstrName, string bstrValue)
```

C++

```
HRESULT AddParameter([in] BSTR bstrName, [in] BSTR bstrValue );
```

VB

```
Sub AddParameter(ByVal bstrName As String, ByVal bstrValue As String)
```

ClearParameterList

Löscht die Liste der Parameter.

C#

```
void ClearParameterList()
```

C++

```
HRESULT ClearParameterList();
```

VB

```
Sub ClearParameterList()
```

Generate

Generiert anhand der mit `TransfPath` definierten PXF-Datei eine oder mehrere Ausgabedateien (HTML, PDF, RTF, und/oder DOCX). Der Befehl transformiert die XML-Input-Datei (in der PXF-Datei der XML-Arbeitsdatei) anhand des in der PXF-Datei enthaltenen XSLT-Dokuments. Gibt bei Erfolg `TRUE`, andernfalls `FALSE` zurück.

C#

```
bool Generate(string bstrTransfPath)
```

C++

```
HRESULT Generate( [in] BSTR bstrTransfPath, [out, retval] VARIANT_BOOL*
pbSuccess );
```

VB

```
Function Generate(ByVal bstrTransfPath As String) As Boolean
```

Eigenschaften

APIMajorVersion

Ruft die Hauptversion der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

C#

```
int APIMajorVersion { get; }
```

C++

```
HRESULT APIMajorVersion([out, retval] INT* pnVal);
```

VB

```
ReadOnly Property APIMajorVersion As Integer
```

APIMinorVersion

Ruft die Nebenversion der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

C#

```
int APIMinorVersion { get; }
```

C++

```
HRESULT APIMinorVersion([out, retval] INT* pnVal);
```

VB

```
ReadOnly Property APIMinorVersion As Integer
```

APIServicePackVersion

Ruft die Service Pack-Version der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

C#

```
int APIServicePackVersion { get; }
```

C++

```
HRESULT APIServicePackVersion([out, retval] INT* pnVal);
```

VB

```
ReadOnly Property APIServicePackVersion As Integer
```

InputXML

Definiert den Namen und Pfad der zu verarbeitenden XML-Datei (der XML-Arbeitsdatei in der PXF-Datei).

C#

```
string InputXML { set; }
```


© 2019-2025 Altova GmbH

C++

```
HRESULT InputXML([in] BSTR bstrPath );
```

VB

```
Property InputXML As String
```

Is64Bit

Gibt `TRUE` zurück, wenn der StyleVision Server-Prozessor eine ausführbare 64-Bit-Datei ist.

C#

```
bool Is64Bit { get; }
```

C++

```
HRESULT Is64Bit([out, retval] VARIANT_BOOL* pbVal);
```

VB

```
ReadOnly Property Is64Bit As Boolean
```

LastExecutionMessage

Ruft die beim letzten `Generate`-Befehl erhaltene Meldung auf.

C#

```
string LastExecutionMessage { get; }
```

C++

```
HRESULT LastExecutionMessage([out, retval] BSTR* pbstrResult );
```

VB

```
ReadOnly Property LastExecutionMessage As String
```

MajorVersion

Ruft die Hauptversion von StyleVision Server ab.

C#

```
int MajorVersion { get; }
```

C++

```
HRESULT MajorVersion([out, retval] INT* pnVal);
```

VB

```
ReadOnly Property MajorVersion As Integer
```

MinorVersion

Ruft die Nebenversion von StyleVision Server ab.

C#

```
int MinorVersion { get; }
```

C++

```
HRESULT MinorVersion([out, retval] INT* pnVal);
```

VB

```
ReadOnly Property MinorVersion As Integer
```

OutputDOCX

Definiert den Namen und Pfad der DOCX-Ausgabedatei.

C#

```
string OutputDOCX { set; }
```

C++

```
HRESULT OutputDOCX([in] BSTR bstrPath );
```

VB

```
Property OutputDOCX As String
```

OutputFO

Definiert den Namen und Pfad der FO-Ausgabedatei.

C#

```
string OutputFO { set; }
```

C++

```
HRESULT OutputFO([in] BSTR bstrPath );
```

VB

```
Property OutputFO As String
```

OutputHTML

Definiert den Namen und Pfad der HTML-Ausgabedatei.

C#

```
string OutputHTML { set; }
```

C++

```
HRESULT OutputHTML([in] BSTR bstrPath );
```

VB

© 2019-2025 Altova GmbH

```
Property OutputHTML As String
```

OutputPDF

Definiert den Namen und Pfad der PDF-Ausgabedatei.

C#

```
string OutputPDF { set; }
```

C++

```
HRESULT OutputPDF([in] BSTR bstrPath );
```

VB

```
Property OutputPDF As String
```

OutputRTF

Definiert den Namen und Pfad der RTF-Ausgabedatei.

C#

```
string OutputRTF { set; }
```

C++

```
HRESULT OutputRTF([in] BSTR bstrPath );
```

VB

```
Property OutputRTF As String
```

OutputText

Definiert den Namen und Pfad der Textausgabedatei.

C#

```
string OutputText { set; }
```

C++

```
HRESULT OutputRTF([in] BSTR bstrPath );
```

VB

```
Property OutputRTF As String
```

ProductName

Ruft den Namen des Produkts ab: "StyleVision Server"

C#

```
string ProductName { get; }
```

C++

```
HHRESULT ProductName([out, retval] BSTR* pstrVal);
```

VB

```
ReadOnly Property ProductName As String
```

ProductNameAndVersion

Ruft den vollständigen Namen des Produkts einschließlich Versionsnummer ab: `"StyleVision Server 2014r2 sp1 (x64)"`.

C#

```
string ProductNameAndVersion { get; }
```

C++

```
HHRESULT ProductNameAndVersion([out, retval] BSTR* pstrVal);
```

VB

```
ReadOnly Property ProductNameAndVersion As String
```

ServerPath

Ruft den Pfad zur ausführbaren StyleVision Server Datei ab oder definiert ihn.

C#

```
string ServerPath { set; get; }
```

C++

```
HRESULT ServerPath([in] BSTR bstrServerFile );
HRESULT ServerPath([out, retval] BSTR* pbstrServerFile );
```

VB

```
Property ServerPath As String
```

ServicePackVersion

Ruft die Service Pack-Version von StyleVision Server ab (z.B.: **1** für `Altova StyleVision Server 2014 r2 sp1 (x64)`.)

C#

```
int ServicePackVersion { get; }
```

C++

```
HRESULT ServicePackVersion([out, retval] INT* pnVal);
```


© 2019-2025 Altova GmbH

VB

```
ReadOnly Property ServicePackVersion As Integer
```

WhereClause

Definiert eine SQL `WHERE`-Klausel, mit der festgelegt wird, welche Zeilen einer DB-XML-Schemaquelle verarbeitet werden sollen.

C#

```
string WhereClause { set; }
```

C++

```
HRESULT WhereClause([in] BSTR bstrPath );
```

VB

```
Property WhereClause As String
```

WorkingDirectory

Ruft das aktuelle Verzeichnis für laufende Aufträge ab oder definiert dieses. Relative Pfade werden anhand des Arbeitsverzeichnisses ausgewertet.

C#

```
string WorkingDirectory { set; get; }
```

C++

```
HRESULT WorkingDirectory([in] BSTR bstrWorkingDirectory );
HRESULT WorkingDirectory([out, retval] BSTR* pbstrWorkingDirectory );
```

VB

```
Property WorkingDirectory As String
```

5.5.2 Java

Das Paket `com.altova.stylevisionserver` besteht aus den folgenden Klassen:

- `public class` **`StyleVisionServer`** *(siehe Beschreibung unten)*
- `public class` **`StyleVisionServerException`** `extends` **`Exception`**

Die Klasse `StyleVisionServer`
Die Klasse `StyleVisionServer` erstellt eine neue StyleVision Server Objektinstanz und bietet Zugriff auf StyleVision Server. Im Folgenden finden Sie eine Beschreibung der Methoden der Schnittstelle `StyleVisionServer`.

Methoden der Klasse `StyleVisionServer`

Im Folgenden finden Sie eine alphabetisch geordnete Liste der Methoden der Klasse `StyleVisionServer`.

addParameter

```
public void addParameter(String name, String value)
```

Fügt den Namen und Wert eines neuen Parameters hinzu. Jeder Parameter und sein Wert werden in einem separaten Methodenaufruf definiert. Parameter müssen im XSLT-Dokument deklariert sein.

Parameter:

`name`: Enthält den Namen des Parameters als String.
`value`: Enthält den Wert des Parameters als String.

clearParameterList

```
public void clearParameterList()
```

Löscht die Liste der Parameter.

generate

```
public boolean generate(String transfPath)
```

Verarbeitet die in `transfPath` definierte PXF-Datei. Löst `StyleVisionServerException` aus.

Parameter:

`transfPath`: Eine absolute URL, die den Pfad zur <%PXF%>-Datei angibt.

Rückgabe:

`true()` bei erfolgreicher Ausführung
`false()` wenn die Ausführung fehlschlägt
Verwenden Sie bei einem Fehler `getLastExecutionMessage()`

getAPIMajorVersion

```
public int getAPIMajorVersion()
```

Ruft die Hauptversion der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

getAPIMinorVersion

```
public int getAPIMinorVersion()
```

Ruft die Nebenversion der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

getAPIServicePackVersion

```
public int getAPIServicePackVersion()
```

Ruft die Service Pack-Version der StyleVision Server API ab. Sie unterscheidet sich eventuell von der Produktversion, wenn die API mit einem anderen Server verbunden ist.

getLastExecutionMessage

```
public String getLastExecutionMessage()
```

Ruft die beim letzten generate-Befehl erhaltene Meldung auf.

© 2019-2025 Altova GmbH

getMajorVersion

```
public int getMajorVersion()
```
Ruft die Hauptversion der Applikation ab.

getMinorVersion

```
public int getMinorVersion()
```
Ruft die Nebenversion der Applikation ab.

getProductName

```
public String getProductName()
```
Ruft den Produktnamen ab.

getProductNameAndVersion

```
public String getProductNameAndVersion()
```
Ruft den vollständigen Namen und die Versionsnummer des Produkts ab.

getServerPath

```
public String getServerPath()
```
Ruft den Pfad zur ausführbaren Binärdatei des Servers ab.

getServicePackVersion

```
public int getServicePackVersion()
```
Ruft die Service Pack-Version von StyleVision Server ab.

getWorkingDirectory

```
public String getWorkingDirectory()
```
Ruft das aktuelle Arbeitsverzeichnis ab.

is64bit

```
public boolean is64bit()
```
Überprüft, ob die ausführbare Datei eine 64-Bit-Datei ist.
Rückgabe:
`true()` für StyleVision Server (x64), andernfalls `false()`.

setInputXML

```
public void setInputXML(String path)
```
Definiert die zu verarbeitende XML-Datei. Dabei muss es sich um den Pfad der in der PXF-Datei definierten XML-Arbeitsdatei handeln.
Parameter:
`path`: Enthält den Pfad der XML-Arbeitsdatei in der PXF-Datei.

setOutputDOCX

```
public void setOutputDOCX(String path)
```
Definiert den Namen und Pfad der zu generierenden `DOCX`-Datei.
Parameter:

path: Der Name und Pfad der zu generierenden DOCX-Datei.

setOutputFO

```
public void setOutputFO(String path)
```

Definiert den Namen und Pfad der zu generierenden FO-Datei.
Parameter:
path: Der Name und Pfad der zu generierenden FO-Datei.

setOutputHTML

```
public void setOutputHTML(String path)
```

Definiert den Namen und Pfad der zu generierenden HTML-Datei.
Parameter:
path: Der Name und Pfad der zu generierenden HTML-Datei.

setOutputPDF

```
public void setOutputPDF(String path)
```

Definiert den Namen und Pfad der zu generierenden PDF-Datei.
Parameter:
path: Der Name und Pfad der zu generierenden PDF-Datei.

setOutputRTF

```
public void setOutputRTF(String path)
```

Definiert den Namen und Pfad der zu generierenden RTF-Datei.
Parameter:
path: Der Name und Pfad der zu generierenden RTF-Datei.

setOutputText

```
public void setOutputText(String path)
```

Definiert den Namen und Pfad der zu generierenden Textdatei.
Parameter:
path: Der Name und Pfad der zu generierenden Textdatei.

setServerPath

```
public void setServerPath(String serverFile)
```

Definiert den Pfad der ausführbaren StyleVisionServer Datei.
Parameter:
serverFile: Der Pfad der ausführbaren StyleVisionServer Datei.

setWhereClause

```
public void setWhereClause(String whereClause)
```

Definiert eine SQL WHERE-Klausel, mit der festgelegt wird, welche Zeilen einer DB-XML-Schemaquelle verarbeitet werden sollen.
Parameter:
whereClause: Die SQL WHERE-Klausel, die angibt, welche Zeilen einer DB-XML-Schemaquelle verarbeitet werden sollen.

setWorkingDirectory

© 2019-2025 Altova GmbH

```
public void setWorkingDirectory(String workingDirectory)
```
Definiert ein Standardverzeichnis. Relative Pfade werden relativ zu diesem Verzeichnis aufgelöst.
Parameter:
`workingDirectory`: Der Pfad des Standard (Arbeits)-Verzeichnisses

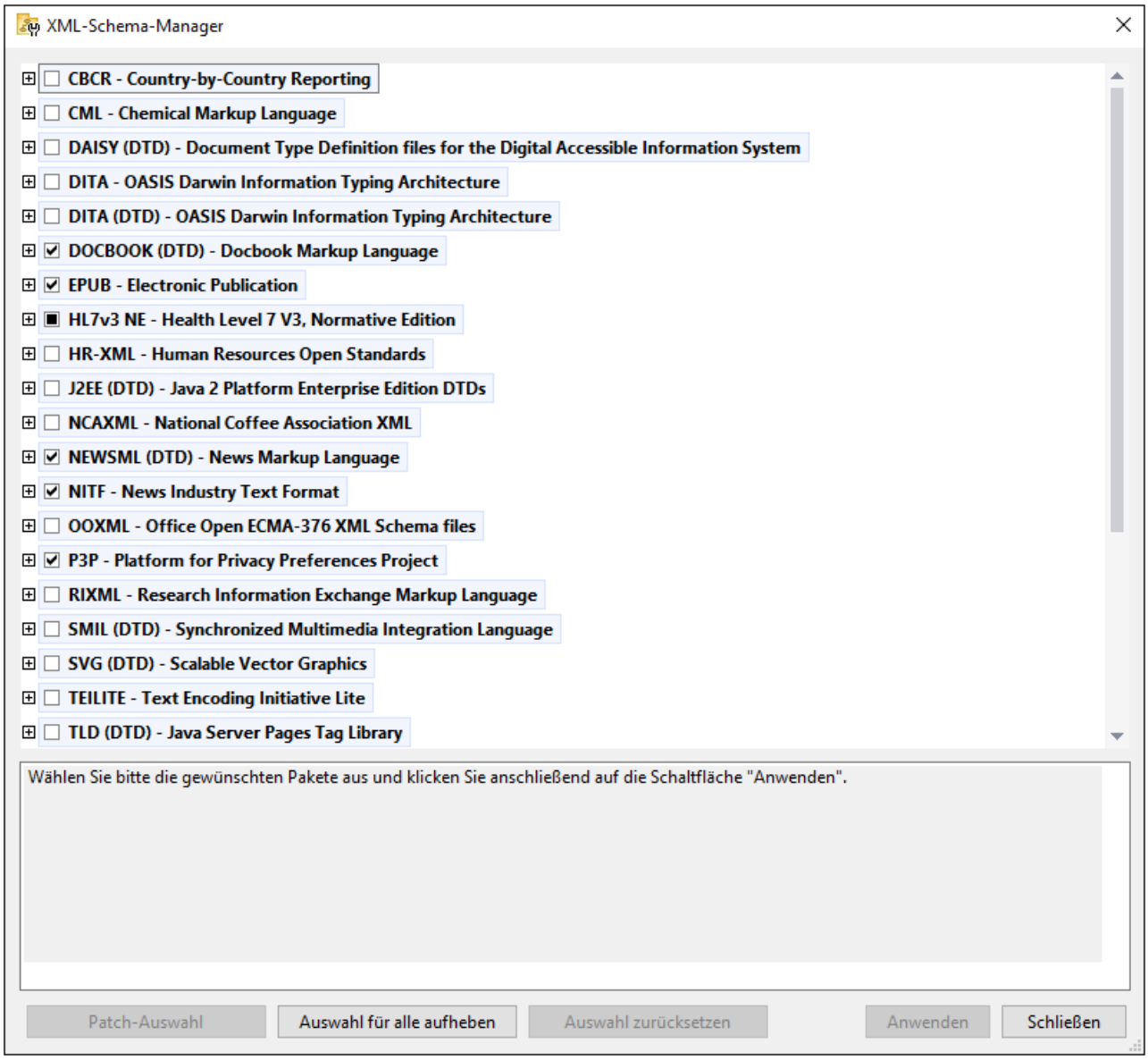
stop

```
public void stop()
```
Beendet den Server-Prozess.

6 Schema-Manager

Der XML-Schema-Manager ist ein Altova-Tool, mit dem Sie XML-Schemas (DTDs für XML-Dateien und XML-Schemas) zentral installieren und verwalten können, um diese in allen XML-Schema-fähigen Applikationen von Altova einschließlich StyleVision Server verwenden zu können.

- Unter Windows hat der Schema-Manager eine grafische Benutzeroberfläche (*siehe Abbildung unten*) und steht auch über die Befehlszeile zur Verfügung. (Die Desktop-Applikationen von Altova stehen nur unter Windows zur Verfügung; *siehe Liste unten*).
- Unter Linux und MacOS steht der Schema-Manager nur über die Befehlszeile zur Verfügung. (Die Server-Applikationen von Altova stehen unter Windows, Linux und macOS zur Verfügung; *siehe Liste unten*).



 © 2019-2025 Altova GmbH

Altova-Applikationen, die mit Schema-Manager arbeiten:

Desktop-Applikationen (nur Windows)	Server-Applikationen (Windows, Linux, macOS)
XMLSpy (alle Editionen)	RaptorXML Server, RaptorXML+XBRL Server
MapForce (alle Editionen)	StyleVision Server
StyleVision (alle Editionen)	
Authentic Desktop Enterprise Edition	

Installation und Deinstallation des Schema-Managers

Der Schema-Manager wird bei der ersten Installation einer neuen Version des Altova Mission Kit oder einer der XML-Schema-fähigen Applikationen von Altova (*siehe Tabelle oben*) automatisch installiert.

Ebenso wird er auch automatisch entfernt, wenn Sie die letzte XML-Schema-fähige Applikation von Altova auf Ihrem Rechner deinstallieren.

Schema-Manager-Funktionalitäten

Im Schema-Manager stehen die folgenden Funktionalitäten zur Verfügung:

- Anzeigen der auf Ihrem Rechner installierten XML-Schemas und Überprüfung, ob neue Versionen zum Download zur Verfügung stehen.
- Download neuer Versionen von XML-Schemas unabhängig vom Altova Produkt-Release-Zyklus. (Die Schemas werden von Altova online bereitgestellt und können über den Schema-Manager heruntergeladen werden).
- Installation oder Deinstallation jeder beliebigen (oder ggf. aller) der zahlreichen Versionen eines bestimmten Schemas.
- Ein XML-Schema kann Abhängigkeiten von anderen Schemas aufweisen. Bei der Installation oder Deinstallation eines bestimmten Schemas informiert Sie der Schema-Manager über davon abhängige Schemas und installiert bzw. entfernt diese ebenfalls automatisch.
- Der Schema-Manager ordnet Schema-Referenzen mit Hilfe des XML-Katalogs lokalen Dateien zu. Dadurch lassen sich große XML-Schemas schneller verarbeiten, als wenn sie sich unter einem entfernten Pfad befinden.
- Alle wichtigen Schemas werden über den Schema-Manager bereitgestellt und regelmäßig auf die jeweils neuesten Version aktualisiert. Dadurch können alle Ihre Schemas zentral verwaltet werden und stehen allen XML-Schema-fähigen Applikationen von Altova jederzeit zur Verfügung.
- Im Schema-Manager vorgenommene Änderungen werden für alle auf dem Rechner installierten Altova-Produkte wirksam.
- Wenn Sie versuchen ein Dokument in einem Altova-Produkt anhand eines nicht installierten aber über Schema-Manager verfügbaren Schemas zu validieren, wird das Schema automatisch installiert. Wenn das Schema-Paket jedoch Namespace-Zuordnungen enthält, wird das Schema nicht automatisch installiert; in diesem Fall müssen Sie Schema-Manager starten, das/die gewünschte(n) Paket(e) auswählen und die Installation starten. Wenn Ihre offene Altova-Applikation nach der Installation nicht automatisch neu gestartet wird, müssen Sie sie manuell neu starten.

Funktionsweise

Alle in Altova-Produkten verwendeten XML-Schemas werden von Altova online bereitgestellt. Dieser Speicher wird bei Veröffentlichung neuer Versionen der Schemas aktualisiert. Im Schema-Manager werden sowohl bei Aufruf über die Benutzeroberfläche als auch über das CLI Informationen über die neuesten verfügbaren Schemas angezeigt. Sie können die gewünschten Schemas dann über den Schema-Manager installieren, aktualisieren oder deinstallieren.

Schemas können vom Schema-Manager auch auf eine weitere Art installiert werden. Sie können ein Schemas und die davon abhängigen Schemas auf der Altova Website (https://www.altova.com/de/schema-manager) auswählen. Daraufhin wird auf der Website eine Datei des Typs **.altova_xmlschemas** mit Informationen über Ihre ausgewählten Schemas zum Download vorbereitet. Bei Doppelklick auf diese Datei oder bei Übergabe an den Schema-Manager über das CLI als Argument des Befehls `install`[99] installiert der Schema-Manager die ausgewählten Schemas.

Lokaler Cache: Überprüfung Ihrer Schemas

Alle Informationen über installierte Schemas werden in einem zentralen Cache-Verzeichnis auf Ihrem Rechner aufgezeichnet. Das Verzeichnis befindet sich hier:

Windows	`C:\ProgramData\Altova\pkgs\.cache`
Linux	`/var/opt/Altova/pkgs\.cache`
macOS	`/var/Altova/pkgs`

Dieses Cache-Verzeichnis wird regelmäßig mit dem neuesten Status der Schemas aus dem Online-Speicher von Altova aktualisiert. Diese Aktualisierungen finden unter den folgenden Bedingungen statt:

- bei jedem Start von Schema-Manager.
- Wenn Sie StyleVision Server zum ersten Mal an einem bestimmten Kalendertag starten.
- Wenn StyleVision Server länger als 24 Stunden geöffnet ist, findet alle 24 Stunden eine Aktualisierung des Cache statt.
- Sie können den Cache auch durch Ausführung des `update`[102]-Befehls über die Befehlszeilenschnittstelle aktualisieren.

Der Schema-Manager kann somit Ihre installierten Schemas über den Cache ständig anhand der online verfügbaren Schemas auf der Altova Website überprüfen.

Nehmen Sie keine manuellen Änderungen am Cache vor!

Das lokale Cache-Verzeichnis wird automatisch auf Basis der installierten oder deinstallierten Schemas verwaltet; es sollte nicht manuell geändert oder gelöscht werden. Falls Sie den Schema-Manager je in seinen Originalzustand zurücksetzen möchten, (i) führen Sie den CLI-Befehl `reset`[100] der Befehlszeilenschnittstelle und (ii) anschließend den Befehl `initialize`[98] aus. (Führen Sie alternativ dazu den Befehl **reset** mit der Option **-i** aus).

© 2019-2025 Altova GmbH

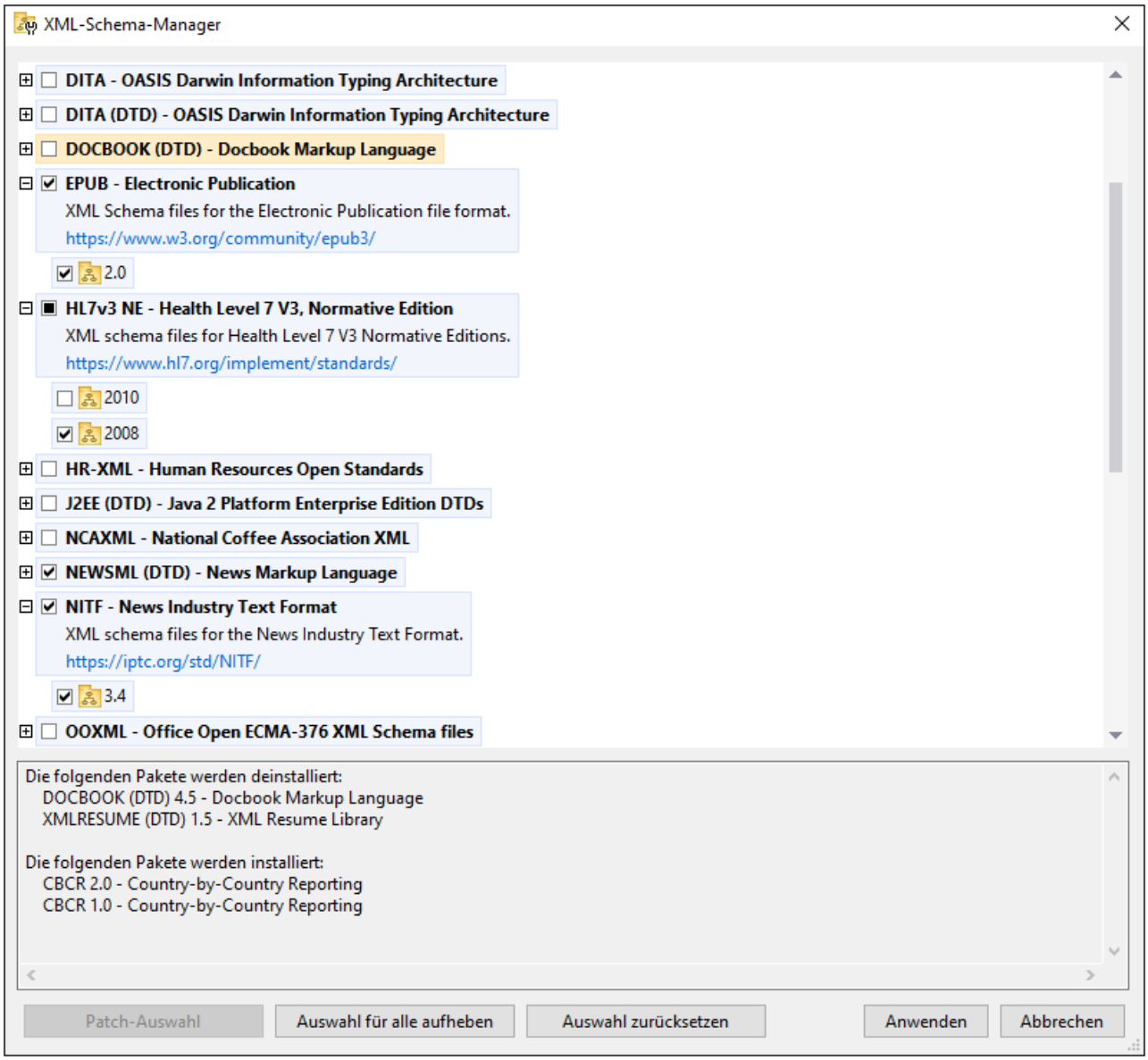
6.1 Ausführen des Schema-Managers

Grafische Benutzeroberfläche

Sie können die Benutzeroberfläche des Schema-Managers auf eine der folgenden Arten aufrufen:

- *Bei der Installationen von StyleVision Server:* Aktivieren Sie gegen Ende der Installation das Kontrollkästchen *Altova-Schema-Manager aufrufen*, wodurch Sie die Benutzeroberfläche des Schema-Managers direkt aufrufen können. Auf diese Art können Sie Schemas während der Installation Ihrer Altova-Applikation installieren.
- über die von der Altova Webseite heruntergeladene **`.altova_xmlschemas`**-Datei: Doppelklicken Sie auf die heruntergeladene Datei, um den Schema-Manager zu starten, der daraufhin die (auf der Website) ausgewählten Schemas installiert.

Nachdem die Benutzeroberfläche des Schema-Managers geöffnet wurde (*Abbildung unten*), werden bereits installierte Schemas markiert angezeigt. Wenn ein zusätzliches Schema installiert werden soll, aktivieren Sie dieses. Wenn ein bereits installiertes Schema deinstalliert werden soll, deaktivieren Sie dieses. Nachdem Sie Ihre Auswahl getroffen haben, können Ihre Änderungen angewendet werden. Die Schemas, die installiert bzw. deinstalliert werden, werden markiert und im Fenster "Meldungen" am unteren Rand des Schema-Manager-Fensters (*siehe Abbildung*) erscheint eine Meldung über die bevorstehenden Änderungen.



Wenn Sie auf **Anwenden** klicken, wird der Fortschritt der Installation angezeigt. Bei Auftreten eines Fehlers (z.B. eines Verbindungsfehlers) wird eine Fehlermeldung angezeigt. Klicken Sie in diesem Fall auf die Schaltfläche **Erweitert**, die im Dialogfeld erscheint, überprüfen Sie die Schemaauswahl und versuchen Sie es erneut durch Klicken auf **Anwenden**.

Befehlszeilenschnittstelle

Sie können den Schema-Manager über eine Befehlszeilenschnittstelle starten, indem Sie Befehle an die ausführbare Datei **`xmlschemamanager.exe`** senden.

Die Datei **`xmlschemamanager.exe`** steht im folgenden Ordner zur Verfügung:

- *Unter Windows:* `C:\ProgramData\Altova\SharedBetweenVersions`

© 2019-2025 Altova GmbH

- *Unter Linux oder macOS (nur Server-Applikationen):* `%INSTALLDIR%/bin`, wobei `%INSTALLDIR%` das Installationsverzeichnis des Programms ist.

Anschließend können Sie jeden der im Abschnitt zur CLI-Befehlsreferenz 97 aufgelisteten Befehle verwenden.

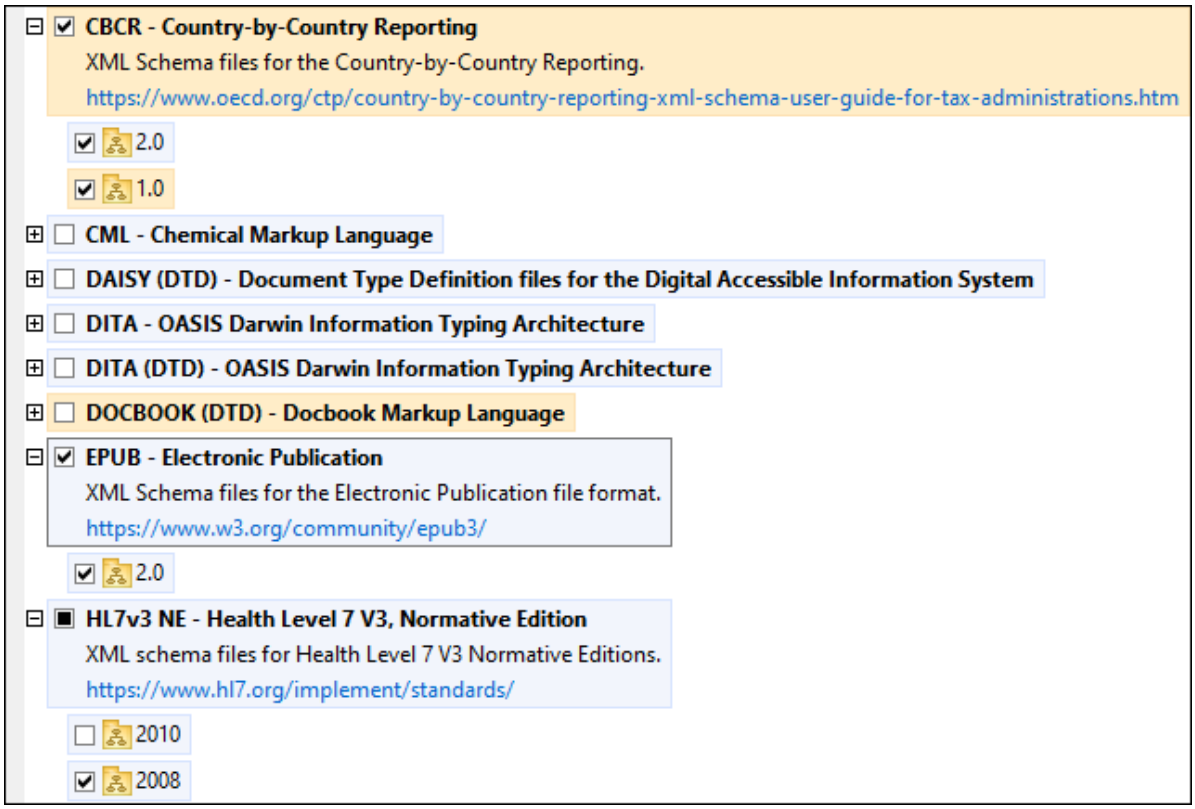
Um die Hilfe zu den Befehlen anzuzeigen, führen Sie den folgenden Befehl aus:

- *Unter Windows:* `xmlschemamanager.exe --help`
- *Unter Linux oder macOS (nur Server-Applikationen):* `sudo ./ xmlschemamanager --help`

6.2 Statuskategorien

Der Schema-Manager unterscheidet folgendermaßen zwischen den von ihm verwalteten Schemas:

- *installierte Schemas.* Diese werden auf der Benutzeroberfläche mit einem Häkchen angezeigt (*in der Abbildung unten sind die mit einem Häkchen versehenen und blau angezeigten Versionen der EPUB- und HL7v3 NE-Schemas installiert*). Wenn alle Versionen eines Schemas ausgewählt sind, wird ein Häkchen angezeigt. Wenn zumindest eine Version nicht ausgewählt ist, wird ein gefülltes Quadrat angezeigt. Sie können das Kontrollkästchen für ein installiertes Schema deaktivieren, um es zu **deinstallieren**; (*in der Abbildung unten ist die DocBook DTD installiert und wurde deaktiviert; sie wird daher für die Deinstallation vorbereitet*).
- *Nicht installierte verfügbare Schemas.* Diese werden auf der Benutzeroberfläche mit einem deaktivierten Kontrollkästchen angezeigt. Sie können die Schemas, die **installiert** werden sollen, auswählen.



- *Schemas, für die ein Upgrade zur Verfügung steht* sind diejenigen, die seit ihrer Installation vom Herausgeber überarbeitet wurden. Sie werden auf der Benutzeroberfläche durch ein -Symbol gekennzeichnet. Sie können für ein installiertes Schema ein **Patch** der verfügbaren überarbeiteten Version installieren.

Wichtige Punkte

- In der Abbildung oben sind beide CBCR-Schemas ausgewählt. Dasjenige mit einem blauen Hintergrund ist bereits installiert. Dasjenige mit dem gelben Hintergrund ist nicht installiert und wurde für die Installation ausgewählt. Beachten Sie, dass das Schema "HL7v3 NE 2010" nicht installiert ist und nicht für die Installation ausgewählt wurde.

© 2019-2025 Altova GmbH

- Ein gelber Hintergrund bedeutet, dass das Schema auf irgendeine Art geändert wird, wenn Sie auf die Schaltfläche **Anwenden** klicken. Wenn ein Schema deaktiviert ist und einen gelben Hintergrund hat, bedeutet dies, dass es bei Klick auf die Schaltfläche **Anwenden** deinstalliert wird. In der Abbildung oben ist dies bei der DocBook DTD der Fall.
- Bei Ausführung des Schema-Managers über die Befehlszeile wird der Befehl `list` 100 mit verschiedenen Optionen verwendet, um verschiedene Schemakategorien aufzulisten:

`xmlschemamanager.exe list`	Listet alle installierten und verfügbaren Schemas auf; auch verfügbare Upgrades werden angezeigt.
`xmlschemamanager.exe list -i`	Listet nur installierte Schemas auf; auch verfügbare Upgrades werden angezeigt.
`xmlschemamanager.exe list -u`	Listet Schemas auf, für die Upgrades zur Verfügung stehen

Anmerkung: Verwenden Sie unter Linux und macOS **`sudo ./xmlschemamanager list`**

6.3 Anwenden eines Patch oder Installation eines Schemas

Anwenden eines Patch auf ein installiertes Schema

Von Zeit zu Zeit werden von den Herausgebern der XML-Schemas Patches (Upgrades oder Überarbeitungen) veröffentlicht. Wenn der Schema-Manager erkennt, dass Patches zur Verfügung stehen, werden diese in der Schemaliste des Schema-Managers angezeigt und Sie können diese Patches schnell installieren.

Über die Benutzeroberfläche

Patches werden mit dem Symbol gekennzeichnet. (*Siehe auch vorhergehendes Kapitel über Statuskategorien*[92]*).* Falls Patches zur Verfügung stehen, ist die Schaltfläche **Patch-Auswahl** aktiv. Klicken Sie darauf, um alle Patches für die Installation auszuwählen und vorzubereiten. Auf der Benutzeroberfläche ändert sich das Symbol von Schemas, für die ein Patch installiert wird von in , und im Fenster "Meldungen" am unteren Rand des Dialogfelds werden die Patches, die angewendet werden, aufgelistet. Sobald Sie mit der Auswahl fertig sind, klicken Sie auf **Anwenden**. Alle Patches werden gemeinsam angewendet. Beachten Sie, dass ein für die Installation eines Patch markiertes Schema deinstalliert wird, wenn Sie die Auswahl aufheben.

Über das CLI

So wenden Sie einen Patch über die Befehlszeilenschnittstelle an:

1. Führen Sie den Befehl `list -u`[100] aus. Daraufhin werden alle Schemas, für die Upgrades zur Verfügung stehen, aufgelistet.
2. Führen Sie den Befehl `upgrade`[103] aus, um alle Patches zu installieren.

Installieren eines verfügbaren Schemas

Sie können Schemas entweder über die Benutzeroberfläche des Schema-Managers oder durch Senden der Schema-Manager-Installationsbefehle über die Befehlszeile installieren.

Anmerkung: Wenn das aktuelle Schema andere Schemas referenziert, werden auch die referenzierten Schemas installiert.

Über die Benutzeroberfläche

Um Schemas über die Benutzeroberfläche des Schema-Managers zu installieren, wählen Sie die gewünschten Schemas aus und klicken Sie auf **Anwenden**.

Sie können die gewünschten Schemas auch auf der Altova Website auswählen und eine herunterladbare **`.altova_xmlschemas`**-Datei generieren. Bei Doppelklick auf diese Datei wird der Schema-Manager aufgerufen, in dem die gewünschten Schemas bereits vorausgewählt sind. Sie müssen nur mehr auf **Anwenden** klicken.

Über das CLI

Um Schemas über die Befehlszeile zu installieren, rufen Sie den Befehl `install`[99] auf:

```
xmlschemamanager.exe install [options] Schema+
```

wobei es sich bei **`Schema`** um das/die gewünschte(n) Schema(s) bzw. eine **`.altova_xmlschemas`**-Datei handelt. Ein Schema wird von einem Identifier im Format **`<name>-<version>`** referenziert. (Die Identifier von

© 2019-2025 Altova GmbH

Schemas werden angezeigt, wenn Sie den Befehl `list`[100] ausführen.) Sie können beliebig viele Schemas eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls `install`[99].

Anmerkung: Verwenden Sie unter Linux oder macOS den Befehl **`sudo ./xmlschemamanager`**.

Installation eines benötigten Schemas

Wenn Sie einen XML-Schema-Befehl in StyleVision Server ausführen und StyleVision Server erkennt, dass ein zur Ausführung des Befehls erforderliches Schema nicht vorhanden oder unvollständig ist, wird der Schema-Manager mit Informationen über das/die fehlende(n) Schema(s) aufgerufen. Sie können die gewünschten Schemas dann über den Schema-Manager direkt installieren.

Alle bereits installierten Schemas können jederzeit durch Aufruf des Schema-Managers über **Extras | Schema-Manager** über die Benutzeroberfläche des Schema-Managers angezeigt werden.

6.4 Deinstallieren eines Schemas, Zurücksetzen

Deinstallieren eines Schemas

Sie können Schemas entweder über die Benutzeroberfläche des Schema-Managers oder durch Senden der Schema-Manager-Deinstallationsbefehle über die Befehlszeile deinstallieren.

Anmerkung: Wenn das gewünschte Schema andere Schemas referenziert, so werden auch die referenzierten Schemas deinstalliert.

Über die Benutzeroberfläche

Um Schemas über die Benutzeroberfläche des Schema-Managers zu deinstallieren, deaktivieren Sie die Kontrollkästchen der entsprechenden Schemas und klicken Sie auf **Anwenden**. Daraufhin werden die ausgewählten Schemas und die davon referenzierten Schemas deinstalliert.

Um alle Schemas zu deinstallieren, klicken Sie auf **Auswahl für alle aufheben** und anschließend auf **Anwenden**.

Über das CLI

Um Schemas über die Befehlszeile zu deinstallieren, rufen Sie den Befehl `uninstall`[101] auf:

```
xmlschemamanager.exe uninstall [options] Schema+
```

wobei es sich beim Argument **`Schema`** ein zu deinstallierendes Schema oder eine **`.altova_xmlschemas`**-Datei handelt. Ein Schema wird von einem Identifier im Format **`<name>-<version>`** definiert. (Die Identifier von Schemas werden angezeigt, wenn Sie den Befehl `list`[100] ausführen.) Sie können beliebig viele Schemas eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls `uninstall`[101].

Anmerkung: Verwenden Sie unter Linux oder macOS den Befehl **`sudo ./xmlschemamanager`**.

Zurücksetzen des Schema-Managers

Sie können den Schema-Manager zurücksetzen. Damit werden alle installierten Schemas und das Cache-Verzeichnis entfernt.

- Klicken Sie auf der Benutzeroberfläche auf **Auswahl zurücksetzen**.
- Führen Sie über die Benutzeroberfläche den Befehl `reset`[100] aus.

Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize`[98] ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl `reset`[100] mit der Option `-i` aus.

Beachten Sie, dass mit `reset-i`[100] die Originalinstallation des Produkts wiederhergestellt wird, daher wird empfohlen, nach dem Zurücksetzen auch den Befehl `update`[102] auszuführen. Führen Sie alternativ dazu den Befehl `reset`[100] mit den Optionen `-i` und `-u` aus.

© 2019-2025 Altova GmbH

6.5 Befehlszeilenschnittstelle (CLI)

Um den Schema-Manager über die Befehlszeile aufzurufen, müssen Sie den Pfad zur ausführbaren Datei kennen. Standardmäßig befindet sich die ausführbare Schema-Manager-Datei hier:

Windows	`C:\ProgramData\Altova\SharedBetweenVersions\XMLSchemaManager.exe`
Linux	`/opt/Altova/StyleVisionServer2025/bin/xmlschemamanager`
macOS	`/usr/local/Altova/StyleVisionServer2025/bin/xmlschemamanager`

Anmerkung: Nachdem Sie auf Linux- und macOS-Systemen das Verzeichnis in dasjenige, das die ausführbare Datei enthält, geändert haben, können Sie die ausführbare Datei mit **`sudo ./xmlschemamanager`** aufrufen. Das Präfix **`./`** gibt an, dass sich die ausführbare Datei im aktuellen Verzeichnis befindet. Das Präfix **`sudo`** gibt an, dass der Befehl mit Root-Rechten ausgeführt werden muss.

Befehlszeilensyntax

Die allgemeine Syntax zur Verwendung der Befehlszeile lautet folgendermaßen:

```
<exec> -h | --help | --version | <command> [options] [arguments]
```

Der senkrechte Balken **`|`** im Codefragment oben trennt eine Gruppe einander gegenseitig ausschließender Elemente. Optionale Elemente stehen innerhalb von eckigen Klammern **`[]`**. Im Prinzip können Sie den Pfad zur ausführbaren Datei, gefolgt von entweder `--h`, `--help` oder `--version`-Optionen oder gefolgt von einem Befehl eingeben. Jeder Befehl kann Optionen und Argumente haben. Die Liste der Befehle wird in den folgenden Abschnitten beschrieben.

6.5.1 help

Mit diesem Befehl erhalten Sie Hilfe zu Befehlen zur ausführbaren Schema-Manager-Datei.

Syntax

```
<exec> help [Befehl]
```

`[Befehl]` ist hierbei ein optionales Argument zur Angabe jedes beliebigen gültigen Befehlsnamens.

Beachten Sie dazu Folgendes:

- Sie können die Hilfe zu einem Befehl auch durch Eingabe des Befehls, gefolgt von **`-h`** oder **`--help`** aufrufen, z.B: **`<exec> list -h`**
- Wenn Sie **`-h`** oder **`--help`** direkt nach dem Namen der ausführbaren Datei und vor einem Befehl eingeben, wird die allgemeine Hilfe (und nicht die Hilfe zu einem bestimmten Befehl) angezeigt, z.B: **`<exec> -h list`**

Beispiel

Mit dem folgenden Befehl wird Hilfe zum Befehl `list` angezeigt:

```
xmlschemamanager help list
```

6.5.2 info

Mit diesem Befehl werden ausführliche Informationen über die einzelnen als `Schema`-Argument angegebenen Schemas angezeigt. Darin enthalten sind Titel, Version, Beschreibung, Herausgeber der jeweils angegebenen Schemas und davon referenzierte Schemas sowie die Information, ob das Schema installiert ist oder nicht.

Syntax

```
<exec> info [options] Schema+
```

- Das Argument **`Schema`** ist der Name eines Schemas oder Teil eines Schemanamens. (Die Paket-ID eines Schemas und detaillierte Informationen über ihren Installationsstatus erhalten Sie mit dem Befehl `list` 100.)
- Mit **`<exec> info -h`** können Sie die Hilfe zum Befehl anzeigen.

Beispiel

Mit dem folgenden Befehl werden Informationen über das jeweils neueste **`DocBook-DTD`**- und **`NITF`**-Schemas angezeigt.

```
xmlschemamanager info doc nitf
```

6.5.3 initialize

Mit diesem Befehl wird die Schema-Manager-Umgebung initialisiert. Sie erstellen damit ein Cache-Verzeichnis, in dem Informationen über alle Schemas lokal gespeichert werden. Die Initialisierung erfolgt automatisch bei der ersten Installation einer Schema-fähigen Altova-Applikation. Normalerweise muss dieser Befehl nicht ausgeführt werden. Nach Ausführung des **`reset`**-Befehls ist dies allerdings erforderlich.

Syntax

```
<exec> initialize | init [options]
```

Optionen

Für den Befehl **`initialize`** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **`false`**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **`false`**.

© 2019-2025 Altova GmbH

`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl wird der Schema-Manager initialisiert:

```
xmlschemamanager initialize
```

6.5.4 install

Mit diesem Befehl installieren Sie ein oder mehrere Schemas.

Syntax

```
<exec> install [options] Schema+
```

Um mehrere Schemas zu installieren, fügen Sie das Argument **Schema** mehrmals hinzu.

Als **Schema**-Argument kann eines der folgenden verwendet werden:

- Ein Schema-Identifier (im Format **<name>-<version>**, z.B: **cbcr-2.0**). Um die Schema-Identifier der gewünschten Schemas zu eruieren, führen Sie den Befehl `list` 100 aus. Sie können auch einen abgekürzten Identifier verwenden, sofern dieser eindeutig ist, z.B. **docbook**. Falls Sie einen abgekürzten Identifier verwenden, wird die neueste Version dieses Schemas installiert.
- Der Pfad zu einer von der Altova-Website heruntergeladenen **.altova_xmlschemas**-Datei. Informationen zu diesen Dateien finden Sie in der *Einführung zu Schema-Manager: Funktionsweise* 86.

Optionen
Für den Befehl **install** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl werden das CBCR 2.0 (Country-By-Country Reporting)-Schema und die neueste DocBook-DTD installiert:

```
xmlschemamanager install cbcr-2.0 docbook
```

6.5.5 list

Mit diesem Befehl werden vom Schema-Manager verwaltete Schemas aufgelistet. In der Liste wird eine der folgenden Informationen angezeigt:

- alle verfügbaren Schemas
- Schemas, die im Namen den im Argument **Schema** angegebenen String enthalten
- nur installierte Schemas
- Nur Schemas, für die ein Upgrade installiert werden kann

Syntax

```
<exec> list | ls [options] Schema?
```

Wenn kein **Schema**-Argument angegeben wird, werden alle verfügbaren Schemas aufgelistet. Andernfalls werden die durch die angegebenen Optionen definierten Schemas aufgelistet (*siehe Beispiel unten*). Beachten Sie, dass Sie das Argument **Schema** mehrfach angeben können.

Optionen
Für den Befehl **list** stehen die folgenden Optionen zur Verfügung:

`--installed, --i`	Auflisten nur der installierten Schemas. Der Standardwert ist **false**.
`--upgradeable, --u`	Auflisten nur derjenigen Schemas, für die Upgrades (Patches) zur Verfügung stehen. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiele

- Um alle verfügbaren Schemas aufzulisten, führen Sie den folgenden Befehl aus: **xmlschemamanager list**
- Um nur installierte Schemas aufzulisten, führen Sie **xmlschemamanager list -i** aus.
- Um Schemas, die in ihrem Namen entweder "doc" oder "nitf" enthalten, aufzulisten, führen Sie **xmlschemamanager list doc nitf** aus.

6.5.6 reset

Mit diesem Befehl werden alle installierten Schemas und das Cache-Verzeichnis entfernt. Ihre Schemaumgebung wird vollständig zurückgesetzt. Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize` [98] ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl **reset** mit der Option **-i** aus. Da mit **reset-i** die Originalinstallation des Produkts wiederhergestellt wird, wird empfohlen, nach dem Zurücksetzen und Initialisieren auch den Befehl `update` [102] auszuführen. Führen Sie alternativ dazu den Befehl **reset** mit den Optionen **-i** und **-u** aus.

© 2019-2025 Altova GmbH

Syntax

```
<exec> reset [Optionen]
```

Optionen
Für den Befehl **reset** stehen die folgenden Optionen zur Verfügung:

`--init, --i`	Initialisierung des Schema-Managers nach dem Zurücksetzen. Der Standardwert ist **false**.
`--update, --u`	Aktualisiert die Liste der verfügbaren Schemas im Cache. Der Standardwert ist **false**.
`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiele

- Um den Schema-Manager zurückzusetzen, führen Sie den folgenden Befehl aus: **xmlschemamanager reset**
- Um den Schema-Manager zurückzusetzen und ihn zu initialisieren, führen Sie **xmlschemamanager reset -i** aus.
- Um den Schema-Manager zurückzusetzen, ihn zu initialisieren und seine Schemaliste zu aktualisieren, führen Sie **xmlschemamanager reset -i -u** aus.

6.5.7 uninstall

Mit diesem Befehl deinstallieren Sie ein oder mehrere Schemas. Standardmäßig werden auch alle Schemas, die vom der aktuellen Schema referenziert werden, deinstalliert. Um nur das aktuelle Schema zu deinstallieren und die referenzierten Schemas beizubehalten, setzen Sie die Option `--k`.

Syntax

```
<exec> uninstall [options] Schema+
```

Um mehrere Schemas zu deinstallieren, fügen Sie das Argument **Schema** mehrmals hinzu.

Als **Schema**-Argument kann eines der folgenden verwendet werden:

- Ein Schema-Identifier (im Format **<name>-<version>**, z.B: **cbcr-2.0**). Um die Schema-Identifier der installierten Schemas zu eruieren, führen Sie den Befehl `list -i`[100] aus. Sie können auch einen abgekürzten Schemanamen verwenden, sofern dieser eindeutig ist, z.B. **docbook**. Falls Sie einen abgekürzten Namen verwenden, werden alle Schemas, die die Abkürzung in ihrem Namen enthalten, deinstalliert.

- Der Pfad zu einer von der Altova-Website heruntergeladenen **.altova_xmlschemas**-Datei. Informationen zu diesen Dateien finden Sie in der *Einführung zu Schema-Manager: Funktionsweise* 86.

Optionen
Für den Befehl **uninstall** stehen die folgenden Optionen zur Verfügung:

`--keep-references, --k`	Definieren Sie diese Option, um referenzierte Schemas beizubehalten. Der Standardwert ist **false**.
`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl werden die Schemas CBCR 2.0 und EPUB 2.0 und deren Abhängigkeiten deinstalliert:

```
xmlschemamanager uninstall cbcr-2.0 epub-2.0
```

Mit dem folgenden Befehl wird das **eba-2.10**-Schema, nicht aber die davon referenzierten Schemas deinstalliert:

```
xmlschemamanager uninstall --k cbcr-2.0
```

6.5.8 update

Mit diesem Befehl wird die Liste der über den Online-Speicher verfügbaren Schemas abgefragt und das lokale Cache-Verzeichnis wird aktualisiert. Normalerweise muss dieser Befehl nur ausgeführt werden, wenn Sie `reset` 100 und `initialize` 98 ausgeführt haben.

Syntax

```
<exec> update [options]
```

Optionen
Für den Befehl **update** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl wird der lokale Cache mit der Liste der neuesten Schemas aktualisiert:

© 2019-2025 Altova GmbH

```
xmlschemamanager update
```

6.5.9 upgrade

Mit diesem Befehl werden alle installierten Schemas, für die ein Upgrade installiert werden kann, auf die neueste verfügbare *Patch*-Version aktualisiert. Um herauszufinden, welche Schemas aktualisiert werden können, starten Sie den Befehl `list-u`[100].

Anmerkung: Der Befehl **`upgrade`** entfernt ein veraltetes Schema, falls keine neuere Version zur Verfügung steht.

Syntax

```
<exec> upgrade [Optionen]
```

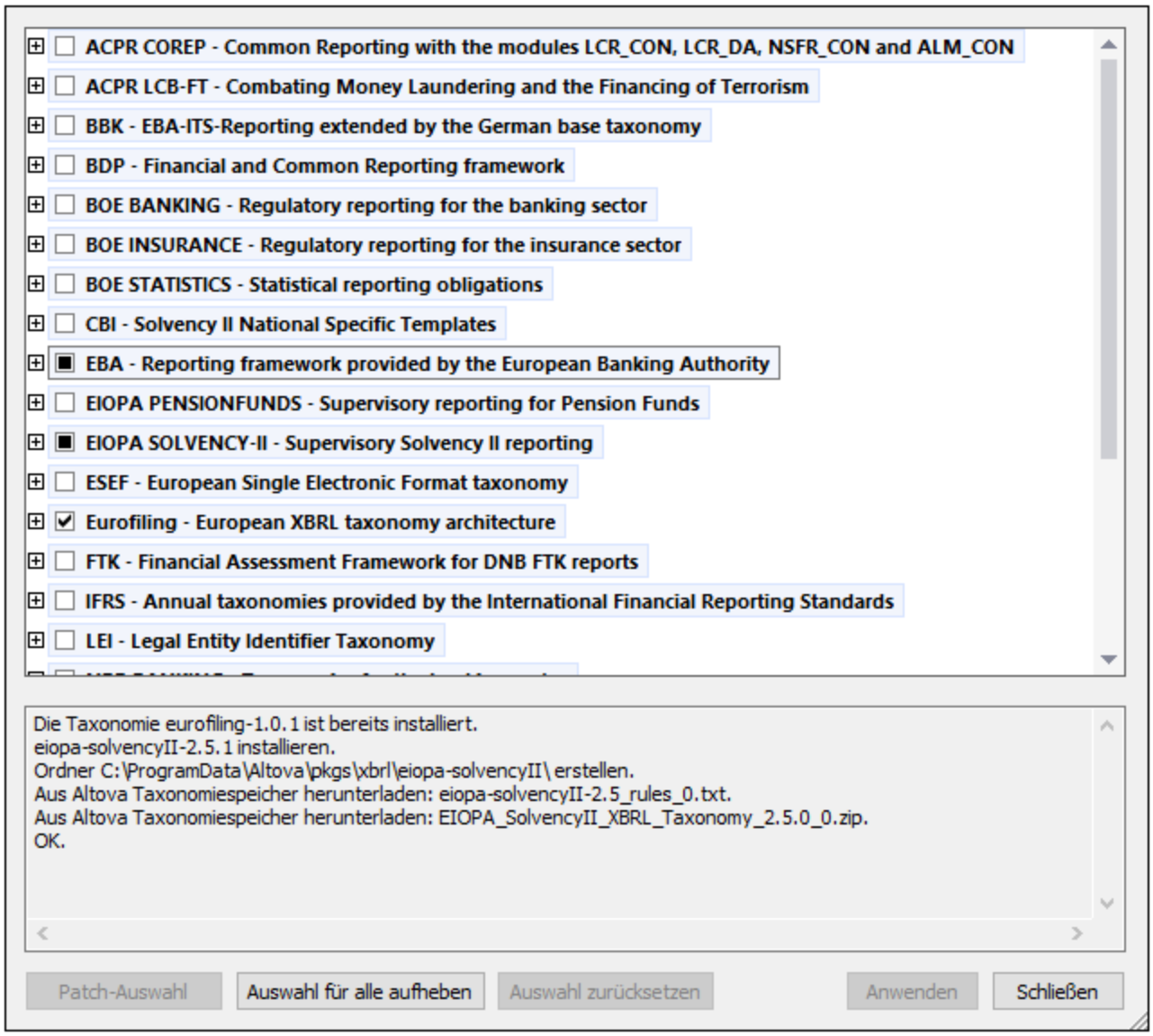
Optionen
Für den Befehl **`upgrade`** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **`false`**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **`false`**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

7 Taxonomie-Manager

Der XBRL-Taxonomie-Manager ist ein Altova-Tool, mit dem Sie XBRL-Taxonomien zentral installieren und verwalten können, um diese in allen XBRL-fähigen Applikationen von Altova einschließlich StyleVision Server verwenden zu können.

- Unter Windows hat der Taxonomie-Manager eine grafische Benutzeroberfläche (*siehe Abbildung unten*) und steht auch über die Befehlszeile zur Verfügung. (Die Desktop-Applikationen von Altova stehen nur unter Windows zur Verfügung; *siehe Liste unten*).
- Unter Linux und MacOS steht der Taxonomie-Manager nur über die Befehlszeile zur Verfügung. (Die Server-Applikationen von Altova stehen unter Windows, Linux und macOS zur Verfügung; *siehe Liste unten*).



 © 2019-2025 Altova GmbH

XBRL-fähige Applikationen von Altova

Desktop-Applikationen (nur Windows)	Server-Applikationen (Windows, Linux, macOS)
Altova XBRL Add-ins für Excel (EBA, ESEF, Solvency II, WIP)	MapForce Server (Standard und Advanced Edition)
MapForce Enterprise Edition	RaptorXML+XBRL Server
StyleVision Enterprise Edition	StyleVision Server
XMLSpy Enterprise Edition	

Installation und Deinstallation des Taxonomie-Managers

Der Taxonomie-Manager wird bei der ersten Installation einer neuen Version der Altova Mission Kit Enterprise Edition oder einer der XBRL-fähigen Applikationen von Altova (*siehe Tabelle oben*) automatisch installiert.

Ebenso wird er auch automatisch entfernt, wenn Sie die letzte XBRL-fähige Applikation von Altova auf Ihrem Rechner deinstallieren.

Taxonomie-Manager-Funktionalitäten

Im Taxonomie-Manager stehen die folgenden Funktionalitäten zur Verfügung:

- Anzeigen der auf Ihrem Rechner installierten XBRL-Taxonomien und Überprüfung, ob neue Versionen zum Download zur Verfügung stehen.
- Download neuer Versionen von XBRL-Taxonomien unabhängig vom Altova Produkt-Release-Zyklus. (Die Taxonomien werden von Altova online bereitgestellt und können über den Taxonomie-Manager heruntergeladen werden).
- Installation oder Deinstallation jeder beliebigen (oder ggf. aller) der zahlreichen Versionen einer bestimmten Taxonomie.
- Eine XBRL-Taxonomie kann Abhängigkeiten von anderen Taxonomien aufweisen. Bei der Installation oder Deinstallation einer bestimmten Taxonomie informiert Sie der Taxonomie-Manager über davon abhängige Taxonomien und installiert bzw. entfernt diese ebenfalls automatisch.
- Der Taxonomie-Manager ordnet Schema-Referenzen mit Hilfe des XML-Katalogs lokalen Dateien zu. Dadurch lassen sich große XBRL-Taxonomien schneller verarbeiten, als wenn sie sich unter einem entfernten Pfad befinden.
- Alle wichtigen Taxonomien werden über den Taxonomie-Manager bereitgestellt und regelmäßig auf die jeweils neuesten Version aktualisiert. Dadurch können alle Ihre Taxonomien zentral verwaltet werden und stehen allen XBRL-fähigen Applikationen von Altova jederzeit zur Verfügung.
- Im Taxonomie-Manager vorgenommene Änderungen werden für alle auf dem Rechner installierten Altova-Produkte wirksam.

Benutzerdefinierte XBRL-Taxonomien

Wenn Sie mit benutzerdefinierten XBRL-Taxonomien, die nicht Teil des Taxonomie-Manager sind, arbeiten müssen, können Sie diese zur Gruppe der benutzerdefinierten Pakete, die StyleVision Server referenzieren kann, hinzufügen. Gehen Sie dazu folgendermaßen vor:

- *In Altova-Desktop-Applikationen:* Wählen Sie den Menübefehl **Extras | Optionen** und gehen Sie anschließend zum Abschnitt *XBRL | Taxonomiepakete*. Navigieren Sie zum ZIP-Paket Ihrer

benutzerdefinierten XBRL-Taxonomie. Nähere Informationen dazu finden Sie in Ihrer Desktop-Produktdokumentation unter der Beschreibung dieses Befehls.

- *In Altova-Server-Applikationen:* Geben Sie bei der Befehlszeilenausführung von Befehlen, die benutzerdefinierte Taxonomien unterstützen, die Option **`--taxonomy-package`** oder **`--taxonomy-package-config-file`** ein. Diese Optionen werden z.B. in RaptorXML+XBRL Server von XBRL-Validierungsbefehlen wie valxbrl oder valxbrltaxonomy oder in MapForce vom Befehl run unterstützt.

Funktionsweise

Alle in Altova-Produkten verwendeten XBRL-Taxonomien werden von Altova online bereitgestellt. Dieser Speicher wird bei Veröffentlichung neuer Versionen der Taxonomien aktualisiert. Im Taxonomie-Manager werden sowohl bei Aufruf über die Benutzeroberfläche als auch über das CLI Informationen über die neuesten verfügbare Taxonomien angezeigt. Sie können die gewünschten Taxonomien dann über den Taxonomie-Manager installieren, aktualisieren oder deinstallieren.

Taxonomien können vom Taxonomie-Manager auch auf eine weitere Art installiert werden. Sie können eine Taxonomie und die davon abhängigen Taxonomien auf der Altova Website (https://www.altova.com/de/taxonomy-manager) auswählen. Daraufhin wird auf der Website eine Datei des Typs **`.altova_taxonomies`** mit Informationen über Ihre ausgewählte Taxonomie zum Download vorbereitet. Bei Doppelklick auf diese Datei oder bei Übergabe an den Taxonomie-Manager über das CLI als Argument des Befehls **`install`**[118] installiert der Taxonomie-Manager die ausgewählten Taxonomien.

Lokaler Cache: Überprüfung Ihrer Taxonomien

Alle Informationen über installierte Taxonomien werden in einem zentralen Cache-Verzeichnis auf Ihrem Rechner aufgezeichnet. Das Verzeichnis befindet sich hier:

Windows	`C:\ProgramData\Altova\pkgs\.cache`
Linux	`/var/opt/Altova/pkgs\.cache`
macOS	`/var/Altova/pkgs`

Dieses Cache-Verzeichnis wird regelmäßig mit dem neuesten Status der Taxonomien im Online-Speicher von Altova aktualisiert. Diese Aktualisierungen finden unter den folgenden Bedingungen statt:

- bei jedem Start von Taxonomie-Manager.
- Wenn Sie StyleVision Server zum ersten Mal an einem bestimmten Kalendertag starten.
- Wenn StyleVision Server länger als 24 Stunden geöffnet ist, findet alle 24 Stunden eine Aktualisierung des Cache statt.
- Sie können den Cache auch durch Ausführung des `update`[121]-Befehls über die Befehlszeilenschnittstelle aktualisieren.

Der Taxonomie-Manager kann somit Ihre installierten Taxonomien über den Cache ständig anhand der online verfügbaren Taxonomien auf der Altova Website überprüfen.

Nehmen Sie keine manuellen Änderungen am Cache vor!

Das lokale Cache-Verzeichnis wird automatisch auf Basis der installierten oder deinstallierten Taxonomien verwaltet; es sollte nicht manuell geändert oder gelöscht werden. Falls Sie den Taxonomie-Manager je in seinen Originalzustand zurücksetzen möchten, (i) führen Sie den CLI-Befehl `reset`[119] der Befehlszeilenschnittstelle und (ii) anschließend den Befehl `initialize`[117] aus. (Führen Sie alternativ dazu den Befehl **`reset`** mit der Option **`-i`** aus).

© 2019-2025 Altova GmbH

HTTP-Proxy

Sie können für Taxonomie-Manager-Verbindungen einen HTTP-Proxy verwenden. Es werden die Proxy-Einstellungen des Systems verwendet.

7.1 Ausführen des Taxonomie-Managers

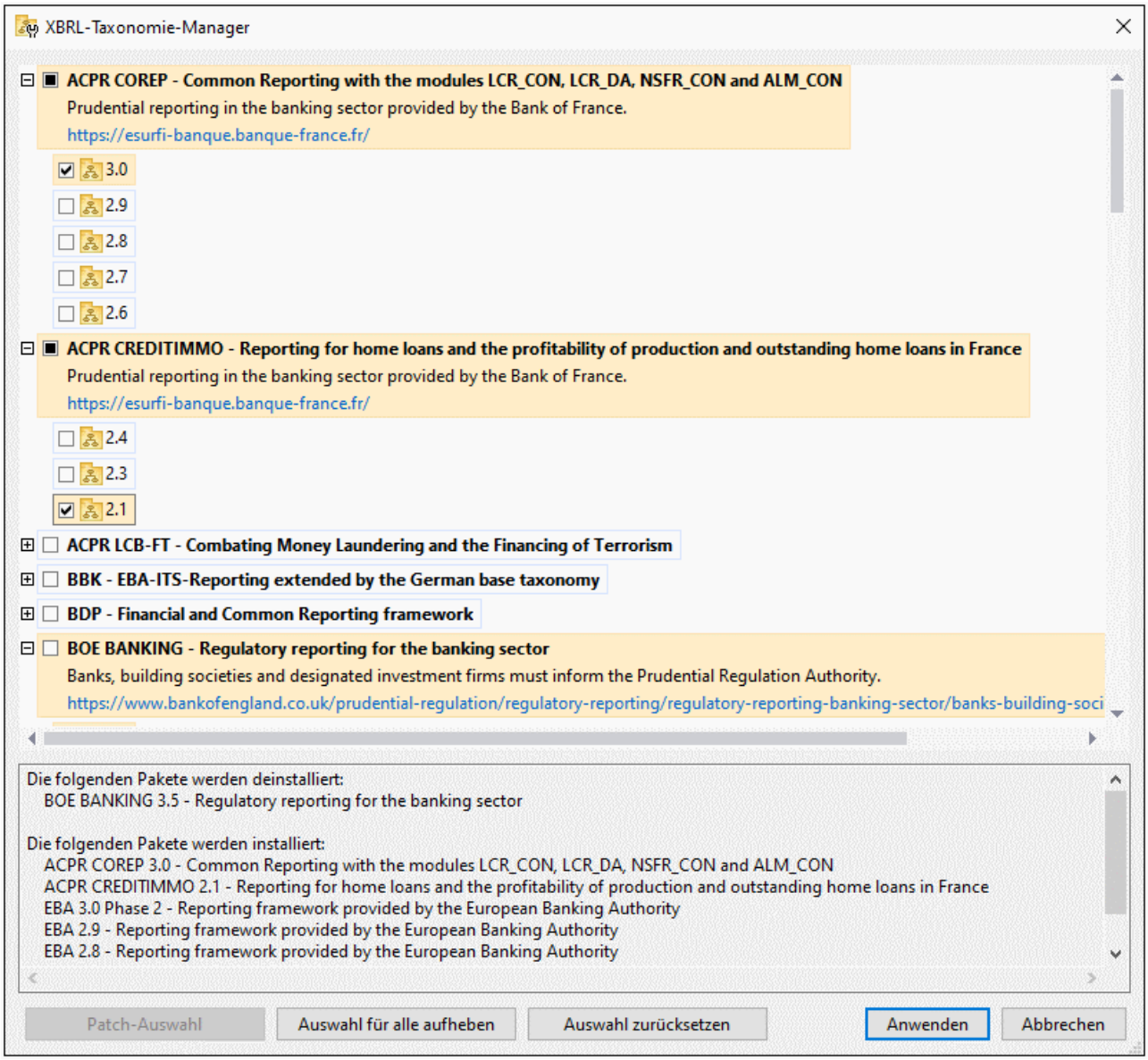
Grafische Benutzeroberfläche

Sie können die Benutzeroberfläche des Taxonomie-Managers auf eine der folgenden Arten aufrufen:

- *Bei der Installationen von StyleVision Server:* Aktivieren Sie gegen Ende der Installation das Kontrollkästchen *Altova-Taxonomie-Manager aufrufen*, wodurch Sie die Benutzeroberfläche des XBRL-Taxonomie-Managers direkt aufrufen können. Auf diese Art können Sie Taxonomien während der Installation Ihrer Altova-Applikation installieren.
- über die vom XBRL-Taxonomie-Download-Center von Altova heruntergeladene **`.altova_taxonomies`**-Datei: Doppelklicken Sie auf die heruntergeladene Datei, um den Taxonomie-Manager zu starten, der daraufhin die (auf der Website) ausgewählten Taxonomien installiert.

Nachdem die Benutzeroberfläche des Taxonomie-Managers geöffnet wurde (*Abbildung unten*), werden bereits installierte Taxonomien markiert angezeigt. Wenn eine zusätzliche Taxonomie installiert werden soll, aktivieren Sie diese. Wenn eine bereits installierte Taxonomie deinstalliert werden soll, deaktivieren Sie diese. Nachdem Sie Ihre Auswahl getroffen haben, können Ihre Änderungen angewendet werden. Die Taxonomien, die installiert bzw. deinstalliert werden, werden markiert und im Fenster "Meldungen" am unteren Rand des Taxonomie-Manager-Fensters (*siehe Abbildung*) erscheint eine Meldung über die bevorstehenden Änderungen.

© 2019-2025 Altova GmbH



Wenn Sie auf **Anwenden** klicken, wird der Fortschritt der Installation in einem vereinfachten Taxonomie-Manager-Dialogfeld angezeigt. Bei Auftreten eines Fehlers (z.B. eines Verbindungsfehlers) wird im Dialogfeld eine Fehlermeldung angezeigt. Klicken Sie in diesem Fall auf die Schaltfläche **Erweitert**, überprüfen Sie die Taxonomieauswahl und versuchen Sie es erneut durch Klicken auf **Anwenden**.

Befehlszeilenschnittstelle

Sie können den Taxonomie-Manager über eine Befehlszeilenschnittstelle starten, indem Sie Befehle an die ausführbare Datei **`taxonomymanager.exe`** senden.

Die Datei **`taxonomymanager.exe`** steht im folgenden Ordner zur Verfügung:

- *Unter Windows:* `C:\ProgramData\Altova\SharedBetweenVersions`

- *Unter Linux oder macOS (nur Server-Applikationen):* `%INSTALLDIR%/bin`, wobei `%INSTALLDIR%` das Installationsverzeichnis des Programms ist.

Anschließend können Sie jeden der im Abschnitt zur CLI-Befehlsreferenz aufgelisteten Befehle verwenden.

Um die Hilfe zu den Befehlen anzuzeigen, führen Sie den folgenden Befehl aus:

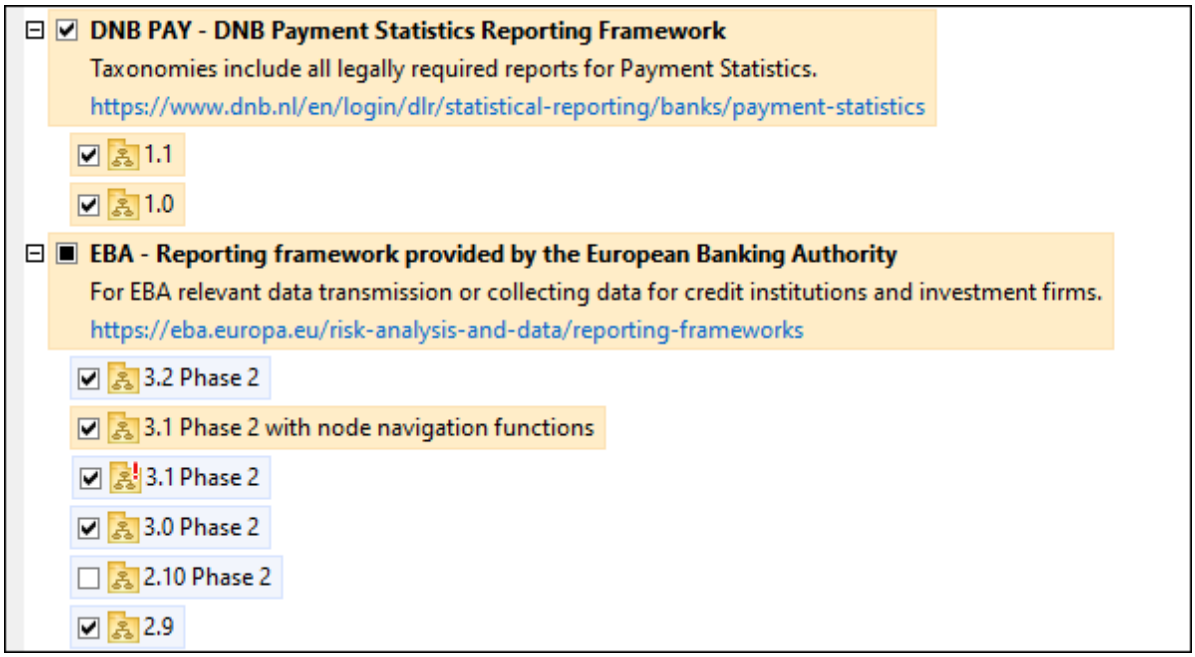
- *Unter Windows:* `taxonomymanager.exe --help`
- *Unter Linux oder macOS (nur Server-Applikationen):* `sudo ./taxonomymanager --help`

© 2019-2025 Altova GmbH

7.2 Statuskategorien

Der Taxonomie-Manager unterscheidet folgendermaßen zwischen den von ihm verwalteten Taxonomien:

- *Installierte Taxonomien* Diese werden auf der Benutzeroberfläche mit einem Häkchen angezeigt (*in der Abbildung unten sind die mit einem Häkchen versehenen Versionen der DNB- und EBA-Taxonomie installiert*). Wenn alle Versionen einer Taxonomie ausgewählt sind, wird ein Häkchen angezeigt. Wenn zumindest eine Version nicht ausgewählt ist, wird die Auswahl in Form eines gefüllten Quadrats angezeigt (siehe EBA in der Abbildung unten). Sie können die Auswahl für eine installierte Taxonomie aufheben, um sie zu **deinstallieren**.
- *Nicht installierte verfügbare Taxonomien.* Diese werden auf der Benutzeroberfläche mit einem deaktivierten Kontrollkästchen angezeigt. Sie können die Taxonomien, die **installiert** werden sollen, auswählen.



- *Taxonomien, für die ein Upgrade zur Verfügung steht* sind diejenigen, die seit ihrer Installation vom Herausgeber überarbeitet wurden. Sie werden auf der Benutzeroberfläche durch ein Symbol gekennzeichnet (*siehe Abbildung oben*). Sie können für eine installierte Taxonomie ein **Patch** der überarbeiteten Version installieren.

Wichtige Punkte

- In der Abbildung oben sind beide DNB-Taxonomien und einige der EBA-Taxonomien ausgewählt. Diejenigen mit einem blauen Hintergrund sind bereits installiert. Diejenige mit gelbem Hintergrund sind nicht installiert und wurden für die Installation ausgewählt. Beachten Sie, dass (i) die EBA 2.10 Phase 2-Taxonomie nicht installiert wurde und nicht für die Installation ausgewählt wurde, (ii) die EBA 3.1 Phase 2-Taxonomie installiert wurde, dafür aber seit ihrer Installation ein Patch vom Herausgeber zur Verfügung gestellt wurde, das noch nicht installiert wurde.
- Bei Ausführung des Taxonomie-Managers über die Befehlszeile wird der Befehl `list` [119] mit verschiedenen Optionen verwendet, um verschiedene Taxonomiekategorien aufzulisten:

taxonomymanager.exe list	Listet alle installierten und verfügbaren Taxonomien auf; auch verfügbare Upgrades werden angezeigt.
taxonomymanager.exe list -i	Listet nur installierte Taxonomien auf; auch verfügbare Upgrades werden angezeigt.
taxonomymanager.exe list -u	Listet für Taxonomien auf, für die Upgrades zur Verfügung stehen

Anmerkung: Verwenden Sie unter Linux und macOS, **sudo ./taxonomymanager list**

© 2019-2025 Altova GmbH

7.3 Anwenden eines Patch oder Installation einer Taxonomie

Anwenden eines Patch auf eine installierte Taxonomie

Von Zeit zu Zeit werden von den Herausgebern der XBRL-Taxonomien Patches (Upgrades oder Überarbeitungen) veröffentlicht. Wenn der Taxonomie-Manager erkennt, dass Patches zur Verfügung stehen, werden diese in der Taxonomieliste des Taxonomie-Managers angezeigt und Sie können diese Patches schnell installieren.

Über die Benutzeroberfläche

Patches werden mit dem Symbol gekennzeichnet. (*Siehe auch vorhergehendes Kapitel über Statuskategorien*[111]*).* Falls Patches zur Verfügung stehen, ist die Schaltfläche **Patch-Auswahl** aktiv. Klicken Sie darauf, um alle Patches für die Installation auszuwählen und vorzubereiten. Auf der Benutzeroberfläche ändert sich das Symbol von Taxonomien, für die ein Patch installiert wird von in , und im Fenster "Meldungen" am unteren Rand des Dialogfelds werden die Patches, die angewendet werden, aufgelistet. Sobald Sie mit der Auswahl fertig sind, klicken Sie auf **Anwenden**. Alle Patches werden gemeinsam angewendet. Beachten Sie, dass eine für die Installation eines Patch markierte Taxonomie deinstalliert wird, wenn Sie die Auswahl für diese Taxonomie aufheben.

Über das CLI

So wenden Sie einen Patch über die Befehlszeilenschnittstelle an:

1. Führen Sie den Befehl `list -u`[119] aus. Daraufhin werden alle Taxonomien, für die Patch Upgrades zur Verfügung stehen, aufgelistet.
2. Führen Sie den Befehl `upgrade`[122] aus, um alle Patches zu installieren.

Installieren einer verfügbaren Taxonomie

Sie können Taxonomien entweder über die Benutzeroberfläche des Taxonomie-Managers oder durch Senden der Taxonomie-Manager-Installationsbefehle über die Befehlszeile installieren.

Anmerkung: Wenn die aktuelle Taxonomie andere Taxonomien referenziert, werden auch die referenzierten Taxonomien installiert.

Über die Benutzeroberfläche

Um Taxonomien über die Benutzeroberfläche des Taxonomie-Managers zu installieren, wählen Sie die gewünschten Taxonomien aus und klicken Sie auf **Anwenden**.

Sie können die gewünschten Taxonomien auch auf der Altova Website auswählen und eine herunterladbare `.altova_taxonomies`-Datei generieren. Bei Doppelklick auf diese Datei wird der Taxonomie-Manager aufgerufen, in dem die gewünschten Taxonomien bereits vorausgewählt sind. Sie müssen nur mehr auf **Anwenden** klicken.

Über das CLI

Um Taxonomien über die Befehlszeile zu installieren, rufen Sie den Befehl `install`[118] auf:

```
taxonomymanager.exe install [options] Taxonomy+
```

wobei es sich bei **Taxonomy** um die gewünschte(n) Taxonomie(n) bzw. eine **.altova_taxonomies**-Datei handelt. Eine Taxonomie wird von einem Identifier im Format **<name>-<version>** referenziert. (Die Identifier von Taxonomien werden angezeigt, wenn Sie den Befehl **list**[119] ausführen.) Sie können beliebig viele Taxonomien eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls install[118].

Anmerkung: Verwenden Sie unter Linux oder macOS den Befehl **sudo ./taxonomymanager**.

Installation einer benötigten Taxonomie

Wenn Sie einen XBRL-Befehl in StyleVision Server ausführen und StyleVision Server erkennt, dass eine zur Ausführung des Befehls erforderliche Taxonomie nicht vorhanden oder unvollständig ist, wird der Taxonomie-Manager mit Informationen über die fehlende Taxonomie aufgerufen. Sie können die gewünschten Taxonomien dann über den Taxonomie-Manager direkt installieren.

Alle bereits installierten Taxonomien können jederzeit durch Aufruf des Taxonomie-Managers über **Extras | Taxonomie-Manager** über die Benutzeroberfläche des Taxonomie-Managers angezeigt werden.

 © 2019-2025 Altova GmbH

7.4 Deinstallieren einer Taxonomie, Zurücksetzen

Deinstallieren einer Taxonomie

Sie können Taxonomien entweder über die Benutzeroberfläche des Taxonomie-Managers oder durch Senden der Taxonomie-Manager-Deinstallationsanweisungen über die Befehlszeile deinstallieren.

Anmerkung: Wenn die gewünschte Taxonomie andere Taxonomien referenziert, so werden auch die referenzierten Taxonomien deinstalliert.

Über die Benutzeroberfläche

Um Taxonomien über die Benutzeroberfläche des Taxonomie-Managers zu deinstallieren, deaktivieren Sie die Kontrollkästchen der entsprechenden Taxonomien und klicken Sie auf **Anwenden**. Daraufhin werden die ausgewählten Taxonomien und die davon referenzierten Taxonomien deinstalliert.

Um alle Taxonomien zu deinstallieren, klicken Sie auf **Auswahl für alle aufheben** und anschließend auf **Anwenden**.

Über das CLI

Um Taxonomien über die Befehlszeile zu deinstallieren, rufen Sie den Befehl **`uninstall`** auf:

```
taxonomymanager.exe uninstall [options] Taxonomy+
```

wobei es sich beim Argument **`Taxonomy`** eine zu deinstallierende Taxonomie oder eine **`.altova_taxonomies`**-Datei handelt. Eine Taxonomie wird von einem Identifier im Format **`<name>-<version>`** definiert. (Die Identifier von Taxonomien werden angezeigt, wenn Sie den Befehl **`list`**[119] ausführen.) Sie können beliebig viele Taxonomien eingeben. Nähere Informationen dazu finden Sie unter der Beschreibung des Befehls `uninstall`[120].

Anmerkung: Verwenden Sie unter Linux oder macOS den Befehl **`sudo ./taxonomymanager`**.

Zurücksetzen des Taxonomie-Managers

Sie können den Taxonomie-Manager zurücksetzen.

- Klicken Sie auf der Benutzeroberfläche auf **Auswahl zurücksetzen**. Damit wird die Benutzeroberfläche zurückgesetzt, und Sie sehen, welche Taxonomien derzeit installiert sind. Jegliche Auswahl bzw. Aufhebung einer Auswahl, die der Benutzer in der aktuellen Sitzung vorgenommen hat, wird aufgehoben.
- Führen Sie über die Benutzeroberfläche den Befehl `reset`[119] aus. Damit werden alle installierten Taxonomien und das Cache-Verzeichnis entfernt.

Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize`[117] ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl `reset`[119] mit der Option `-i` aus.

Beachten Sie, dass mit `reset-i`[119] die Originalinstallation des Produkts wiederhergestellt wird, daher wird empfohlen, nach dem Zurücksetzen auch den Befehl `update`[121] auszuführen. Führen Sie alternativ dazu den Befehl `reset`[119] mit den Optionen `-i` und `-u` aus.

7.5 Befehlszeilenschnittstelle (CLI)

Um den Taxonomie-Manager über die Befehlszeile aufzurufen, müssen Sie den Pfad zur ausführbaren Datei kennen. Standardmäßig befindet sich die ausführbare Taxonomie-Manager-Datei hier:

Windows	`C:\ProgramData\Altova\SharedBetweenVersions\TaxonomyManager.exe`
Linux	`/opt/Altova/StyleVisionServer2025/bin/taxonomymanager`
macOS	`/usr/local/Altova/StyleVisionServer2025/bin/taxonomymanager`

Anmerkung: Nachdem Sie auf Linux- und macOS-Systemen das Verzeichnis in dasjenige, das die ausführbare Datei enthält, geändert haben, können Sie die ausführbare Datei mit **`sudo ./taxonomymanager`** aufrufen. Das Präfix **`./`** gibt an, dass sich die ausführbare Datei im aktuellen Verzeichnis befindet. Das Präfix **`sudo`** gibt an, dass der Befehl mit Root-Rechten ausgeführt werden muss.

Befehlszeilensyntax

Die allgemeine Syntax zur Verwendung der Befehlszeile lautet folgendermaßen:

```
<exec> -h | --help | --version | <command> [options] [arguments]
```

Der senkrechte Balken **`|`** im Codefragment oben trennt eine Gruppe einander gegenseitig ausschließender Elemente. Optionale Elemente stehen innerhalb von eckigen Klammern **`[]`**. Im Prinzip können Sie den Pfad zur ausführbaren Datei, gefolgt von entweder `--h`, `--help` oder `--version`-Optionen oder gefolgt von einem Befehl eingeben. Jeder Befehl kann Optionen und Argumente haben. Die Liste der Befehle wird in den folgenden Abschnitten beschrieben.

7.5.1 help

Mit diesem Befehl erhalten Sie Hilfe zu Befehlen zur ausführbaren Taxonomie-Manager-Datei.

Syntax

```
<exec> help [Befehl]
```

`[Befehl]` ist hierbei ein optionales Argument zur Angabe jedes beliebigen gültigen Befehlsnamens.

Beachten Sie dazu Folgendes:

- Sie können die Hilfe zu einem Befehl auch durch Eingabe des Befehls, gefolgt von **`-h`** oder **`--help`** aufrufen, z.B: **`<exec> list -h`**
- Wenn Sie **`-h`** oder **`--help`** direkt nach dem Namen der ausführbaren Datei und vor einem Befehl eingeben, wird die allgemeine Hilfe (und nicht die Hilfe zu einem bestimmten Befehl) angezeigt, z.B: **`<exec> -h list`**

© 2019-2025 Altova GmbH

Beispiel

Mit dem folgenden Befehl wird Hilfe zum Befehl `list` angezeigt:

```
taxonomymanager help list
```

7.5.2 info

Mit diesem Befehl werden ausführliche Informationen über die einzelnen als `Taxonomy`-Argument angegebenen Taxonomien angezeigt. Darin enthalten sind Titel, Version, Beschreibung, Herausgeber der jeweils angegebenen Taxonomie und davon abhängige Taxonomien sowie die Information, ob die Taxonomie installiert ist oder nicht.

Syntax

```
<exec> info [options] Taxonomy+
```

- Das Argument **`Taxonomy`** ist der Name einer Taxonomie oder Teil eines Taxonomienamens. (Die Paket-ID einer Taxonomie und detaillierte Informationen über ihren Installationsstatus erhalten Sie mit dem Befehl `list`[119].)
- Mit **`<exec> info -h`** können Sie die Hilfe zum Befehl anzeigen.

Beispiel

Mit dem folgenden Befehl werden Informationen über die Taxonomien **`eba-2.10`** und **`us-gaap-2020.0`** angezeigt:

```
taxonomymanager info eba-2.10 us-gaap-2020.0
```

7.5.3 initialize

Mit diesem Befehl wird die Taxonomie-Manager-Umgebung initialisiert. Sie erstellen damit ein Cache-Verzeichnis, in dem Informationen über alle Taxonomien lokal gespeichert werden. Die Initialisierung erfolgt automatisch bei der ersten Installation einer XBRL-fähigen Altova-Applikation. Normalerweise muss dieser Befehl nicht ausgeführt werden. Nach Ausführung des **`reset`**-Befehls ist dies allerdings erforderlich.

Syntax

```
<exec> initialize | init [options]
```

Optionen

Für den Befehl **`initialize`** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **`false`**.

`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **`false`**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl wird der Taxonomie-Manager initialisiert:

```
taxonomymanager initialize
```

7.5.4 install

Mit diesem Befehl installieren Sie eine oder mehrere Taxonomien.

Syntax

```
<exec> install [options] Taxonomy+
```

Um mehrere Taxonomien zu installieren, fügen Sie das Argument **`Taxonomy`** mehrmals hinzu.

Als **`Taxonomy`**-Argument kann eines der folgenden verwendet werden:

- Ein Taxonomie-Identifier (im Format **`<name>-<version>`**, z.B: **`eba-2.10`**). Um die Taxonomie-Identifier der gewünschten Taxonomien zu eruieren, führen Sie den Befehl `list` 119 aus. Sie können auch einen abgekürzten Identifier verwenden, sofern dieser eindeutig ist, z.B. **`eba`**. Falls Sie einen abgekürzten Identifier verwenden, wird die neueste Version dieser Taxonomie installiert.
- Der Pfad zu einer von der Altova-Website heruntergeladenen **`.altova_taxonomies`**-Datei. Informationen zu diesen Dateien finden Sie in *der Einführung zum Taxonomie-Manager.: Funktionsweise* 104.

Optionen

Für den Befehl **`install`** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **`false`**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **`false`**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl werden die neuesten **`eba`** (European Banking Authority) und **`us-gaap`** (US Generally Accepted Accounting Principles) Taxonomien installiert:

```
taxonomymanager install eba us-gaap
```

 © 2019-2025 Altova GmbH

7.5.5 list

Mit diesem Befehl werden vom Taxonomie-Manager verwaltete Taxonomien aufgelistet. In der Liste wird eine der folgenden Informationen angezeigt:

- Alle verfügbaren Taxonomien
- Taxonomien, die im Namen den im Argument **Taxonomy** angegebenen String enthalten
- Nur installierte Taxonomien
- Nur Taxonomien, für die ein Upgrade installiert werden kann

Syntax

```
<exec> list | ls [options] Taxonomy?
```

Wenn kein **Taxonomy**-Argument angegeben wird, werden alle verfügbaren Taxonomien aufgelistet. Andernfalls werden die durch die angegebenen Optionen definierten Taxonomien aufgelistet (*siehe Beispiel unten*). Beachten Sie, dass Sie das Argument **Taxonomy** mehrfach angeben können.

Optionen
Für den Befehl **list** stehen die folgenden Optionen zur Verfügung:

`--installed, --i`	Auflisten nur der installierten Taxonomien. Der Standardwert ist **false**.
`--upgradeable, --u`	Auflisten nur derjenigen Taxonomien, für die Upgrades (Patches) zur Verfügung stehen. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiele

- Um alle verfügbaren Taxonomien aufzulisten, führen Sie den folgenden Befehl aus: **taxonomymanager list**
- Um installierte Taxonomien aufzulisten, führen Sie **taxonomymanager list -i** aus.
- Um Taxonomien, die in ihrem Namen entweder "eba" oder "us-gaap" enthalten, aufzulisten, führen Sie **taxonomymanager list eba us-gaap** aus.

7.5.6 reset

Mit diesem Befehl werden alle installierten Taxonomien und das Cache-Verzeichnis entfernt. Ihre Taxonomieumgebung wird vollständig zurückgesetzt. Nachdem Sie diesen Befehl ausgeführt haben, muss der Befehl `initialize` 117 ausgeführt werden, um das Cache-Verzeichnis neu zu erstellen. Führen Sie alternativ dazu den Befehl `reset` mit der Option `-i` aus. Da mit `reset-i` die Originalinstallation des Produkts wiederhergestellt wird, wird empfohlen, nach dem Zurücksetzen und Initialisieren auch den Befehl `update` 121 auszuführen. Führen Sie alternativ dazu den Befehl reset mit den Optionen `-i` und `-u` aus.

Syntax

```
<exec> reset [Optionen]
```

Optionen
Für den Befehl **reset** stehen die folgenden Optionen zur Verfügung:

`--init, --i`	Initialisierung des Taxonomie-Managers nach dem Zurücksetzen. Der Standardwert ist **false**.
`--update, --u`	Aktualisiert die Liste der verfügbaren Taxonomien im Cache. Der Standardwert ist **false**.
`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiele

- Um den Taxonomie-Manager zurückzusetzen, führen Sie den folgenden Befehl aus: **taxonomymanager reset**
- Um den Taxonomie-Manager zurückzusetzen und ihn zu initialisieren, führen Sie **taxonomymanager reset -i** aus.
- Um den Taxonomie-Manager zurückzusetzen, ihn zu initialisieren und seine Taxonomieliste zu aktualisieren, führen Sie **taxonomymanager reset -i -u** aus.

7.5.7 uninstall

Mit diesem Befehl deinstallieren Sie eine oder mehrere Taxonomien. Standardmäßig werden auch alle Taxonomien, die von der aktuellen Taxonomie referenziert werden, deinstalliert. Um nur die aktuelle Taxonomie zu deinstallieren und die referenzierten Taxonomien beizubehalten, setzen Sie die Option `--k`.

Syntax

```
<exec> uninstall [options] Taxonomy+
```

Um mehrere Taxonomien zu deinstallieren, fügen Sie das Argument **Taxonomy** mehrmals hinzu.

Als **Taxonomy**-Argument kann eines der folgenden verwendet werden:

- Ein Taxonomie-Identifier (im Format **<name>-<version>**, z.B: **eba-2.10**). Um die Taxonomie-Identifier der installierten Taxonomien zu eruieren, führen Sie den Befehl `list -i` 119 aus. Sie können auch einen abgekürzten Taxonomienamen verwenden, sofern dieser eindeutig ist, z.B. **eba**. Falls Sie einen abgekürzten Namen verwenden, werden alle Taxonomien, die die Abkürzung in ihrem Namen enthalten, deinstalliert.

© 2019-2025 Altova GmbH

- Der Pfad zu einer von der Altova-Website heruntergeladenen **.altova_taxonomies**-Datei. Informationen zu diesen Dateien finden Sie *der Einführung zum Taxonomie-Manager.: Funktionsweise*[104].

Optionen
Für den Befehl **uninstall** stehen die folgenden Optionen zur Verfügung:

`--keep-references, --k`	Definieren Sie diese Option, um referenzierte Taxonomien beizubehalten. Der Standardwert ist **false**.
`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl werden die Taxonomien **eba-2.10** und **us-gaap-2020.0** und deren Abhängigkeiten deinstalliert:

```
taxonomymanager uninstall eba-2.10 us-gaap-2020.0
```

Mit dem folgenden Befehl wird die **eba-2.10**-Taxonomie, nicht aber die davon referenzierten Taxonomien deinstalliert:

```
taxonomymanager uninstall --k eba-2.10
```

7.5.8 update

Mit diesem Befehl wird die Liste der über den Online-Speicher verfügbaren Taxonomien abgefragt und das lokale Cache-Verzeichnis wird aktualisiert. Normalerweise muss dieser Befehl nur ausgeführt werden, wenn Sie `reset`[119] und `initialize`[117] ausgeführt haben.

Syntax

```
<exec> update [options]
```

Optionen
Für den Befehl **update** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

Beispiel

Mit dem folgenden Befehl wird der lokale Cache mit der Liste der neuesten Taxonomien aktualisiert:

```
taxonomymanager update
```

7.5.9 upgrade

Mit diesem Befehl werden alle installierten Taxonomien, für die ein Upgrade installiert werden kann, auf die neueste verfügbare *Patch*-Version aktualisiert. Um herauszufinden, welche Taxonomien aktualisiert werden können, starten Sie den Befehl `list -u` [119] .

Anmerkung: Der Befehl **upgrade** würde eine veraltete Taxonomie entfernen, falls keine neuere Version zur Verfügung steht.

Syntax

```
<exec> upgrade [Optionen]
```

Optionen
Für den Befehl **upgrade** stehen die folgenden Optionen zur Verfügung:

`--silent, --s`	Nur Fehlermeldungen anzeigen. Der Standardwert ist **false**.
`--verbose, --v`	Anzeige detaillierter Informationen während der Ausführung. Der Standardwert ist **false**.
`--help, --h`	Anzeige der Hilfe zum Befehl.

© 2019-2025 Altova GmbH

Index

A

Altova ServiceController, 16

D

Datenbankverbindungen auf macOS, 31
Datenbankverbindungen unter Linux, 25
Deinstallation, 12
Deinstallieren, 12

E

Einrichten,
auf macOS, 27
unter Linux, 20
unter Windows, 12
Einrichten von StyleVision Server, 11

F

FlowForce Server,
und StyleVision Server, 9
FOP-Anforderungen, 35

I

Installation auf macOS, 27
Installation auf Windows Server Core, 13
Installation unter Linux, 21
Installation unter Windows, 12
Installation von LicenseServer auf macOS, 29
Installation von LicenseServer unter Linux, 22
Installieren von LicenseServer unter Windows, 15
Installieren von StyleVision Server, 11

J

JRE, 35

L

LicenseServer-Versionen, 15, 22, 29
Linux,
Installation, 21
Lizenzieren von StyleVision Server, 11
Lizenz auf macOS zuweisen, 30
Lizenz unter Linux zuweisen, 24
unter Windows eine Lizenz zuweisen, 18

M

macOS,
Installation, 27
Migrieren von StyleVision Server auf einen neuen Rechner, 34

O

OpenJDK, 35

R

Registrieren von StyleVision Server auf LicenseServer auf macOS, 30
Registrieren von StyleVision Server auf LicenseServer unter Linux, 23
Registrieren von StyleVision Server auf LicenseServer unter Windows, 17

S

Schema-Manager,
CLI-Befehl Help, 97

© 2019-2025 Altova GmbH

Schema-Manager,
CLI-Befehl Info, 98
CLI-Befehl Initialize, 98
CLI-Befehl Install, 99
CLI-Befehl List, 100
CLI-Befehl Reset, 100
CLI-Befehl Uninstall, 101
CLI-Befehl Update, 102
CLI-Befehl Upgrade, 103
CLI-Übersicht, 97
Patch für Schema installieren, 94
Schema deinstallieren, 96
Schema installieren, 94
Schemas nach Status auflisten in, 92
starten, 89
Status von Schemas, 92
Übersicht, 86
Upgrade für Schema installieren, 94
zurücksetzen, 96
Security considerations, 36
Starten von LicenseServer auf macOS, 29
Starten von LicenseServer unter Linux, 23
Starten von LicenseServer unter Windows, 16
Starten von StyleVision Server auf macOS, 29
Starten von StyleVision Server unter Linux, 23
Starten von StyleVision Server unter Windows, 16
StyleVision Server,
auf einen neuen Rechner migrieren, 34
Features, 8
Funktionalität, 8
im FlowForce Workflow, 9
StyleVision Server DLL,
Registrierung, 19

T

Taxonomie-Manager,
CLI-Befehl Help, 116
CLI-Befehl Info, 117
CLI-Befehl Initialize, 117
CLI-Befehl Install, 118
CLI-Befehl List, 119
CLI-Befehl Reset, 119
CLI-Befehl Uninstall, 120
CLI-Befehl Update, 121
CLI-Befehl Upgrade, 122
CLI-Übersicht, 116
Installieren einer Taxonomie, 113
Patch auf Taxonomie anwenden, 113
starten, 108
Status von Taxonomien, 111
Taxonomie deinstallieren, 115
Taxonomien nach Status auflisten in, 111
Übersicht, 104
Upgrade für eine Taxonomie installieren, 113
zurücksetzen, 115

U

Umgebungseinstellungen auf macOS, 31
Umgebungseinstellungen unter Linux, 25
Upgraden von StyleVision Server unter Windows, 33

W

Windows,
Installation, 12
Upgraden von StyleVision Server unter, 33

Z

Zuweisen einer Lizenz zu StyleVision Server auf macOS, 30
Zuweisen einer Lizenz zu StyleVision Server unter Linux, 24
Zuweisen einer Lizenz zu StyleVision Server unter Windows, 18

© 2019-2025 Altova GmbH