

# Altova StyleVision Server 2025



**Manuel de l'utilisateur et de référence**

# **Altova StyleVision Server 2025**

## **Manuel de l'utilisateur et de référence**

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

---

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>6</b>
<b>2</b>	<b>Fonctionnement</b>	<b>7</b>
2.1	Dans le flux de travail FlowForce.....	8
2.2	En tant que serveur autonome.....	9
<b>3</b>	<b>Installation et licence</b>	<b>10</b>
3.1	Configuration sur Windows.....	11
3.1.1	Installation sur Windows.....	11
3.1.2	Installer sur Windows Server Core.....	12
3.1.3	Installer LicenseServer (Windows).....	14
3.1.4	Démarrer LicenseServer, StyleVision Server (Windows).....	15
3.1.5	Inscrire StyleVision Server (Windows).....	16
3.1.6	Attribuer des licences StyleVision Server (Windows).....	17
3.1.7	Notes de configuration supplémentaires (Windows).....	17
3.2	Configuration sur Linux.....	19
3.2.1	Installer sur Linux.....	19
3.2.2	Installer LicenseServer (Linux).....	21
3.2.3	Démarrer LicenseServer, StyleVision Server (Linux).....	22
3.2.4	Enregistrer StyleVision Server (Linux).....	22
3.2.5	Attribuer licence (Linux).....	23
3.2.6	Notes relatives à l'environnement (Linux).....	24
3.3	Configuration sur macOS.....	26
3.3.1	Installer sur macOS.....	26
3.3.2	Installer LicenseServer (macOS).....	28
3.3.3	Démarrer LicenseServer, StyleVision Server (macOS).....	28
3.3.4	Inscrire StyleVision Server (macOS).....	29
3.3.5	Attribuer une licence (macOS).....	29

---

3.3.6	Notes relatives à l'environnement (macOS).....	30
3.4	Mise à jour StyleVision Server.....	32
3.5	Migrer StyleVision Server vers un nouvel appareil.....	33
3.6	Exigences FOP.....	34
<b>4</b>	<b>Ligne de commande StyleVision Server</b>	<b>35</b>
4.1	accepteula (Linux uniquement).....	37
4.2	assignlicense.....	38
4.3	exportresourcestrings.....	40
4.4	generate.....	42
4.5	help .....	47
4.6	licenseserver.....	48
4.7	pdfdata.....	50
4.8	setdeflang.....	52
4.9	setfopath.....	53
4.10	verifylicense.....	55
4.11	version.....	57
<b>5</b>	<b>StyleVision Server API</b>	<b>58</b>
5.1	À propos de l'interface .NET.....	59
5.2	À propos de l'interface COM.....	60
5.3	À propos de l'interface Java.....	61
5.4	Exemples de code.....	63
5.4.1	C# .....	63
5.4.2	C++ .....	64
5.4.3	Java .....	66
5.4.4	VBScript.....	67
5.4.5	Visual Basic.....	69
5.5	Référence API.....	71
5.5.1	COM et .NET.....	71
5.5.2	Java .....	78
<b>6</b>	<b>Gestionnaire de schéma</b>	<b>82</b>

6.1	Exécuter Schema Manager.....	86
6.2	Catégories de statut.....	89
6.3	Retoucher ou Installer un schéma.....	91
6.4	Désinstaller un schéma, Réinitialiser.....	93
6.5	Interface de ligne de commande (CLI).....	94
6.5.1	help .....	94
6.5.2	info .....	95
6.5.3	initialize.....	95
6.5.4	install .....	96
6.5.5	list .....	96
6.5.6	reset .....	97
6.5.7	uninstall.....	98
6.5.8	update .....	99
6.5.9	upgrade.....	99

## **7 Gestionnaire de taxonomie 101**

7.1	Exécuter le Gestionnaire de taxonomie.....	106
7.2	Catégories de statut.....	109
7.3	Retoucher ou Installer une taxonomie.....	111
7.4	Désinstaller une taxonomie, Réinitialiser.....	113
7.5	Interface de ligne de commande (CLI).....	114
7.5.1	help .....	114
7.5.2	info .....	115
7.5.3	initialize.....	115
7.5.4	install .....	116
7.5.5	list .....	116
7.5.6	reset .....	117
7.5.7	uninstall.....	118
7.5.8	update.....	119
7.5.9	upgrade.....	119

## **Index 121**

# 1 Introduction

**Altova StyleVision Server** est une mise en œuvre du moteur d'exécution intégré de [Altova StyleVision](#). Il peut être utilisé comme [produit de serveur autonome](#) ou comme module de [FlowForce Server](#) d'Altova.



StyleVision Server exécute les packages de transformation qui sont stockés sur le disque ou qui ont été déployés sur [FlowForce Server](#) :

- La fonction de StyleVision Server peut être invoquée [via la ligne de commande](#) pour exécuter le package de transformation et générer les fichiers spécifiés dans le package.
- Les transformations de StyleVision Server sont initiées par [FlowForce Server](#) basées sur le déclencheurs de temps, les déclencheurs de fichier, ou des déclencheurs à distance définis dans FlowForce Server.

StyleVision Server est disponible aussi bien pour les appareils Windows 32-bit et 64-bit. Pour les détails sur l'installation et la mise sous licence, définir les sections de configuration pour [Windows](#), [Linux](#) et [macOS](#).

**Note** : si les parties à remplir d'un PDF à remplir manquent lorsqu'un PDF est ouvert dans un système macOS, une des causes est que Java 6 n'est pas installé sur la machine. Dans ce cas, vous pouvez installer Java 6 depuis [https://support.apple.com/kb/dl1572?locale=en\\_US](https://support.apple.com/kb/dl1572?locale=en_US). Si une version plus récente que Java 6 a déjà été installée sur votre système, la version plus ancienne de Java 6 n'affectera pas le fonctionnement de la version plus récente, qui sera la version par défaut du système.

*Dernière mise à jour : 13.02.2025*

## 2 Fonctionnement

StyleVision Server transforme les fichiers XML dans des documents de sortie HTML, PDF, RTF et DOCX à l'aide des feuilles de style XSLT. Ces feuilles de style XSLT sont obtenues par le biais de fichiers PXF qui ont été créées dans l'application Altova de conception de feuilles de style, [Altova StyleVision](#).

StyleVision Server peut être utilisé de deux manières :

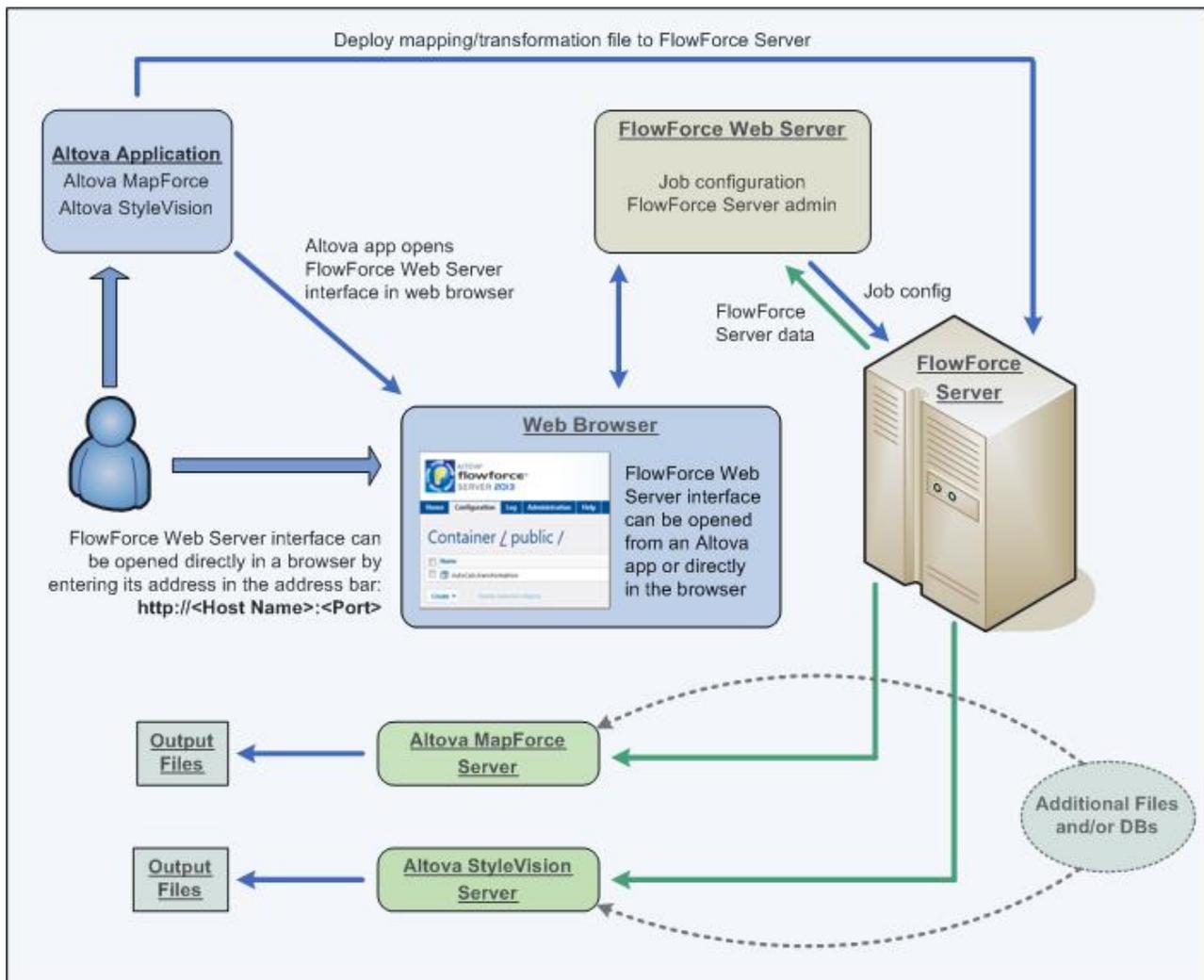
- En tant que partie prenante du [flux de travail FlowForce Altova](#)<sup>8</sup>. Pour plus d'information concernant [Altova FlowForce](#), consulter le [site web Altova](#).
- [En tant qu'un produit de serveur autonome](#)<sup>8</sup> qui est accédé par le biais de l'interface de ligne de commande (CLI).

Un fichier d'entrée XML et un fichier PXF sont envoyés au StyleVision Server, qui produit les documents de sortie requis.

## 2.1 Dans le flux de travail FlowForce

Une tâche FlowForce est créée dans [Altova FlowForce Server](#). La tâche FlowForce spécifie : (i) les entrées et les sorties d'une transformation StyleVision Server ; et (ii) les déclencheurs aux moments où la tâche devra être exécutée, comme un moment spécifique tous les jours. Au moment de l'exécution, Altova FlowForce Server passe les instructions de transformation au StyleVision Server, qui effectue ensuite de la transformation.

Le rôle de StyleVision Server dans le flux de travail FlowForce est affiché dans le diagramme ci-dessous. (Le rôle de MapForce Server dans le flux de travail est aussi affiché puisque des tâches FlowForce peuvent être créées qui envoient les mappages [Altova MapForce](#) vers [Altova MapForce Server](#) pour l'exécution.)



Outre le fait d'être invoqué par une tâche FlowForce, StyleVision Server peut aussi être invoqué par le biais de la ligne de commande. L'utilisation est décrite dans la section, [Utilisation de la ligne de commande StyleVision](#) <sup>35</sup>.

## 2.2 En tant que serveur autonome

StyleVision Server peut être installé en tant que produit autonome sur les systèmes Windows, Linux et macOS. Dans cette version, cette fonction est invoquée uniquement par le biais de la ligne de commande. L'utilisation est décrite dans cette section [Utilisation de la ligne de commande](#)<sup>35</sup>.

## 3 Installation et licence

Cette section décrit l'installation, la gestion de licence et d'autres procédures de configuration. Elle est organisée en sections comme suit :

- [Configuration sur Windows](#) <sup>11</sup>
- [Configuration sur Linux](#) <sup>19</sup>
- [Configuration sur macOS](#) <sup>26</sup>
- [Mise à niveau de StyleVision Server](#) <sup>32</sup>
- [Migrer StyleVision Server vers un nouvel appareil](#) <sup>33</sup>

## 3.1 Configuration sur Windows

Cette section décrit l'[installation](#)<sup>11</sup> et la licence de StyleVision Server sur les systèmes Windows. La configuration comprend les étapes suivantes :

1. [Installer StyleVision Server](#)<sup>11</sup>
2. [Installer LicenseServer](#)<sup>14</sup>
3. [Démarrer LicenseServer et StyleVision Server](#)<sup>15</sup>
4. [Inscrire StyleVision Server avec LicenseServer](#)<sup>16</sup>
5. [Attribuer une licence à StyleVision Server](#)<sup>17</sup>

Les étapes de configuration décrites ci-dessus apparaissent exactement dans le même ordre dans lequel elles sont recensées. Toutefois, vous devrez procéder à l'installation avant de commencer. Et vous devez inscrire StyleVision Server avec LicenseServer avant de pouvoir attribuer une licence à StyleVision Server depuis LicenseServer.

### Exigences de système (Windows)

Notez les exigences système suivantes :

- Windows 10, Windows 11
- Windows Server 2016 ou plus récent

### Prérequis

Notez les exigences préalables suivantes :

- Réaliser une installation en tant qu'utilisateur de privilèges administratifs.
- À partir de la version 2021, une version 32-bit de StyleVision Server ne peut pas être installée sur la version 64-bit, ou une version 64-bit sur une version 32-bit. Vous devez soit (i) supprimer la version plus ancienne avant d'installer la nouvelle version ou (ii) mettre à niveau vers la version plus nouvelle qui est la même version bit que votre ancienne installation.

### 3.1.1 Installation sur Windows

#### Installation de StyleVision Server

StyleVision Server peut être installé sur les systèmes Windows comme suit :

- En tant que produit de serveur autonome séparé. Pour installer StyleVision Server, téléchargez et exécutez le programme d'installation StyleVision Server. Suivez les instructions sur écran.
- Pour installer StyleVision Server comme partie du package [FlowForce Server](#), téléchargez et exécutez le programme d'installation FlowForce Server. Suivez les installations sur écran veillez à cocher l'option pour installer StyleVision Server.

Les programmes d'installation pour StyleVision Server et [FlowForce Server](#) sont disponibles au Centre de téléchargement Altova (<https://www.altova.com/fr/download.html>). Vous pouvez sélectionner votre langue d'installation depuis la zone inférieure gauche de l'assistant. Notez que cette sélection définit également la

langue par défaut de StyleVision Server. Vous pouvez changer la langue plus tard depuis la ligne de commande.

Après l'installation, le programme d'exécution StyleVision Server sera situé par défaut sous le chemin suivant :

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

Tous les enregistrements nécessaires pour utiliser StyleVision Server via une interface COM, comme une interface Java, et dans l'environnement .NET seront effectués par le programme d'installation.

## Désinstaller StyleVision Server

Désinstaller StyleVision Server comme suit :

1. Cliquez de la touche droite sur le bouton Windows **Démarrer** et sélectionnez **Paramètres**.
2. Ouvrez le panneau de configuration (commencez à écrire « Panneau de configuration » et cliquez sur l'entrée suggérée).
3. Sous *Programmes*, cliquez **Désinstaller un programme**.
4. Dans le panneau de configuration, sélectionnez StyleVision Server et cliquez sur **Désinstaller**.

## Licence d'évaluation

Pendant le processus d'installation, vous recevrez l'option de demander une licence d'évaluation de 30 jours pour StyleVision Server. Après avoir soumis la demande, une licence d'évaluation sera envoyée à l'adresse e-mail que vous avez enregistré.

### 3.1.2 Installer sur Windows Server Core

Windows Server Core est une installation Windows minimale qui n'utilise qu'un certain nombre de fonctions de la GUI. Vous pouvez installer StyleVision Server sur un appareil Windows Server Core comme suit :

1. Téléchargez le programme d'installation StyleVision Server exécutable depuis le site web d'Altova. Ce fichier est nommé `styleVisionServer<version>.exe`. Assurez-vous que l'exécutable corresponde à votre plate-forme de serveur (32-bit ou 64-bit).
2. Sur un appareil standard Windows (pas l'appareil Windows Server Core), exécutez la commande `styleVisionServer<version>.exe /u`. Ceci déballe le fichier `.msi` vers le même dossier que celui du programme d'installation.
3. Copiez le fichier décompressé `.msi` vers l'appareil Windows Server Core.
4. Si vous mettez à jour une version antérieure de StyleVision Server, fermez StyleVision Server avant de réaliser la prochaine étape.
5. Utilisez le fichier `.msi` pour l'installation en exécutant la commande `msiexec /i styleVisionServer.msi`. Ceci lance l'installation de Windows Server Core.

#### Important : Garder le fichier MSI !

Veillez noter les points suivants :

- Gardez le fichier `1'extraction.msi` à un endroit sûr. Vous en aurez besoin plus tard pour désinstaller, réparer ou modifier votre installation.

- Si vous voulez renommer le fichier MSI, faites-le avant d'installer StyleVision Server.
- Le nom du fichier MSI est stocké dans le registre. Vous pouvez mettre à jour son nom ici si le nom du fichier a changé.

## Inscrire StyleVision Server avec LicenseServer

Si vous installez StyleVision Server pour la première fois ou si vous mettez à jour vers une **version majeure**, vous allez devoir enregistrer StyleVision Server avec Altova LicenseServer sur votre réseau. Si vous êtes en train de mettre à jour vers une version non majeure de StyleVision Server, alors l'enregistrement précédent de LicenseServer sera connu par l'installation et vous n'aurez pas besoin d'enregistrer StyleVision Server avec LicenseServer. Toutefois, si vous voulez changer le LicenseServer qui est utilisé par StyleVision Server à tout moment, vous allez devoir enregistrer StyleVision Server avec le nouveau LicenseServer.

Pour enregistrer StyleVision Server avec Altova LicenseServer pendant l'installation, exécutez la commande d'installation avec la propriété `REGISTER_WITH_LICENSE_SERVER`, telle que recensée ci-dessous, fournissant le nom ou l'adresse de la machine de LicenseServer en tant que valeur de la propriété, par exemple :

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

Pour enregistrer StyleVision Server avec un Altova LicenseServer après l'installation, exécutez la commande suivante :

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

## Commandes utiles

Vous trouverez ci-dessous un ensemble de commandes utiles dans le contexte de l'installation.

Pour tester la valeur retour de votre installation, exécutez un script semblable à celui ci-dessous. Le code retour sera dans la variable d'environnement `%errorlevel%` . Un code retour 0 indique un succès.

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

Pour une installation silencieuse avec un code retour et un log de la procédure d'installation :

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

Pour modifier l'installation :

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

Pour réparer l'installation :

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

Pour désinstaller StyleVision Server :

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

Pour la désinstallation de StyleVision Server en silence et faire rapport du résultat détaillé dans un fichier log :

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

Pour installer StyleVision Server en utilisant un autre langage (les codes de langage disponibles sont : allemand=de; espagnol=es; français=fr) :

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

**Note :** Sur Windows Server Core, la fonctionnalité des graphiques et du code-barre de StyleVision Server ne sera pas disponible.

**Note :** Pour installer les taxonomies, utilisez le Taxonomy Package Manager par le biais de la ligne de commande. Consultez le manuel StyleVision Server pour des informations sur comment faire.

### 3.1.3 Installer LicenseServer (Windows)

Pour que StyleVision Server fonctionne, il doit être mis sous licence par le biais de [Altova LicenseServer](#) sur votre réseau. Lorsque vous installez StyleVision Server ou FlowForce Server sur les systèmes Windows, vous pouvez installer LicenseServer ensemble avec StyleVision Server ou FlowForce Server. Si une licence est déjà installée sur votre réseau, vous ne devez plus en installer une autre - sauf si une version plus récente de LicenseServer est requise. (Voir le *prochain point*, versions de [LicenseServer](#).)

Pendant la procédure d'installation de StyleVision Server ou de FlowForce Server, activez ou désactivez l'option pour installer LicenseServer, si besoin. Veuillez noter les points suivants :

- Si vous n'avez pas encore installé LicenseServer, laissez les paramètres par défaut tels quels. L'assistant installera la dernière version sur l'ordinateur sur lequel vous exécutez l'assistant.
- Si vous n'avez pas encore installé LicenseServer et souhaitez installer Altova LicenseServer sur un autre ordinateur, décochez la case *Installer Altova LicenseServer sur l'appareil* et choisissez **Inscrire plus tard**. Dans ce cas, vous devrez installer LicenseServer séparément et inscrire StyleVision Server après.
- Si LicenseServer a déjà été installé sur votre ordinateur mais qu'il s'agit d'une version inférieure à celle indiquée par l'assistant d'installation, laissez les paramètres par défaut (pour mettre à jour à la version plus récente) tels quels. Dans ce cas, l'assistant d'installation mettra automatiquement votre version de LicenseServer à jour. L'information d'inscription et de licence existante sera reportée à la nouvelle version de LicenseServer.
- Si LicenseServer a été installé sur votre ordinateur ou sur votre réseau et a la même version que celle indiquée par l'assistant, suivez les étapes suivantes :
  - Décochez la case *Installer Altova LicenseServer sur l'appareil*.
  - Sous *Inscrire ce produit avec*, choisissez le serveur de licence avec lequel vous voulez vous inscrire StyleVision Server. De manière alternative, choisissez **Inscrire plus tard**. Veuillez noter que vous pouvez toujours sélectionner **Inscrire plus tard** si vous voulez ignorer les associations de LicenseServer et poursuivre l'installation de StyleVision Server.

Pour plus d'information sur l'[inscription](#)<sup>16</sup> et la [licence](#)<sup>17</sup> StyleVision Server avec [Altova LicenseServer](#), voir la section [Documentation LicenseServer](#).

#### Versions de LicenseServer

- Les produits de Altova doivent être dotés d'une licence soit () avec une version de LicenseServer qui correspond à la version installée StyleVision Server, soit pour une version ultérieure de LicenseServer.
- La version LicenseServer qui correspond à la version actuelle de StyleVision Server est [3.17](#).
- Sur Windows, vous pouvez installer la version correspondante de LicenseServer comme faisant partie de l'installation de StyleVision Server ou installez LicenseServer séparément. Sur Linux et macOS, vous devez installer LicenseServer séparément.
- Avant que la nouvelle version de LicenseServer ne soit installée, chaque version plus ancienne doit être désinstallée.
- Lors de la désinstallation de LicenseServer, toute l'information liée à l'inscription et à la licence

contenue dans la version plus ancienne de LicenseServer sera enregistrée dans une base de données sur votre appareil de serveur. Ces données seront importées automatiquement dans la version plus nouvelle lorsque celle-ci est installée.

- Les versions LicenseServer sont rétro-compatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- La dernière version de LicenseServer disponible sur le site web d'Altova. Cette version fonctionnera avec toute version actuelle ou ancienne de StyleVision Server.
- Le numéro de version de la licence LicenseServer actuellement installée est indiqué au bas de la [page de configuration LicenseServer](#) (tous les onglets).

### 3.1.4 Démarrer LicenseServer, StyleVision Server (Windows)

Altova LicenseServer (désigné LicenseServer) et StyleVision Server sont tous les deux démarrés depuis Altova ServiceController.

#### Altova ServiceController

Altova ServiceController (ServiceController en abrégé) est une application permettant de lancer, d'arrêter et de configurer confortablement les services Altova **ms**. ServiceController est installé avec l'Altova LicenseServer et avec les produits de serveur Altova installés en tant que services (DiffDog Server, FlowForce Server, Mobile Together Server et RaptorXML(+XBRL) Server). ServiceController peut être accédé par le biais de la barre d'état système (voir la capture d'écran ci-dessous).

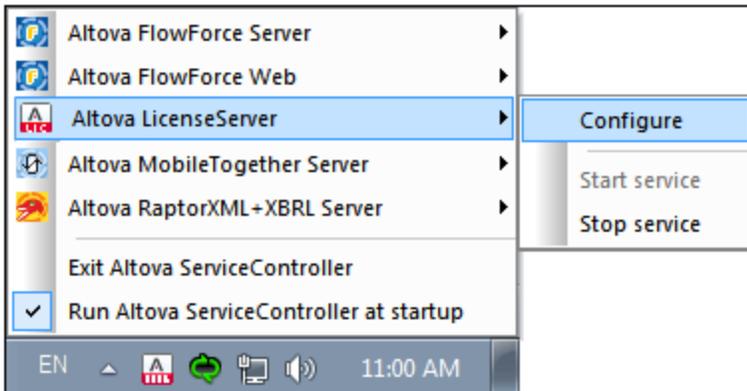


Pour spécifier que ServiceController démarre automatiquement lors de la connexion au système, cliquer sur l'icône **ServiceController** dans la zone de notification pour afficher le menu **ServiceController** (voir la capture d'écran ci-dessous) et basculez sur la commande **Run Altova ServiceController at Startup**. (Cette commande est activée par défaut.) Pour quitter ServiceController, cliquer sur l'icône **ServiceController** dans la zone de notification et, dans le menu qui apparaît (voir la capture d'écran ci-dessous), cliquez sur **Exit Altova ServiceController**.



## Démarrer LicenseServer

Pour démarrer LicenseServer, cliquez sur l'icône **ServiceController** dans la zone de modification, passez au-dessus de **Altova LicenseServer** dans le menu qui s'ouvre (*voir la capture d'écran ci-dessous*), puis sélectionnez **Start Service** depuis le sous-menu de LicenseServer. Si LicenseServer est déjà en cours d'exécution, l'option *Démarrer le service* sera désactivée. Vous pouvez également arrêter le service par le biais de ServiceController.



### 3.1.5 Inscire StyleVision Server (Windows)

Afin de pouvoir détenir une licence StyleVision Server depuis Altova LicenseServer, StyleVision Server doit être inscrit avec LicenseServer. Pour enregistrer StyleVision Server depuis l'interface de la ligne de commande, utilisez la commande `licenseserver` et donnez l'adresse de l'appareil LicenseServer (*voir ci-dessous*).

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

Par exemple, si `localhost` est le nom du serveur sur lequel LicenseServer est installé, utilisez la commande suivante :

```
StyleVisionServer licenseserver localhost
```

Si StyleVision Server a été installée faisant partie d'une installation de [FlowForce Server](#), enregistrer FlowForce Server avec LicenseServer enregistrera aussi automatiquement StyleVision Server. Procédez comme suit : (i) démarrez Altova FlowForce Web par le biais de ServiceController (*voir les points précédents*); (ii) saisissez votre mot de passe pour accéder à la page Setup; (iii) sélectionnez le nom ou l'adresse du LicenseServer et cliquez sur **Inscire avec LicenseServer**. Pour plus d'informations, voir [Inscire FlowForce Server](#).

Après avoir réussi l'inscription, allez à l'[onglet de gestion des clients de la page de configuration de LicenseServer](#) pour attribuer une licence à StyleVision Server.

Pour plus d'informations sur l'inscription des produits Altova avec LicenseServer, voir le [manuel utilisateur de LicenseServer](#).

### 3.1.6 Attribuer des licences StyleVision Server (Windows)

Après avoir réussi l'inscription de StyleVision Server, elle sera recensée dans l'onglet de gestion des clients de la page de configuration de LicenseServer. Allez-y et [attribuez une licence](#) à StyleVision Server.

L'obtention de la licence des produits de serveur Altova, est basée sur le nombre de cœurs de processeurs disponibles sur la machine du produit. Par exemple, un processeur double cœur a deux cœurs, un processeur quadricœur a quatre cœurs, un processeur hexacœur a six cœurs, etc. Le nombre de cœurs pour lesquels une licence a été délivrée pour un produit doit être supérieur ou égal au nombre de cœurs disponibles sur cette machine de serveur, que ce serveur soit une machine physique ou virtuelle. Par exemple, si un serveur a huit cœurs (un processeur octacœur), vous devrez acheter au moins une licence octacœur. Vous pouvez aussi additionner les licences pour obtenir le nombre souhaité de cœurs. Ainsi, deux licences de quadricœurs peuvent être utilisées pour un serveur octacœur au lieu d'acheter une licence octacœur.

Si vous utilisez un serveur d'ordinateur avec un grand nombre de cœurs CPU, mais ne disposez que d'un faible volume à traiter, vous pouvez aussi créer une machine virtuelle qui disposera d'un plus petit nombre de cœurs et acheter une licence pour ce nombre de cœurs. Il va de soi que la vitesse de traitement d'un tel déploiement sera moins rapide que si tous les cœurs disponibles sur le serveur étaient utilisés.

**Note :** Chaque licence de produit de serveur Altova peut être utilisée pour une seule machine client à la fois, même si la licence a une capacité de licence qui n'est pas utilisée (l'appareil client est l'appareil sur lequel le produit de serveur Altova est installé). Par exemple, si une licence de 10-cœurs est utilisée pour une machine client qui détient 6 cœurs CPU, les 4 cœurs restants de la capacité de licence ne pourront pas être utilisés simultanément pour une autre machine client.

#### Exécution thread unique

Si un produit de serveur Altova permet une exécution single-thread, une option pour une *exécution single-thread* sera disponible. Dans ces cas, la licence produit du serveur Altova pour uniquement un cœur est disponible dans le pool des licences, un appareil avec des cœurs multiples peut être assigné à cette licence à one-core. Dans un tel cas, l'appareil exécutera ce produit en single-core. Le traitement sera donc plus lent car le multi-threading (qui est possible sur de multiples cœurs) ne sera pas disponible. Le produit sera exécuté en mode single thread sur cet appareil.

Pour assigner une licence single-core à un appareil multiple-core dans LicenseServer, sélectionnez la case à cocher *Limit to single thread execution* pour ce produit.

#### Estimation des exigences core

Il existe de nombreux facteurs externes divers qui influent sur les volumes de données et les temps de traitement que votre serveur arrive à gérer (par exemple : le matériel, la charge actuelle sur le CPU, et l'attribution de la mémoire d'autres applications exécutées sur le serveur). Afin de mesurer la performance aussi précisément que possible, testez les applications dans votre environnement avec les volumes de données et les conditions qui établissent aussi fidèlement que possible des situations professionnelles réelles.

### 3.1.7 Notes de configuration supplémentaires (Windows)

Pour exécuter les exemples de Windows qui sont packagés avec StyleVision Server dans le sous-dossier `etc\examples` du dossier de l'application, StyleVision Server DLL doit être correctement inscrit avec le système. Une erreur d'inscription apparaît généralement si vous avez installé avec le temps différentes versions-bit de StyleVision Server (32-bit et 64-bit) sur un seul appareil.

Pour inscrire correctement StyleVision Server DLL (soit la version 32-bit ou 64-bit) sur les appareils Windows, suivez les étapes suivantes :

1. Ouvrez une invite de commande en mode administrateur
2. Basculez vers le dossier dans lequel se situe le DLL. La commande qu'il vous faut est la suivante : `cd C:\Program Files\Altova\StyleVisionServer2025\bin`
3. Exécutez la commande suivante pour inscrire le DLL (soit 32-bit ou 64-bit) : `regsvr32 styleVisionServer.dll`
4. Assurez-vous qu'il ait un popup confirmant l'inscription
5. Open Visual Studio
6. Chargez le projet en utilisant `styleVisionServerAPI_Sample.sln`
7. Confirmez que votre fichier `Program.cs` contient des chemins valides
8. Exécutez le projet en utilisant **Ctrl+F5**

**Note** : Le chemin vers le dossier d'application sur les systèmes Windows est généralement : `C:\Program Files\Altova\Common2025`

## 3.2 Configuration sur Linux

Cette section décrit l'[installation](#)<sup>19</sup> et la licence de StyleVision Server sur les systèmes Linux (Debian, Ubuntu, CentOS, RedHat). La configuration comprend les étapes suivantes :

1. [Installer StyleVision Server](#)<sup>19</sup>
2. [Installer LicenseServer](#)<sup>21</sup>
3. [Démarrer LicenseServer](#)<sup>22</sup>
4. [Inscrire StyleVision Server avec LicenseServer](#)<sup>22</sup>
5. [Attribuer une licence à StyleVision Server](#)<sup>23</sup>

Les étapes de configuration décrites ci-dessus apparaissent exactement dans le même ordre dans lequel elles sont recensées. Toutefois, vous devrez procéder à l'installation avant de commencer. Et vous devez inscrire StyleVision Server avec LicenseServer avant de pouvoir attribuer une licence à StyleVision Server depuis LicenseServer.

### Exigences de système (Linux)

- Red Hat Enterprise Linux 7 ou plus récent
- CentOS 7, CentOS Stream 8
- Debian 10 ou plus récent
- Ubuntu 20.04, 22.04, 24.04
- AlmaLinux 9.0
- Rocky Linux 9.0

### Prérequis

- Réaliser l'installation soit en tant qu'utilisateur **root**, soit en tant qu'utilisateur avec des privilèges **sudo**.
- La version précédente de StyleVision Server doit être désinstallée avant d'installer une nouvelle.
- Si vous avez l'intention d'utiliser la fonctionnalité des graphiques d'Altova, vous devez installer au moins une police sur votre système pour vous assurer que les graphiques seront rendus correctement. Pour recenser les polices installées, utilisez par exemple la commande `fc-list` de [Fontconfig library](#).
- Les bibliothèques suivantes sont requises comme prérequis pour installer et exécuter l'application. Si les packages ci-dessous ne sont pas déjà disponibles sur l'appareil Linux, exécutez la commande `yum` (ou `apt-get`, si applicable) pour les installer.

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

### 3.2.1 Installer sur Linux

StyleVision Server est disponible pour l'installation sur les systèmes Linux. Réaliser installation soit en tant qu'utilisateur `root`, soit en tant qu'utilisateur avec des privilèges `sudo`.

## Intégration de FlowForce Server et d'autres produits de serveur d'Altova

Si vous installez StyleVision Server ensemble avec FlowForce Server, il est recommandé d'installer d'abord FlowForce Server. Si vous installez StyleVision Server avant FlowForce Server, il faut, après avoir installé StyleVision Server et FlowForce Server, exécuter la commande suivante :

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

Cette commande copie le fichier `.tool` depuis le répertoire `/etc` de StyleVision Server vers le répertoire FlowForce Server `/tools`. Le fichier `.tool` est requis par FlowForce Server. Il contient le chemin vers l'StyleVision Server exécutable. Vous n'avez pas besoin d'exécuter cette commande si vous installez FlowForce Server avant d'installer StyleVision Server.

## Désinstaller StyleVision Server

Avant d'installer StyleVision Server, vous devez désinstaller toute version plus ancienne.

Pour vérifier quels produits de serveur Altova sont installés :

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

Pour désinstaller une version plus ancienne de StyleVision Server:

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

Sur les systèmes Debian et Ubuntu, il se pourrait que StyleVision Server apparaisse encore dans la liste des produits installés après son installation. Dans ce cas, exécutez la commande `purge` pour effacer StyleVision Server de la liste. Vous pouvez aussi utiliser la commande `purge` au lieu de la commande `remove` recensée ci-dessus.

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

## Télécharger le package Linux StyleVision Server

Les packages d'installation pour StyleVision Server pour les systèmes Linux suivants sont disponibles sur le [site web d'Altova](#).

Distribution	Extension de package
Debian	.deb
Ubuntu	.deb
CentOS	.rpm
RedHat	.rpm

Après avoir téléchargé le package Linux, copiez-le dans n'importe quel répertoire sur le système Linux. Puisque vous allez devoir mettre sous licence StyleVision Server with an [Altova LicenseServer](#), vous

téléchargez LicenseServer depuis le [site web d'Altova](#) en même temps que vous téléchargez StyleVision Server, plutôt que de le télécharger plus tard.

## Installer StyleVision Server

Dans une fenêtre de terminal, basculez vers le répertoire où vous copiez le package Linux. Par exemple, si vous le copiez dans un répertoire d'utilisateur appelé `MyAltova` qui est situé dans le répertoire `/home/User`, basculez vers ce répertoire comme suit :

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

Installer StyleVision Server utilisant la commande pertinente :

```
[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
```

Vous devrez éventuellement ajuster le nom du package ci-dessus pour qu'il corresponde au release ou à la version service pack actuels.

Le package StyleVision Server sera installé dans le dossier suivant :

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

## 3.2.2 Installer LicenseServer (Linux)

Pour que StyleVision Server fonctionne, il faut qu'il ait une licence par le biais d'un [Altova LicenseServer](#) sur votre réseau. Téléchargez LicenseServer depuis le [site web d'Altova](#) et copiez le package dans n'importe quel répertoire. Installez-le comme vous avez installé StyleVision Server (voir [page précédente](#)<sup>19</sup>).

```
[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
```

Le pack de LicenseServer sera installé dans le chemin suivant :

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

Pour information, voir comment s'[inscrire](#)<sup>22</sup> et [mettre sous licence](#)<sup>23</sup> StyleVision Server avec [Altova LicenseServer](#). Voir également la [documentation LicenseServer](#) pour des informations plus détaillées.

## Versions de LicenseServer

- Les produits de Altova doivent être dotés d'une licence soit () avec une version de LicenseServer qui correspond à la version installée StyleVision Server, soit pour une version ultérieure de LicenseServer.
- La version LicenseServer qui correspond à la version actuelle de StyleVision Server est **3.17**.
- Sur Windows, vous pouvez installer la version correspondante de LicenseServer comme faisant partie de l'installation de StyleVision Server ou installez LicenseServer séparément. Sur Linux et macOS,

vous devez installer LicenseServer séparément.

- Avant que la nouvelle version de LicenseServer ne soit installée, chaque version plus ancienne doit être désinstallée.
- Lors de la désinstallation de LicenseServer, toute l'information liée à l'inscription et à la licence contenue dans la version plus ancienne de LicenseServer sera enregistrée dans une base de données sur votre appareil de serveur. Ces données seront importées automatiquement dans la version plus nouvelle lorsque celle-ci est installée.
- Les versions LicenseServer sont rétro-compatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- La dernière version de LicenseServer disponible sur le site web d'Altova. Cette version fonctionnera avec toute version actuelle ou ancienne de StyleVision Server.
- Le numéro de version de la licence LicenseServer actuellement installée est indiqué au bas de la [page de configuration LicenseServer](#) (tous les onglets).

### 3.2.3 Démarrer LicenseServer, StyleVision Server (Linux)

Démarrer Altova LicenseServer et StyleVision Server soit comme utilisateur `root` ou comme utilisateur avec des privilèges `sudo`.

#### Démarrer LicenseServer

Pour s'enregistrer et gérer la licence StyleVision Server avec LicenseServer, celui-ci doit être exécuté en tant que daemon sur le réseau. Démarrez LicenseServer en tant que daemon avec la commande suivante :

```
sudo systemctl start licenseserver
```

Si à un moment ou un autre, vous êtes amenés à devoir arrêter LicenseServer, remplacez `start` par `arrêter` dans les commandes ci-dessus. Par exemple :

```
sudo systemctl stop licenseserver
```

### 3.2.4 Enregistrer StyleVision Server (Linux)

Afin de pouvoir détenir une licence StyleVision Server depuis Altova LicenseServer, StyleVision Server doit être enregistré avec LicenseServer

Pour inscrire StyleVision Server, allez à son CLI et utilisez la commande `licenseserver` :

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

Par exemple, si `localhost` est le nom du serveur sur lequel LicenseServer est installé :

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

Dans la commande ci-dessus, `localhost` est le nom du serveur sur lequel LicenseServer est installé. Notez également que l'emplacement du programme exécutable de StyleVision Server est :

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

Après avoir réussi l'enregistrement, allez à l'[onglet de gestion des clients de la page de configuration de LicenseServer](#) pour attribuer une licence à StyleVision Server.

Pour plus d'informations sur l'enregistrement des produits Altova avec LicenseServer, voir le [manuel utilisateur de LicenseServer](#).

### 3.2.5 Attribuer licence (Linux)

Après avoir réussi l'inscription de StyleVision Server, elle sera recensée dans l'onglet de gestion des clients de la page de configuration de LicenseServer. Allez-y et [attribuez une licence](#) à StyleVision Server.

L'obtention de la licence des produits de serveur Altova, est basée sur le nombre de cœurs de processeurs disponibles sur la machine du produit. Par exemple, un processeur double cœur a deux cœurs, un processeur quadricœur a quatre cœurs, un processeur hexacœurs a six cœurs, etc. Le nombre de cœurs pour lesquels une licence a été délivrée pour un produit doit être supérieur ou égal au nombre de cœurs disponibles sur cette machine de serveur, que ce serveur soit une machine physique ou virtuelle. Par exemple, si un serveur a huit cœurs (un processeur octacœur), vous devrez acheter au moins une licence octacœur. Vous pouvez aussi additionner les licences pour obtenir le nombre souhaité de cœurs. Ainsi, deux licences de quadricœurs peuvent être utilisées pour un serveur octacœur au lieu d'acheter une licence octacœur.

Si vous utilisez un serveur d'ordinateur avec un grand nombre de cœurs CPU, mais ne disposez que d'un faible volume à traiter, vous pouvez aussi créer une machine virtuelle qui disposera d'un plus petit nombre de cœurs et acheter une licence pour ce nombre de cœurs. Il va de soi que la vitesse de traitement d'un tel déploiement sera moins rapide que si tous les cœurs disponibles sur le serveur étaient utilisés.

**Note :** Chaque licence de produit de serveur Altova peut être utilisée pour une seule machine client à la fois, même si la licence a une capacité de licence qui n'est pas utilisée (l'appareil client est l'appareil sur lequel le produit de serveur Altova est installé). Par exemple, si une licence de 10-cœurs est utilisée pour une machine client qui détient 6 cœurs CPU, les 4 cœurs restants de la capacité de licence ne pourront pas être utilisés simultanément pour une autre machine client.

#### Exécution thread unique

Si un produit de serveur Altova permet une exécution single-thread, une option pour une *exécution single-thread* sera disponible. Dans ces cas, la licence produit du serveur Altova pour uniquement un cœur est disponible dans le pool des licences, un appareil avec des cœurs multiples peut être assigné à cette licence à one-core. Dans un tel cas, l'appareil exécutera ce produit en single-core. Le traitement sera donc plus lent car le multi-threading (qui est possible sur de multiples cœurs) ne sera pas disponible. Le produit sera exécuté en mode single thread sur cet appareil.

Pour assigner une licence single-core à un appareil multiple-core dans LicenseServer, sélectionnez la case à cocher *Limit to single thread execution* pour ce produit.

#### Estimation des exigences core

Il existe de nombreux facteurs externes divers qui influent sur les volumes de données et les temps de traitement que votre serveur arrive à gérer (par exemple : le matériel, la charge actuelle sur le CPU, et l'attribution de la mémoire d'autres applications exécutées sur le serveur). Afin de mesurer la performance aussi précisément que possible, testez les applications dans votre environnement avec les volumes de données et les conditions qui établissent aussi fidèlement que possible des situations professionnelles réelles.

## 3.2.6 Notes relatives à l'environnement (Linux)

### Dossiers

Ci-dessous, vous trouverez une liste de dossiers importants dans votre configuration StyleVision Server.

#### ▣ Racine d'installation

`/opt/Altova/StyleVisionServer2025/`

#### ▣ Fichiers de licence

`/var/opt/Altova/StyleVisionServer`

#### ▣ Paramètres d'environnement

`/etc/profile.d/jdbc.sh`

Le fichier des paramètres de fichier (dénommé `jdbc.sh`) est exécuté lors du démarrage du système. Les définitions y incluses doivent être spécifiques à votre environnement particulier. Le chemin d'exemple ci-dessus a vocation de guide général.

**Note :** Le fichier des paramètres d'environnement définit les variables pour **tous les utilisateurs** sur le système, donc vous devez faire attention lorsque vous modifiez les paramètres. Par exemple, si vous modifiez un chemin d'accès dans ce fichier, les modifications seront alors appliquées dans tout le système. Si vous souhaitez apporter des modifications uniquement pour StyleVision Server, vous voudrez éventuellement utiliser un fichier d'unité (l'explication est dans la section *Connexions JDBC* ci-dessous).

### Connexions de base de données

Sur Linux, les connexions de base de données suivantes sont prises en charge :

- JDBC - vous pouvez utiliser JDBC pour toutes les bases de données prises en charge, à l'exception de Microsoft Access
- Connexions natives — actuellement disponible pour les bases de données SQLite et PostgreSQL

Si vous utilisez JDBC, veuillez noter les points suivants :

- L'environnement d'exécution Java ou SDK doit être installé.
- Les pilotes JDBC pour la base de données cible doivent être installés.
- Les variables d'environnement suivants doivent être définis correctement pour votre environnement :
  - `CLASSPATH` : pour trouver les fichiers jar qui se connectent à la base de données JDBC ; les fichiers jar peuvent être entrés soit dans un (i) script exécutable (comme `jdbc.sh`), qui est exécuté lors du démarrage de système, ou (ii) dans un fichier d'unité qui est exécuté lorsque StyleVision Server est démarré en tant que service. Utiliser un fichier d'unité pour spécifier les fichiers jar a l'avantage que les fichiers requis pour les connexions JDBC de StyleVision Server seront localisés sans que vous ayez à modifier la configuration existante du système. La liste de fichier d'unité est recensée ci-dessous.
  - `PATH` : pour trouver le JRE, mais n'est pas toujours nécessaire, dépendant de l'installation
  - `JAVA_HOME` : si nécessaire, dépendant de l'installation.

## Liste de fichiers importants

Le script shell (ou fichier d'unité) suivant est copié dans le dossier `/opt/Altova/StyleVisionServer/etc` de manière à ne pas écraser des fichiers de configuration existants. Procéder aux changements nécessaires. Voir aussi la section *Connexions JDBC* ci-dessus. Les parties en surbrillance bleue sont spécifiques à l'environnement et devront être ajustées pour correspondre à l'environnement :

### ☐ Script shell (fichier d'unité)

```
#- 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/xdm.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/iserics/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:./:
```

## 3.3 Configuration sur macOS

Cette section décrit l'[installation](#)<sup>26</sup> et la licence de StyleVision Server sur les systèmes macOS. La configuration comprend les étapes suivantes :

1. [Installer StyleVision Server](#)<sup>26</sup>
2. [Installer LicenseServer](#)<sup>28</sup>
3. [Démarrer LicenseServer](#)<sup>28</sup>
4. [Inscrire StyleVision Server avec LicenseServer](#)<sup>29</sup>
5. [Attribuer une licence à StyleVision Server](#)<sup>29</sup>

Les étapes de configuration décrites ci-dessus apparaissent exactement dans le même ordre dans lequel elles sont recensées. Toutefois, vous devrez procéder à l'installation avant de commencer. Et vous devez inscrire StyleVision Server avec LicenseServer avant de pouvoir attribuer une licence à StyleVision Server depuis LicenseServer.

### Exigences de système (macOS)

Notez les exigences système suivantes :

- macOS 12 ou plus récent

### Prérequis

Notez les exigences préalables suivantes :

- Assurez-vous d'avoir installé Altova LicenseServer et qu'il est exécuté.
- Réaliser l'installation soit en tant qu'utilisateur `root`, soit en tant qu'utilisateur avec des privilèges `sudo`.
- La version précédente de StyleVision Server doit être désinstallée avant d'installer une nouvelle.
- Si vous avez l'intention d'utiliser la fonctionnalité des graphiques d'Altova, vous devez installer au moins une police sur votre système pour vous assurer que les graphiques seront rendus correctement. Pour recenser les polices installées, utilisez par exemple la commande `fc-list` de [Fontconfig library](#).
- L'appareil macOS doit être configuré de telle façon que son nom se résout en adresse IP. Ceci signifie que vous devez être en mesure d'effectuer un ping avec succès du nom de l'hôte depuis le terminal en utilisant la commande `ping <hostname>`.

### 3.3.1 Installer sur macOS

Ce chapitre décrit l'installation et la configuration de StyleVision Server sur les systèmes macOS.

#### Intégration dans FlowForce

Si vous installez StyleVision Server ensemble avec FlowForce Server, il est recommandé d'installer d'abord FlowForce Server. Si vous installez StyleVision Server avant FlowForce Server, il faut, après avoir installé les deux, exécuter la commande suivante :

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

Cette commande copie le fichier `.tool` depuis le répertoire `/etc` de StyleVision Server vers le répertoire FlowForce Server `/tools`. Le fichier `.tool` est requis par FlowForce Server. Il contient le chemin vers l'StyleVision Server exécutable. Vous n'avez pas besoin d'exécuter cette commande si vous installez FlowForce Server avant d'installer StyleVision Server.

## Désinstaller StyleVision Server

Dans le dossier Applications, cliquez avec la touche de droite sur l'icône StyleVision Server et choisissez **Déplacer vers la Corbeille**. L'application sera déplacée dans la corbeille. Vous allez toutefois devoir déplacer l'application du dossier `usr`. Pour ce faire, utilisez la commande suivante :

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

Si vous devez désinstaller une ancienne version de Altova LicenseServer, vous devez d'abord arrêter son exécution en tant que service. Pour ce faire, utilisez la commande suivante :

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

Pour vérifier si le service a été arrêté, ouvrez le moniteur d'activités et assurez-vous que LicenseServer n'est pas dans la liste. Puis, procédez à la désinstallation de la même manière que décrit ci-dessus pour StyleVision Server.

## Installer StyleVision Server

Pour installer StyleVision Server, suivez les étapes suivantes :

1. Téléchargez le fichier de l'image de disque (`.dmg`) de StyleVision Server depuis le site Altova website (<https://www.altova.com/fr/download.html>).
2. Cliquez pour ouvrir l'image de disque téléchargée (`.dmg`). Ceci fait que le programme d'installation StyleVision Server apparaît en tant que nouveau lecteur virtuel sur votre ordinateur.
3. Sur le nouveau lecteur virtuel, double-cliquez sur le package d'installation (`.pkg`).
4. Suivez les étapes successives de l'assistant d'installation. Celles-ci sont explicites et incluent un étape lors de laquelle vous devez accepter le contrat de licence avant de pouvoir continuer.
5. Pour éjecter le lecteur après l'installation, cliquez de la touche droite sur le lecteur et sélectionnez **Éjecter**.

Le package StyleVision Server sera installé dans le dossier :

```
/usr/local/Altova/StyleVisionServer2025 (application binaries)  
/var/Altova/StyleVisionServer (data files : database and logs)
```

Le server daemonStyleVision Server démarre automatiquement après l'installation et redémarre l'appareil. Vous pouvez toujours démarrer StyleVision Server en tant que daemon avec les commandes suivantes :

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

### 3.3.2 Installer LicenseServer (macOS)

Altova LicenseServer peut être téléchargé depuis le site web d'Altova (<https://www.altova.com/fr/download.html>). Exécutez l'installation tel que décrit [ici](#) <sup>26</sup>.

Le package de LicenseServer sera installé dans le suivantes dossier :

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

Pour information, voir comment s'[inscrire](#) <sup>29</sup> et [mettre sous licence](#) <sup>29</sup> StyleVision Server avec [Altova LicenseServer](#). Voir également la [documentation LicenseServer](#) pour des informations plus détaillées.

#### Versions de LicenseServer

- Les produits de Altova doivent être dotés d'une licence soit () avec une version de LicenseServer qui correspond à la version installée StyleVision Server, soit pour une version ultérieure de LicenseServer.
- La version LicenseServer qui correspond à la version actuelle de StyleVision Server est **3.17**.
- Sur Windows, vous pouvez installer la version correspondante de LicenseServer comme faisant partie de l'installation de StyleVision Server ou installez LicenseServer séparément. Sur Linux et macOS, vous devez installer LicenseServer séparément.
- Avant que la nouvelle version de LicenseServer ne soit installée, chaque version plus ancienne doit être désinstallée.
- Lors de la désinstallation de LicenseServer, toute l'information liée à l'inscription et à la licence contenue dans la version plus ancienne de LicenseServer sera enregistrée dans une base de données sur votre appareil de serveur. Ces données seront importées automatiquement dans la version plus nouvelle lorsque celle-ci est installée.
- Les versions LicenseServer sont rétro-compatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- La dernière version de LicenseServer disponible sur le site web d'Altova. Cette version fonctionnera avec toute version actuelle ou ancienne de StyleVision Server.
- Le numéro de version de la licence LicenseServer actuellement installée est indiqué au bas de la [page de configuration LicenseServer](#) (tous les onglets).

### 3.3.3 Démarrer LicenseServer, StyleVision Server (macOS)

Démarrer Altova LicenseServer et StyleVision Server soit comme utilisateur `root`, soit comme utilisateur avec des privilèges `sudo`.

#### Démarrer LicenseServer

Pour s'enregistrer et gérer la licence StyleVision Server correctement avec LicenseServer, celui-ci doit être exécuté en tant que daemon. Lancez LicenseServer en tant que daemon avec la commande suivante :

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

Si à un moment ou un autre, vous êtes amenés à devoir arrêter LicenseServer, remplacez `charger` par `unload` dans les commandes ci-dessus.

### 3.3.4 Inscrire StyleVision Server (macOS)

Afin de pouvoir détenir une licence StyleVision Server depuis Altova LicenseServer, StyleVision Server doit être inscrit avec LicenseServer

Pour inscrire StyleVision Server depuis l'interface de ligne de commande, utilisez la commande `licenseserver` :

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

Par exemple, si `localhost` est le nom du serveur sur lequel LicenseServer est installé :

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

Dans la commande ci-dessus, `localhost` est le nom du serveur sur lequel LicenseServer est installé. Notez également que l'emplacement du programme exécutable de StyleVision Server est :

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

Après avoir réussi l'inscription, allez à l'[onglet de gestion des clients de la page de configuration de LicenseServer](#) pour attribuer une licence à StyleVision Server.

Pour plus d'informations sur l'inscription des produits Altova avec LicenseServer, voir le [manuel utilisateur de LicenseServer](#).

### 3.3.5 Attribuer une licence (macOS)

Après avoir réussi l'inscription de StyleVision Server, elle sera recensée dans l'onglet de gestion des clients de la page de configuration de LicenseServer. Allez-y et [attribuez une licence](#) à StyleVision Server.

L'obtention de la licence des produits de serveur Altova, est basée sur le nombre de cœurs de processeurs disponibles sur la machine du produit. Par exemple, un processeur double cœur a deux cœurs, un processeur quadricœur a quatre cœurs, un processeur hexacœurs a six cœurs, etc. Le nombre de cœurs pour lesquels une licence a été délivrée pour un produit doit être supérieur ou égal au nombre de cœurs disponibles sur cette machine de serveur, que ce serveur soit une machine physique ou virtuelle. Par exemple, si un serveur a huit cœurs (un processeur octacœur), vous devrez acheter au moins une licence octacœur. Vous pouvez aussi additionner les licences pour obtenir le nombre souhaité de cœurs. Ainsi, deux licences de quadricœurs peuvent être utilisées pour un serveur octacœur au lieu d'acheter une licence octacœur.

Si vous utilisez un serveur d'ordinateur avec un grand nombre de cœurs CPU, mais ne disposez que d'un faible volume à traiter, vous pouvez aussi créer une machine virtuelle qui disposera d'un plus petit nombre de cœurs et acheter une licence pour ce nombre de cœurs. Il va de soi que la vitesse de traitement d'un tel déploiement sera moins rapide que si tous les cœurs disponibles sur le serveur étaient utilisés.

**Note** : Chaque licence de produit de serveur Altova peut être utilisée pour une seule machine client à la fois, même si la licence a une capacité de licence qui n'est pas utilisée (l'appareil client est l'appareil sur lequel le produit de serveur Altova est installé). Par exemple, si une licence de 10-cœurs est utilisée pour une machine

client qui détient 6 cœurs CPU, les 4 cœurs restants de la capacité de licence ne pourront pas être utilisés simultanément pour une autre machine client.

#### Exécution thread unique

Si un produit de serveur Altova permet une exécution single-thread, une option pour une *exécution single-thread* sera disponible. Dans ces cas, la licence produit du serveur Altova pour uniquement un cœur est disponible dans le pool des licences, un appareil avec des cœurs multiples peut être assigné à cette licence à one-core. Dans un tel cas, l'appareil exécutera ce produit en single-core. Le traitement sera donc plus lent car le multi-threading (qui est possible sur de multiples cœurs) ne sera pas disponible. Le produit sera exécuté en mode single thread sur cet appareil.

Pour assigner une licence single-core à un appareil multiple-core dans LicenseServer, sélectionnez la case à cocher *Limit to single thread execution* pour ce produit.

#### Estimation des exigences core

Il existe de nombreux facteurs externes divers qui influent sur les volumes de données et les temps de traitement que votre serveur arrive à gérer (par exemple : le matériel, la charge actuelle sur le CPU, et l'attribution de la mémoire d'autres applications exécutées sur le serveur). Afin de mesurer la performance aussi précisément que possible, testez les applications dans votre environnement avec les volumes de données et les conditions qui établissent aussi fidèlement que possible des situations professionnelles réelles.

## 3.3.6 Notes relatives à l'environnement (macOS)

### Dossiers

Ci-dessous, vous trouverez une liste de dossiers importants dans votre configuration StyleVision Server.

#### ☐ Racine d'installation

`/usr/local/Altova/StyleVisionServer2025/`

#### ☐ Fichiers de licence

`/var/Altova/StyleVisionServer`

#### ☐ Paramètres d'environnement

`/Library/LaunchDaemons/com.altova.StyleVisionServer.plist`

Le fichier des paramètres d'environnement doit être défini conformément à votre environnement spécifique. Le chemin d'exemple ci-dessus a vocation de guide général.

**Note** : Ces variables d'environnement sont uniquement définies pour le processus de StyleVision Server et n'ont pas d'incidence sur d'autres utilisateurs.

### Connexions de base de données

Sur MacOS, les connexions de base de données suivantes sont prises en charge :

- JDBC - vous pouvez utiliser JDBC pour toutes les bases de données prises en charge, à l'exception de Microsoft Access

- Connexions natives — actuellement disponibles pour les bases de données SQLite et PostgreSQL

Si vous utilisez JDBC, veuillez noter les points suivants :

- L'environnement d'exécution Java ou SDK doit être installé.
- JDBC-Connects pour la base de données cible doit être installée.
- Les variables d'environnement suivants doivent être définis correctement pour votre environnement :
  - **CLASSPATH**: pour trouver les jar-files; class path est défini dans le fichier `plist`.
  - **PATH**: pour trouver le JRE, mais n'est pas toujours nécessaire, dépendant de l'installation
  - **JAVA\_HOME**: si nécessaire, dépendant de l'installation

## Java 6 pour des formulaires PDF qui se laissent remplir

Si les parties à remplir des PDF qui se laissent remplir manquent quand le PDF est ouvert sur un système MacOS, l'une des causes probables est que Java 6 n'est pas installé sur votre appareil. Dans ce cas, vous pouvez installer Java 6 depuis [https://support.apple.com/kb/dl1572?locale=en\\_US](https://support.apple.com/kb/dl1572?locale=en_US). Si une version plus récente que Java 6 a déjà été installée, la version plus ancienne de Java 6 n'affectera pas le fonctionnement de la version plus récente, qui sera la version par défaut du système.

## 3.4 Mise à jour StyleVision Server

La manière la plus simple de reporter une licence depuis la version précédente de StyleVision Server vers la version plus récente est par le biais du processus d'installation. Les étapes clés au cours de l'installation sont :

1. Enregistrer la nouvelle version de StyleVision Server avec le serveur de licence qui détient la licence utilisée par l'ancienne version de StyleVision Server.
2. Acceptez le contrat de licence de StyleVision Server. (Si vous n'acceptez pas le contrat, la nouvelle version ne sera pas installée.)

**Note** : Si vous n'inscrivez pas StyleVision Server avec LicenseServer au cours du processus d'installation, vous pouvez faire ceci plus tard et ensuite compléter le processus de licence.

## 3.5 Migrer StyleVision Server vers un nouvel appareil

Si vous voulez migrer StyleVision Server depuis un appareil vers un autre (y compris sur des plateformes prises en charge), suivez les directives ci-dessous.

Migrer StyleVision Server vers un nouvel appareil consiste à attribuer une nouvelle fois la licence depuis l'ancien appareil vers le nouveau. Pour ce faire, procédez comme suit :

1. Installez StyleVision Server sur votre nouvel appareil. Si l'installation a déjà été réalisée en tant que partie de l'installation de FlowForce Server, ignorez cette étape.
2. Sur le nouvel appareil, enregistrez StyleVision Server avec Altova LicenseServer.
3. Sur l'ancien appareil, assurez-vous qu'aucun client n'utilise le serveur.
4. Ouvrir la page d'administration de Altova LicenseServer. Désactivez la licence de l'ancien appareil StyleVision Server et réattribuez-la au nouvel appareil.

**Note :** Si vous êtes en train d'utiliser des catalogues XML sur les anciens appareils, migrez-les sur votre nouvel appareil.

## 3.6 Exigences FOP

Par défaut, StyleVision Server utilise le processeur FO du [Projet Apache FOP](#) quand il génère des documents PDF par le biais de sa commande `generate`. (Vous pouvez sélectionner d'autres processeurs FO avec la commande `setfopath`.)

FOP aura été installé avec StyleVision Server. Sur les systèmes Windows, FOP est installé dans le dossier `c:\ProgramData\Altova\SharedBetweenVersions`. Sur les systèmes Linux et macOS, il est installé dans un dossier descendant de `styleVisionServer2025`.

Pour toute information sur l'exécution de FOP 2.9, voir la [page sur ce chapitre](#) sur le site web Apache.org. (Si vous utilisez une autre version de FOP, allez à la page correspondante de cette version FOP.)

### Environnement d'exécution de Java

Pour créer des PDF, FOP requiert l'installation de l'Environnement d'exécution Java (JRE) sur votre appareil. Vérifiez la version JRE minimum requise pour votre version FOP sur le [site web Apache.org](#). Par exemple, pour la version FOP 2.8, JRE 1.8 ou plus récent est requis. Si JRE n'est pas installé et que vous tentez d'exécuter FOP pour générer un PDF, vous recevrez a message disant qu'il y avait une erreur en appelant FOP.

#### Configurer OpenJDK

StyleVision Server a été essayé de manière régulière avec la dernière [version OpenJDK](#) disponible. Au moment de l'écriture, la dernière version est 19.0.2. Après avoir téléchargé et dézippé le pack OpenJDK, assurez-vous d'ajouter (i) le chemin `java\bin` aux chemins de système et (ii) définir la variable d'environnement `JAVA_HOME`.

Sur Windows, définissez la variable d'environnement `JAVA_HOME` comme suit :

1. Sur la barre de la tâche Windows, cliquez avec la touche de droite sur l'icône Windows et sélectionnez *System*.
2. Dans la fenêtre des Paramètres, sous *Related Settings*, cliquez sur *Advanced System Settings*.
3. Dans l'onglet *Avancé*, cliquez sur *Variables d'environnement*,
4. Cliquez Nouveau pour créer une nouvelle variable d'environnement et nommez-la `JAVA_HOME`.
5. Définir la valeur de la variable d'environnement. Par exemple, sur `c:\Programs\openjdk\bin\java.exe`, définissez la valeur comme `c:\Programs\openjdk`.

## 4 Ligne de commande StyleVision Server

### Emplacement par défaut du programme exécutable StyleVision Server

Ci-dessous, vous trouverez les emplacements par défaut du programme exécutable StyleVision Server :

<i>Linux</i>	<code>/opt/Altova/StyleVisionServer2025/bin/stylevisionserver</code>
<i>Mac</i>	<code>/usr/local/Altova/StyleVisionServer2025/bin/stylevisionserver</code>
<i>Windows</i>	<code>&lt;ProgramFilesFolder&gt;\Altova\StyleVisionServer2025\bin\StyleVisionServer.exe</code>

### Utilisation et liste de commandes CLI

La syntaxe de ligne de commande est :

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

- `--help` (short form `--h`) affiche le texte d'aide d'une commande donnée. Si aucune commande n'est nommée, alors toutes les commandes du programme d'exécution sont recensées, chacune avec une brève description de la commande.
- `--version` affiche le numéro de version de StyleVision Server.
- `<command>` est la commande à exécuter. Les commandes sont décrites dans les sous-sections de cette section (*voir liste ci-dessous*).
- `[options]` sont les options d'une commande ; ils sont listés et décrits avec leurs commandes respectives.
- `[arguments]` sont les arguments d'une commande ; ils sont listés et décrits avec leurs commandes respectives.

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

### Commandes CLI

Les commandes disponibles sont regroupées ci-dessous et sont expliquées dans les sous-sections de cette section.

- [assignlicense](#)<sup>38</sup> : Charge une licence sur LicenseServer et attribue cette licence à StyleVision Server.
- [exportresourcestrings](#)<sup>40</sup> : Exporte tous les strings de ressource d'application vers un fichier XML.
- [generate](#)<sup>42</sup> : Génère un ou plusieurs documents depuis un fichier XML d'entrée et une feuille de style XSLT dans le fichier PXF d'entrée.

- [help](#)<sup>47</sup> : Affiche des informations concernant la commande soumise dans l'argument (ou concernant toutes les commandes si aucun argument n'est soumis).
- [licenseserver](#)<sup>48</sup> : Enregistre StyleVision Server avec un LicenseServer sur le réseau local.
- [pdfdata](#)<sup>50</sup> : Génère des données de formulaires depuis un fichier PDF vers un fichier FDF ou XML.
- [setdeflang](#)<sup>52</sup> : Définit la langue par défaut de StyleVision Server.
- [setfopath](#)<sup>53</sup> : Sélectionne un processeur FO alternatif pour une génération PDF ultérieure.
- [verifylicense](#)<sup>55</sup> : Contrôle si StyleVision Server actuel détient une licence ; en option, contrôle si une clé de licence donnée est attribuée.
- [version](#)<sup>57</sup> : Affiche la version de StyleVision Server.

## 4.1 accepteula (Linux uniquement)

### Syntaxe et description

Afin de pouvoir exécuter StyleVision Server, le contrat de licence de l'utilisateur final de l'application (EULA) doit être accepté. Vous pouvez accepter le contrat EULA de l'application en exécutant la commande `accepteula`.

Cette commande est utile, par exemple, si vous voulez mettre sous licence et exécutez StyleVision Server directement via les processus automatisés qui utilisent les scripts.

```
stylevisionserver accepteula [options]
```

- La commande fonctionne uniquement pour les produits de serveur Altova qui ont été installés sur des machines Linux.
- Vous devez inscrire StyleVision Server avec LicenseServer avant d'exécuter la commande `accepteula`.
- Utiliser l'option `--h, --help` pour afficher les informations concernant la commande.
- Utiliser la minuscule `stylevisionserver`.
- Utiliser la barre oblique sur Linux.

### Exemples

Exemples de la commande `accepteula` :

```
stylevisionserver accepteula
```

### Options

Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

## 4.2 assignlicense

### Syntaxe et description

La commande `assignlicense` télécharge un fichier de licence vers Altova LicenseServer avec lequel StyleVision Server est enregistré (voir la commande `licenseserver`), et attribue la licence à StyleVision Server. Elle prend le chemin d'un fichier de licence en tant que son argument. La commande vous permet aussi de tester la validité d'une licence.

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

- L'argument `FILE` prend le chemin du fichier de licence.
- L'option `--test-only` charge le fichier de licence sur LicenseServer et valide la licence, mais n'attribue pas la licence à StyleVision Server.

Pour plus de détails concernant la licence, voir la documentation LicenseServer (<https://www.altova.com/manual/fr/licenseserver/3.17/>).

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionServer` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

#### ▼ Barre oblique inversée et des espaces sur des systèmes Windows

Dans les systèmes Windows : lorsque des espaces se produisent (par exemple, dans les noms de fichier ou de dossier, ou les noms d'entreprise, de personne ou de produit), utiliser des guillemets : par exemple, "`My File`". Veuillez noter, néanmoins qu'une barre oblique inversée suivie par un double guillemet (par exemple, "`C:\My directory`") peut ne pas être lue correctement. Cela est dû au fait que le caractère de barre oblique inversée est également utilisé pour indiquer le début d'une séquence d'échappement, et la séquence d'échappement `\` représente le caractère de marque de double guillemet. Si vous souhaitez échapper cette séquence des caractères, utiliser une barre oblique inversée précédente comme ceci : `\\`. Pour résumer : Si vous souhaitez écrire un chemin de fichier qui contient des espaces et des barre oblique inversée de fin, l'écrire comme ceci : "`C:\My Directory\\`".

### Exemples

Des exemples de la commande `assignlicense` :

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

- La première commande ci-dessus charge la licence spécifiée sur LicenseServer et l'attribue à

StyleVision Server.

- La troisième commande charge la licence spécifiée sur LicenseServer et la valide, sans l'attribuer à StyleVision Server.

## Options

Les options sont recensées dans la forme abrégée (si disponible) et dans la forme longue. Vous pouvez utiliser un ou deux tirets pour les formes abrégées et longues. Une option peut ou ne peut pas prendre une valeur. Si elle prend une valeur, elle est écrite comme ceci : `--option=value`. Des valeurs peuvent être spécifiées sans guillemets sauf dans deux cas : (i) si le string valeur contient des espaces, ou (ii) s'il est explicitement indiqué dans la description que des guillemets sont exigés. Si une option prend une valeur booléenne et qu'aucune valeur n'est spécifiée, alors la valeur par défaut de l'option est `TRUE`. Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

### ▼ test-only [t]

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

Les valeurs sont `true|false`. Si `true`, alors le fichier de licence est chargé sur LicenseServer et validé, mais pas attribué.

## 4.3 exportresourcestrings

### Syntaxe et description

La commande `exportresourcestrings` sort un fichier XML contenant les strings de ressource de l'application StyleVision Server dans la langue spécifiée. Les langues d'exportation disponibles sont allemand (de), anglais (en), espagnol (es), français (fr) et japonais (ja).

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

- L'argument *LanguageCode* donne la langue des strings de ressource dans le fichier XML de sortie ; il s'agit de la langue d'exportation *export language*. Les langues d'exportation autorisées (avec leurs codes de langue respectifs entre parenthèses) sont : allemand (de), anglais (en), espagnol (es), français (fr) et japonais (ja).
- L'argument *XMLOutputFile* spécifie le chemin et le nom du fichier XML de sortie .

Pour créer des localisations, veuillez noter les points suivants.

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plateformes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

#### ▼ Barre oblique inversée et des espaces sur des systèmes Windows

Dans les systèmes Windows : lorsque des espaces se produisent (par exemple, dans les noms de fichier ou de dossier, ou les noms d'entreprise, de personne ou de produit), utiliser des guillemets : par exemple, "My File". Veuillez noter, néanmoins qu'une barre oblique inversée suivie par un double guillemet (par exemple, "C:\My directory\"") peut ne pas être lue correctement. Cela est dû au fait que le caractère de barre oblique inversée est également utilisé pour indiquer le début d'une séquence d'échappement, et la séquence d'échappement \" représente le caractère de marque de double guillemet. Si vous souhaitez échapper cette séquence des caractères, utiliser une barre oblique inversée précédente comme ceci : \\\". Pour résumer : Si vous souhaitez écrire un chemin de fichier qui contient des espaces et des barre oblique inversée de fin, l'écrire comme ceci : "C:\My Directory\\".

### Exemples

Exemples de la commande `exportresourcestrings` :

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

- La commande ci-dessus crée un fichier appelé `Strings.xml` sous `c:\` qui contient les strings de ressource de StyleVision Server en allemand.

## Créer des versions localisées de StyleVision Server

Vous pouvez créer une version localisée de StyleVision Server pour toute langue de votre choix. Cinq versions localisées (allemand, anglais, espagnol, français et japonais) sont déjà disponibles sous le dossier `C:\Program Files (x86)\Altova\StyleVisionServer2025\bin`, et n'ont donc pas besoin d'être créées.

`\Program Files (x86)\Altova\StyleVisionServer2025\bin`, et n'ont donc pas besoin d'être créées.

Créer une version localisée comme suit :

1. Générer un fichier XML contenant des strings de ressource en utilisant la commande `exportresourcestrings` (voir la *syntaxe de commande ci-dessus*). Les strings de ressource dans ce fichier XML seront ceux d'une des cinq langues prises en charge : allemand (`de`), anglais (`en`), espagnol (`es`), français (`fr`) ou japonais (`ja`), conformément à l'argument `LanguageCode` utilisé avec la commande.
2. Traduire les strings de ressource à partir de l'une des cinq langues prises en charge dans la langue cible. Les strings de ressource sont les contenus des éléments `<string>` dans le fichier XML. Ne pas traduire les variables dans les accolades comme `{option}` ou `{product}`.
3. Contactez [Altova Support](#) pour générer un fichier DLL StyleVision Server localisé à partir de votre fichier XML traduit.
4. Après avoir reçu votre fichier DLL localisé de [Altova Support](#), enregistrez le fichier DLL sous le dossier `C:\Program Files (x86)\Altova\StyleVisionServer2025\bin`. Votre fichier DLL aura un nom de la forme `StyleVisionServer2025_lc.dll`. La partie `_lc` du nom contient le code de langue. Par exemple, dans `StyleVisionServer2025_de.dll`, la partie `de` est le code de langue pour l'allemand (Deutsch).
5. Exécutez la commande `setdeflang` pour définir votre fichier DLL localisé DLL en tant qu'application StyleVision Server à utiliser. Pour l'argument de la commande `setdeflang`, utilisez le code de langue qui fait partie du nom DLL.

**Note** : Altova StyleVision Server est livré avec une prise en charge pour les cinq langues : allemand, anglais, espagnol, français et japonais. Donc vous ne devez pas créer de version localisée pour ces langues. Pour définir une de ces langues en tant que langue par défaut, utilisez la commande StyleVision Server de `setdeflang`.

## 4.4 generate

### Syntaxe et description

La commande `generate` (la forme abrégée est `gen`) génère un ou plusieurs fichiers de sortie (HTML, PDF, RTF, et/ou DOCX) en transformant le fichier d'entrée XML à l'aide du document XSLT contenu dans le fichier d'entrée PXF

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

- L'option `--inputxml` est obligatoire ; elle indique le chemin vers le fichier XML.
- L'argument `InputPXF` spécifie le chemin vers le fichier PXF qui contient le document XSLT qui sera utilisé pour générer les documents de sortie. Les fichiers PXF sont créés avec l'[application Altova StyleVision](#).
- Chaque format de sortie est généré en spécifiant une option pour cette sortie (voir la liste d'Options ci-dessous). La valeur de chaque option est un chemin qui spécifie l'endroit où la sortie sera générée.

**Note** : StyleVision Server utilise [Apache FOP](#), le processeur FO du Projet Apache, pour générer des fichiers PDF depuis FO. Apache FOP est installé avec StyleVision Server à l'emplacement suivant : sur les systèmes Windows, `ProgramData\Altova\SharedBetweenVersions`; sur les systèmes Linux et macOS, dans un fichier descendant du dossier `StyleVisionServer2025`. Veuillez noter que Apache FOP nécessite **Java Runtime Environment 1.6 ou plus** à installer sur l'appareil StyleVision Server. En ce qui concerne StyleVision Server 32-bit, installer Java 32-bit ; pour StyleVision Server 64-bit, installer Java 64-bit. Pour plus d'information sur la configuration de FOP, voir le chapitre [Exigences FOP](#) <sup>34</sup>.

#### ▼ AltovaFOPServer pour une génération de PDF plus rapide

Si vous allez exécuter de multiples transformations au PDF, en particulier des documents volumineux, vous allez vouloir utiliser le AltovaFOPServer pour exécuter des tâches plus rapidement. L'accélération des tâches apparaît car AltovaFOPServer maintient les fichiers Java VM et FOP chargés, économise donc du temps pour charger ces fichiers pour chaque transformation. AltovaFOPServer sera installé avec votre pack de StyleVision. Afin de pouvoir l'utiliser pour des transformations, vous devez d'abord démarrer AltovaFOPServer via la ligne de commande.

#### Start AltovaFOPServer

La syntaxe de la commande pour démarrer AltovaFOPServer est :

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

Un exemple de commande CLI pour démarrer AltovaFOPServer :

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

L'argument `classPath` doit contenir des chemins dans les fichiers `gson.jar` et `AltovaFOPServer.jar` files. Saisissez les emplacements corrects de ces fichiers.

#### Appel AltovaFOPServer dans StyleVision Server

Une fois que AltovaFOPServer a été démarré, vous pouvez l'appeler depuis StyleVision Server en utilisant la commande `generate` avec soit l'option `altova-fopserver-pid-file` ou `altova-fopserver-port` (voir *Options ci-dessous*).

▼ Casse et barres obliques sur la ligne de commande

`styleVisionServer` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

▼ Barre oblique inversée et des espaces sur des systèmes Windows

Dans les systèmes Windows : lorsque des espaces se produisent (par exemple, dans les noms de fichier ou de dossier, ou les noms d'entreprise, de personne ou de produit), utiliser des guillemets : par exemple, "My File". Veuillez noter, néanmoins qu'une barre oblique inversée suivie par un double guillemet (par exemple, "C:\My directory\" ) peut ne pas être lue correctement. Cela est dû au fait que le caractère de barre oblique inversée est également utilisé pour indiquer le début d'une séquence d'échappement, et la séquence d'échappement `\"` représente le caractère de marque de double guillemet. Si vous souhaitez échapper cette séquence des caractères, utiliser une barre oblique inversée précédente comme ceci : `\"`. Pour résumer : Si vous souhaitez écrire un chemin de fichier qui contient des espaces et des barre oblique inversée de fin, l'écrire comme ceci : `"C:\My Directory\"`.

## Exemples

Des exemples de la commande `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
```

- La commande ci-dessus contient l'option `--inputxml` obligatoire, l'argument `InputPXF` (`Test.pxf`) et un minimum d'une option output-creation (`--html` dans tous les exemples ci-dessus).
- Le fichier XML d'entrée à utiliser peut être situé dans le fichier PXF (voir le deuxième et troisième exemple ci-dessus) ou il peut être un fichier XML externe (situé en-dehors du fichier PXF ; voir le troisième et quatrième exemple ci-dessus).
- L'interrupteur `--inputxml` reste ignoré si la source de schéma principale est BD ou BD-XML, mais il doit y figurer pour des raisons de syntaxe et vous devriez utiliser quelque chose du genre `--inputxml=database`.

- Si l'option `output-creation --html` prend un chemin relatif, comme illustré dans l'exemple ci-dessus, l'emplacement du fichier de sortie sera relatif par rapport au dossier dans lequel le fichier PXF se trouve.

## Options

### inputxml [xml]

`--xml, --inputxml = PathToXMLFile`

Cette option est obligatoire. Elle spécifie le chemin vers le fichier XML à traiter. Le fichier XML peut être situé à l'intérieur ou à l'extérieur du fichier PXF. Pour cibler des fichiers XML dans un fichier PXF, utiliser le `|zip` locator (voir les parties marquées dans les exemples ci-dessous). L'option `--inputxml` est ignorée si la source de schéma principale est DB ou DB-XML.

### dbwhere [dbw]

`--dbw, --dbwhere = WHEREClause`

Une clause `WHERE` qui détermine quelles lignes d'une source DB-XML doivent être traitées.

### param [p]

`--p, --param = $ParamName:ParamValue`

Attribue une valeur au paramètre défini dans le fichier PXF. L'interrupteur `--param` doit être utilisé avant chaque paramètre. Utiliser des guillemets si `ParamName` ou `ParamValue` contient un espace. Par exemple : `--p=$company:"Nanonull Inc"`

### prohibit-output-outside-target-folder

`--prohibit-output-outside-target-folder = true|false`

Les valeurs sont `true|false`. Si `true`, ne permet pas la création d'une sortie dans un autre fichier que celui dans lequel le fichier de sortie principal (HTML, PDF, RTF, DOCX, FO) est créé. Cela permet de protéger les autres dossiers, le cas échéant. Le réglage par défaut est `false`.

### outhtml [html]

`--html, --outhtml = FilePath`

Le chemin vers le fichier HTML à générer.

### outpdf [pdf]

`--pdf, --outpdf = FilePath`

Le chemin vers le fichier PDF à générer.

### outrtf [rtf]

`--rtf, --outrtf = FilePath`

Le chemin vers le fichier RTF à générer.

### outdocx [docx]

`--docx, --outdocx = FilePath`

Le chemin vers le fichier DOCX à générer.

### outtext [text]

`--text, --outtext = FilePath`

Chemin vers le fichier texte à générer.

#### ▼ outfo [fo]

`--fo, --outfo = FilePath`

Le chemin vers le fichier FO à générer.

#### ▼ generate-html-output-as-mime

`--generate-html-output-as-mime = true|false`

Les valeurs sont `true|false`. Si l'option n'est pas spécifiée, le réglage par défaut est `false`. Si l'option est spécifiée sans valeur, le réglage par défaut est `true`. Si `true`, la sortie HTML est générée en tant que mime stream.

#### ▼ altova-fopserver-pid-file

`--altova-fopserver-pid-file = PathToFile`

Le chemin vers le fichier PID à utiliser pour se connecter à un AltovaFOPServer. Vous pouvez vous connecter à AltovaFOPServer soit par le biais du fichier PID du serveur, soit par un port. L'option PID est une alternative au port de l'option (*voir prochaine option*). Pour plus d'information comment utiliser AltovaFOPServer, voir la description générale de la commande `generate` ci-dessus.

#### ▼ altova-fopserver-port

`--altova-fopserver-port = Value`

Un numéro de port est une liste de numéros de ports à utiliser pour se connecter au AltovaFOPServer exécuté. Le numéro de port dans une liste est séparée par des virgules. À la place d'un seul numéro dans la liste, vous pouvez en utiliser une variété. Donc vous pouvez soumettre `9090-9100,9999` comme valeur de cette option. Notez que vous pouvez vous connecter au AltovaFOPServer soit par le biais du fichier PID du AltovaFOPServer, soit d'un port. L'option du port est une alternative à l'option PID (*voir l'option précédente*). Pour plus d'information comment utiliser AltovaFOPServer, voir la description générale de la commande `generate` ci-dessus.

#### ▼ taxonomy-package

`--taxonomy-package = FilePath`

Le chemin vers un pack de taxonomie supplémentaire. Ajouter l'option de multiples fois pour spécifier plus d'un pack de taxonomie.

#### ▼ taxonomy-packages-config-file

`--taxonomy-packages-config-file = FilePath`

Le chemin vers le fichier `TaxonomyPackagesConfig.json`, qui est un fichier commun utilisé par les produits d'Altova XMLSpy, MapForce et StyleVision pour configurer les taxonomies XBRL à utiliser parmi ces produits.

#### ▼ verbose [v]

`--v, --verbose = true|false`

Les valeurs sont `true|false`. L'affichage de tous les messages peut être allumé ou éteint, respectivement. Le réglage par défaut est `false` si l'option n'est pas fournie ou `true` si l'option est fournie sans valeur.

#### ▼ lang [l]

```
--l, --lang = en|de|es|fr|ja
```

La langue utilisée pour afficher les messages .

Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

Les options sont recensées dans la forme abrégée (si disponible) et dans la forme longue. Vous pouvez utiliser un ou deux tirets pour les formes abrégées et longues. Une option peut ou ne peut pas prendre une valeur. Si elle prend une valeur, elle est écrite comme ceci : `--option=value`. Des valeurs peuvent être spécifiées sans guillemets sauf dans deux cas : (i) si le string valeur contient des espace, ou (ii) s'il est explicitement indiqué dans la description que des guillemets sont exigés. Si une option prend une valeur booléenne et qu'aucune valeur n'est spécifiée, alors la valeur par défaut de l'option est `TRUE`. Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

## Catalogs

Si vous utilisez le mécanisme de catalogue d'Altova, vous pouvez trouver les fichiers de catalogue pertinents dans le dossier `etc` du dossier d'application StyleVision Server. Pour des informations détaillées, voir la [section Catalogues du manuel de StyleVision d'Altova](#).

Vous pouvez créer `CustomCatalog.xml` depuis le fichier modèle `CustomCatalog_template.xml`. Assurez-vous de renommer le modèle de fichier en `CustomCatalog.xml` étant donné que ce dernier fichier sera le fichier utilisé dans le mécanisme de catalogue (pas le modèle de fichier).

Veillez noter les points suivants :

- Lors d'une nouvelle installation de la même version majeure (mêmes versions ou différentes versions mineures), le modèle de fichier sera remplacé par un nouveau modèle de fichier, mais `CustomCatalog.xml` ne sera pas touché.
- Toutefois, si vous installez une nouvelle version majeure au-dessus de la version majeure précédente, alors le dossier de la version majeure précédente sera supprimé - ensemble avec son `CustomCatalog.xml`. Donc, si vous voulez continuer à utiliser `CustomCatalog.xml`, assurez-vous d'enregistrer `CustomCatalog.xml` depuis le dossier de la version majeure à un endroit sûr. Une fois que la nouvelle version majeure a été installée, vous pouvez copier `CustomCatalog.xml` que vous avez enregistré dans le dossier `etc` de la nouvelle version majeure et l'éditer là tel que requis.

## 4.5 help

### Syntaxe et description

La commande `help` prend un seul argument (`Command`), qui est le nom de la commande pour laquelle l'aide est requise. Elle affiche la syntaxe de la commande, ses options, et d'autres informations pertinentes. Si l'argument `Command` n'est pas spécifié, toutes les commandes du programme d'exécution sont recensés, chacun présentant une brève description de texte.

```
stylevisionserver help Command
```

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionServer` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

### Exemple

Exemple de la commande `help` pour afficher des informations concernant la commande `licenseserver` :

```
stylevisionserver help licenseserver
```

### L'option `--help`

L'information Help à propos d'une commande est également disponible en utilisant l'option `--help` avec la commande pour laquelle l'information d'aide est nécessaire. Les deux commandes ci-dessous produisent les mêmes résultats :

```
stylevisionserver licenseserver --help
```

La commande ci-dessus utilise l'option `--help` de la commande `licenseserver`.

```
stylevisionserver help licenseserver
```

La commande `help` prend `licenseserver` en tant que son argument.

Les deux commandes affichent des informations d'aide concernant la commande `licenseserver`.

## 4.6 licenseserver

### Syntaxe et description

La commande `licenseserver` enregistre StyleVision Server avec Altova LicenseServer spécifié par l'argument *Server-Or-IP-Address*. Pour pouvoir exécuter la commande `licenseserver` correctement, les deux serveurs (StyleVision Server et LicenseServer) doivent être connectés sur le réseau et le LicenseServer doit fonctionner. Vous devez posséder des privilèges d'administrateur pour pouvoir enregistrer StyleVision Server auprès du LicenseServer.

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

- L'argument *Server-Or-IP-Address* prend le nom ou l'adresse IP de l'appareil de LicenseServer.

Une fois que StyleVision Server a été enregistré avec succès auprès de LicenseServer, vous recevrez un message. Le message affichera aussi l'URL du LicenseServer. Vous pouvez maintenant vous rendre sur LicenseServer pour attribuer une licence à StyleVision Server. Pour plus de détails concernant la licence, voir la documentation LicenseServer (<https://www.altova.com/manual/fr/licenseserver/3.17/>).

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

#### ▼ Barre oblique inversée et des espaces sur des systèmes Windows

Dans les systèmes Windows : lorsque des espaces se produisent (par exemple, dans les noms de fichier ou de dossier, ou les noms d'entreprise, de personne ou de produit), utiliser des guillemets : par exemple, "`My File`". Veuillez noter, néanmoins qu'une barre oblique inversée suivie par un double guillemet (par exemple, "`C:\My directory\`") peut ne pas être lue correctement. Cela est dû au fait que le caractère de barre oblique inversée est également utilisé pour indiquer le début d'une séquence d'échappement, et la séquence d'échappement `\` représente le caractère de marque de double guillemet. Si vous souhaitez échapper cette séquence des caractères, utiliser une barre oblique inversée précédente comme ceci : `\`  
`\`". Pour résumer : Si vous souhaitez écrire un chemin de fichier qui contient des espaces et des barre oblique inversée de fin, l'écrire comme ceci : "`C:\My Directory\`".

### Exemples

Exemples de la commande `licenseserver` :

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

Les commandes ci-dessus spécifient, respectivement, la machine nommée `DOC.altova.com`, et la machine de l'utilisateur (`localhost` et `127.0.0.1`) qui fait marcher Altova LicenseServer. Dans tous les cas, la commande enregistre StyleVision Server avec le LicenseServer sur la machine spécifiée. La dernière commande appelle le programme d'exécution de serveur pour exécuter la commande.

## Options

Les options sont recensées dans la forme abrégée (si disponible) et dans la forme longue. Vous pouvez utiliser un ou deux tirets pour les formes abrégées et longues. Une option peut ou ne peut pas prendre une valeur. Si elle prend une valeur, elle est écrite comme ceci : `--option=value`. Des valeurs peuvent être spécifiées sans guillemets sauf dans deux cas : (i) si le string valeur contient des espaces, ou (ii) s'il est explicitement indiqué dans la description que des guillemets sont exigés. Si une option prend une valeur booléenne et qu'aucune valeur n'est spécifiée, alors la valeur par défaut de l'option est `TRUE`. Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

### ▼ json [j]

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

Les valeurs sont `true|false`. Si `true`, imprime le résultat de la tentative d'enregistrement en tant qu'objet JSON parsable par machine.

## 4.7 pdfdata

### Syntaxe et description

La commande `pdfdata` génère un fichier FDP ou un fichier PDF qui est soumis comme argument *InputPDF*.

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

- L'argument *InputPDF* précise le chemin vers le fichier PDF, duquel le fichier de sortie FDF ou XML sera généré. Si le fichier PDF ne contient pas de données de formulaires, le fichier généré ne contiendra pas de données de formulaires.
- Utiliser l'option `--outfdf` pour préciser l'emplacement du fichier FDF généré ou l'option `--outxml` pour préciser l'emplacement du fichier XML généré.

Pour information sur la configuration FOP, que StyleVision Server utilise par défaut pour générer le PDF, voir le chapitre [Exigences FOP](#) <sup>34</sup>.

Pour plus d'information sur les fichiers FDF et sur la conception des formulaires PDF remplissables PDF, voir la documentation [Altova StyleVision \(Enterprise Edition\)](#).

#### ▼ Casing and slashes on the command line

`styleVisionServer` on Windows

`stylevisionserver` on Windows and Unix (Linux, Mac)

\* Note that lowercase (`stylevisionserver`) works on all platforms (Windows, Linux, and Mac), while upper-lower (`StyleVisionServer`) works only on Windows and Mac.

\* Use forward slashes on Linux and Mac, backslashes on Windows.

#### ▼ Backslashes, spaces, and special characters on Windows systems

On Windows systems: When spaces or special characters occur in strings (for example in file or folder names, or company, person or product names), use quotes: for example, "**My File**". Note, however, that a backslash followed by a double-quotation mark (for example, "**C:\My directory\**") might not be read correctly. This is because the backslash character is also used to indicate the start of an escape sequence, and the escape sequence `\"` stands for the double-quotation mark character. If you want to escape this sequence of characters, use a preceding backslash, like this: `\\"`. To summarize: If you need to write a file path that contains spaces or an end backslash, write it like this: "**C:\My Directory\**  
**\**".

### Exemples

Exemples de la commande `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
```

Les exemples créés ci-dessus, respectivement, et le fichier FDP et le fichier XML depuis la même entrée PDF.

## Options

### ▼ outfdf

`--outfdf = FilePath`

Le chemin menant au fichier FDP généré.

### ▼ outxml

`-- outxml = FilePath`

Le chemin menant au fichier XML généré.

### ▼ verbose[v]

`--verbose = true|false`

Les valeurs sont `true|false`. Active ou désactive respectivement l'affichage de tous les messages. Le défaut est `false` si l'option n'est pas fournie, `true` si fournie sans valeur.

Use the `--h, --help` option to display information about the command.

Options are listed in short form (if available) and long form. You can use one or two dashes for both short and long forms. An option may or may not take a value. If it takes a value, it is written like this: `--option=value`. Values can be specified without quotes except in two cases: (i) when the value string contains spaces, or (ii) when explicitly stated in the description of the option that quotes are required. If an option takes a Boolean value and no value is specified, then the option's default value is `TRUE`. Use the `--h, --help` option to display information about the command.

## 4.8 setdeflang

### Syntaxe et description

La commande `setdeflang` (la forme abrégée est `sdl`) définit la langue par défaut de StyleVision Server. Les langues disponibles sont anglais (`en`), allemand (`de`), espagnol (`es`), français (`fr`) et japonais (`ja`). La commande prend un argument `LanguageCode` obligatoire.

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

- L'argument `LanguageCode` est requis et définit la langue par défaut de StyleVision Server. Les valeurs respectives à utiliser sont : `en`, `de`, `es`, `fr`, `ja`.
- Utiliser l'option `--h`, `--help` pour afficher les informations concernant la commande.

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

### Exemples

Exemples de la commande `setdeflang` (`sdl`) :

```
stylevisionserver sdl de
stylevisionserver setdeflang es
```

- La première commande définit le langage par défaut de StyleVision Server en allemand.
- La seconde commande définit le langage par défaut de StyleVision Server en espagnol.

### Options

Utiliser l'option `--h`, `--help` pour afficher des informations à propos de la commande.

## 4.9 setfopath

### Syntaxe et description

La commande `setfopath` (la forme abrégée est `sfp`) spécifie le chemin vers un processeur Apache FOP différent de celui inclut dans le paquet StyleVision Server.

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

- Par défaut, le processeur Apache FOP compris dans StyleVision Server est utilisé pour le traitement des documents FO et pour générer des sorties PDF. Si vous souhaitez utiliser d'autres instances du processeur Apache FOP que le processeur fourni avec StyleVision Server, utiliser la commande `setfopath` avec l'Argument `Path` qui donne le chemin au processeur FO que vous souhaitez utiliser.
- Après qu'un processeur FO ait été spécifié avec la commande `setfopath`, c'est ce processeur qui sera utilisé lorsque le PDF est généré avec les commandes `generate`<sup>42</sup> suivantes. Pour changer à nouveau de processeurs, utiliser à nouveau la commande `setfopath`. Pour repasser au processeur FOP de StyleVision Server, trouver le dossier FOP dans votre système et utiliser ce chemin en tant qu'argument de `setfopath`.
- Sur les systèmes Windows, le dossier FOP qui a été installé avec StyleVision Server sera situé sous `ProgramData\Altova\SharedBetweenVersions`; sur les systèmes Linux et macOS, dans un dossier descendant de `StyleVisionServer2025`.

Pour des informations sur la configuration de FOP, que StyleVision Server utilise par défaut pour générer des PDF, voir le chapitre [Exigences FOP](#)<sup>34</sup>.

Pour plus d'informations concernant les fichiers FDF et la conception des formulaires PDF remplissables, voir la documentation [Altova StyleVision \(Enterprise Edition\)](#).

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plateformes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

#### ▼ Barre oblique inversée et des espaces sur des systèmes Windows

Dans les systèmes Windows : lorsque des espaces se produisent (par exemple, dans les noms de fichier ou de dossier, ou les noms d'entreprise, de personne ou de produit), utiliser des guillemets : par exemple, "`My File`". Veuillez noter, néanmoins qu'une barre oblique inversée suivie par un double guillemet (par exemple, "`C:\My directory\"`") peut ne pas être lue correctement. Cela est dû au fait que le caractère de barre oblique inversée est également utilisé pour indiquer le début d'une séquence d'échappement, et la séquence d'échappement `\` représente le caractère de marque de double guillemet. Si vous souhaitez échapper cette séquence des caractères, utiliser une barre oblique inversée précédente comme ceci : `\ \`. Pour résumer : Si vous souhaitez écrire un chemin de fichier qui contient des espaces et des barre oblique inversée de fin, l'écrire comme ceci : "`C:\My Directory\`".

## Exemples

Après avoir exécuté la commande `setfopath`, vous pouvez utiliser la commande `generate`<sup>42</sup> pour générer un fichier de sortie PDF en utilisant le processeur FO que vous venez de spécifier.

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

Les commandes suivantes fonctionnent de la manière suivante :

1. La commande `setfopath` spécifie que le processeur FO à l'emplacement `C:\FOP\FOP.bat` doit être utilisé pour générer un PDF dans les commandes suivantes de génération de PDF.
2. La commande `generate` génère un fichier PDF depuis le XML d'entrée spécifié, en utilisant les fichiers de transformation contenus dans le fichier PXF. Le processeur FO spécifié dans la commande précédente est utilisé pour générer le PDF.

## Options

Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

Les options sont recensées dans la forme abrégée (si disponible) et dans la forme longue. Vous pouvez utiliser un ou deux tirets pour les formes abrégées et longues. Une option peut ou ne peut pas prendre une valeur. Si elle prend une valeur, elle est écrite comme ceci : `--option=value`. Des valeurs peuvent être spécifiées sans guillemets sauf dans deux cas : (i) si le string valeur contient des espaces, ou (ii) s'il est explicitement indiqué dans la description que des guillemets sont exigés. Si une option prend une valeur booléenne et qu'aucune valeur n'est spécifiée, alors la valeur par défaut de l'option est `TRUE`. Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

## 4.10 verifylicense

### Syntaxe et description

La commande `verifylicense` contrôle si le produit actuel est mis sous licence. De plus, l'option `--license-key` vous permet de vérifier si une clé de licence spécifique est assignée au produit.

```
stylevisionserver verifylicense [options]
```

- Pour contrôler si une licence spécifique est attribuée à StyleVision Server, fournir la clé de licence en tant que la valeur de l'option `--license-key`.

Pour plus de détails concernant la licence, voir la documentation LicenseServer (<https://www.altova.com/manual/fr/licenseserver/3.17/>).

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionserver` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plateformes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

### Exemples

Exemple de la commande `verifylicense` :

```
stylevisionserver verifylicenseserver
stylevisionserver verifylicenseserver --license-key=ABCD123-ABCD123-ABCD123-ABCD123-ABCD123-ABCD123
```

- La première commande contrôle si StyleVision Server est mis sous licence.
- La seconde commande contrôle si StyleVision Server est mis sous licence avec la clé de licence spécifiée avec l'option `--license-key`.

### Options

Les options sont recensées dans la forme abrégée (si disponible) et dans la forme longue. Vous pouvez utiliser un ou deux tirets pour les formes abrégées et longues. Une option peut ou ne peut pas prendre une valeur. Si elle prend une valeur, elle est écrite comme ceci : `--option=value`. Des valeurs peuvent être spécifiées sans guillemets sauf dans deux cas : (i) si le string valeur contient des espace, ou (ii) s'il est explicitement indiqué dans la description que des guillemets sont exigés. Si une option prend une valeur booléenne et qu'aucune valeur n'est spécifiée, alors la valeur par défaut de l'option est `TRUE`. Utiliser l'option `--h, --help` pour afficher des informations à propos de la commande.

#### ▼ license-key [l]

```
--l, --license-key = Value
```

Contrôle si StyleVision Server est mis sous licence avec la clé de licence spécifiée en tant que la valeur de cette option.

## 4.11 version

### Syntaxe et description

La commande `version` affiche le numéro de version de StyleVision Server.

```
stylevisionserver version
```

#### ▼ Casse et barres obliques sur la ligne de commande

`stylevisionServer` sur Windows

`stylevisionserver` sur Windows et Unix (Linux, Mac)

\* Veuillez noter que la casse minuscule (`stylevisionserver`) fonctionne sur toutes les plate-formes (Windows, Linux et Mac), alors que la casse majuscule (`StyleVisionServer`) fonctionne uniquement sur Windows et Mac.

\* Utiliser des barres obliques basculée vers l'avant sur Linux et Mac, des barres obliques basculées vers l'arrière sur Windows.

### Exemple

Exemple de la commande `version` :

```
stylevisionserver version
```

## 5 StyleVision Server API

StyleVision Server propose une interface de programmation d'application (API) auquel vous pouvez accéder par programme depuis votre code sur base .NET, COM, ou Java.

Cette section de référence est organisée comme suit :

- [À propos de l'interface .NET](#) <sup>59</sup>
- [À propos de l'interface COM](#) <sup>60</sup>
- [À propos de l'interface Java](#) <sup>61</sup>
- [Exemples de code](#) <sup>63</sup>
- [Référence API](#) <sup>71</sup>

## 5.1 À propos de l'interface .NET

L'interface .NET est construite en tant que wrapper autour de l'interface COM. Elle est fournie en tant qu'assembly interop primaire signée par Altova ; elle utilise l'espace de noms `Altova.StyleVisionServer`.

Pendant l'installation, StyleVision Server sera enregistré automatiquement en tant qu'objet de serveur COM, il n'y a donc aucune nécessité d'une inscription manuelle. Si vous tombez sur une erreur d'accès, ouvrez les Services de composant et donnez les permissions au même compte qui exécute le pool d'application contenant StyleVision Server.

Afin de pouvoir utiliser StyleVision Server dans votre projet .NET, ajouter une référence au fichier `Altova.StyleVisionServer.dll` (voir les instructions ci-dessous). L'`Altova.StyleVisionServer.dll` est situé dans le dossier `bin` du dossier d'installation StyleVision Server. Ce fichier `.dll` est ajouté automatiquement au global assembly cache (GAC) pendant l'installation de StyleVision Server (le GAC est généralement situé dans le dossier `C:\WINDOWS\assembly` ).

Une fois que StyleVision Server a été enregistré en tant qu'objet de serveur COM, et l'`Altova.StyleVisionServer.dll` est disponible sur l'interface .NET, la fonction API StyleVision Server devient disponible dans votre projet .NET.

**Note** : Si vous avez installé une version StyleVision Server 64-bit, alors la version 32-bit de `Altova.StyleVisionServer.dll` sera située dans le dossier `bin\API_32bit`. De la même manière, si vous avez installé StyleVision Server 32-bit, alors les fichiers de version 64-bit de `Altova.StyleVisionServer.dll` seront situés dans le dossier `bin\API_64bit`.

### Pour ajouter une référence à StyleVision Server DLL dans un projet Visual Studio .NET

1. Le projet .NET ouvert dans Visual Studio, cliquer sur **Projet | Ajouter Référence**. Le dialogue Add Reference s'ouvre.
2. Dans l'onglet Browse, chercher le dossier : `<StyleVisionServer dossier d'application>/bin`, sélectionnez `Altova.StyleVisionServer.dll`, puis cliquez sur **OK**.

Vous pouvez consulter la structure de l'assembly `Altova.StyleVisionServer` avec le Visual Studio Object Browser (pour afficher l'Object Browser, cliquez sur **Object Browser** dans le menu **View**).

## 5.2 À propos de l'interface COM

StyleVision Server est enregistré automatiquement en tant qu'objet de serveur COM pendant l'installation. Pour vérifier si l'enregistrement a réussi, ouvrir l'éditeur de registre (par exemple, en saisissant la commande `regedit.exe` dans la ligne de commande). Si l'enregistrement a réussi, le Registre contiendra les classes `StyleVision.Server`. Cette classe se trouve généralement sous `HKEY_LOCAL_MACHINE\SOFTWARE\Classes`.

Une fois que l'objet de serveur COM est enregistré, vous pouvez l'invoquer depuis des applications et des langages de script qui prennent en charge des appels COM par programme. Si vous souhaitez changer l'emplacement du paquet d'installation StyleVision Server, il vaut mieux désinstaller StyleVision Server et puis de le réinstaller à l'emplacement requis. De cette manière, l'enregistrement ou le désenregistrement sont effectués par le processus de l'installateur.

## 5.3 À propos de l'interface Java

L'API consiste en un fichier JAR (`styleVisionServer.jar`) et en un fichier JNI (`styleVisionServer.dll`). Ces deux fichiers, de même que d'autres fichiers API associés, sont disponibles dans le dossier `bin` du dossier d'installation StyleVision Server. Vous pouvez soit référencer le fichier depuis leur emplacement d'origine soit les copier à un autre emplacement si cela convient à votre configuration de projet. (Sur les systèmes Windows, vous aurez besoin de droits administratifs pour exécuter le programme depuis son emplacement original.)

**Note** : Si vous avez installé StyleVision Server 64-bit, alors les fichiers de version 32-bit de `styleVisionServer.jar` et (`styleVisionServer.dll` seront situés dans le dossier `bin\API_32bit` du dossier d'installation StyleVision Server . Vous allez avoir besoin de ces fichiers si vous utilisez une version Java 32-bit. De la même manière, si vous avez installé StyleVision Server 32-bit, alors les fichiers de version 64-bit de `styleVisionServer.jar` et (`styleVisionServer.dll` seront situés dans le dossier `bin\API_64bit`. Vous allez avoir besoin de l'utilisation de ces fichiers si vous utilisez une version Java 64-bit.

Pour accéder à l'API StyleVision Server depuis le code Java, ajouter les références suivantes au fichier `.classpath` de votre projet Java.

<code>styleVisionServer.jar</code>	La bibliothèque qui communique avec StyleVision Server
<code>styleVisionServer_JavaDoc.zip</code>	Documentation de l'API StyleVision Server

En plus, les besoins `java.library.path` doivent inclure le dossier où le fichier bibliothèque JNI (`styleVisionServer.dll`) est situé.

Si vous déployer votre projet à un serveur d'application, assurez-vous que `styleVisionServer.jar` et `styleVisionServer.dll` sont correctement configurés avec Java sur l'appareil de serveur.

Pour un exemple comment utiliser les fichiers bibliothèque d'API, voir l'exemple de fichier batch `buildAndRun.bat` (*listed below*), qui est situé dans le dossier `etc\Examples\Java` du dossier d'installation StyleVision Server.

### Construire et exécuter un programme Java pour utiliser l'API

Pour voir comment construire et exécuter un programme Java qui utilise l'API StyleVision Server, voir l'exemple de fichier batch `buildAndRun.bat`. Vous pouvez réutiliser ce fichier à exécuter vos propres programmes Java en le modifiant, les cas échéant.

Démarrer le fichier dans une interface de ligne de commande avec la commande suivante :

```
buildAndRun "path_to_Java_bin_folder"
```

**Note** : Pour vérifier si Java est dans votre classpath, vous pouvez exécuter la commande `java --version`. Si Java n'est pas dans votre classpath, alors vous devez fournir le chemin comme paramètre de la commande `buildAndRun`. Si le chemin contient des espaces, alors utiliser des guillemets autour du chemin.

☐ Liste de `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
```

## Ajouter des bibliothèques dans Eclipse

Dans Eclipse, vous pouvez ajouter les références classpath en éditant les propriétés du projet Java. Les exemples d'instructions ci-dessous s'appliquent à Eclipse 4.4.

1. Avec le projet ouvert dans Eclipse, dans le menu **Projet**, cliquez sur les **Propriétés**, puis sélectionnez **Java Build Path**.
2. Dans l'onglet **Bibliothèques**, cliquez sur **Add External JARs**, puis chercher dans le fichier `StyleVisionServer.jar` situé dans le dossier d'installation de StyleVision Server.
3. Sous **JARs and class folders on the build path**, agrandir `StyleVisionServer.jar`, puis double-cliquer **Javadoc location : enregistrement (None)**.
4. S'assurer que les options **Javadoc in archive** et **External file** sont sélectionnées, puis chercher le fichier `StyleVisionServer_JavaDoc.zip` situé dans le dossier d'installation StyleVision Server.
5. Cliquez sur **OK**. La référence à la bibliothèque StyleVision Server et l'archive Javadoc est maintenant `.classpath` ajoutée au fichier.

## 5.4 Exemples de code

Les exemples dans cette section sont réalisés pour les langages de programmation suivants :

- [C#](#) <sup>63</sup>
- [C++](#) <sup>64</sup>
- [Java](#) <sup>66</sup>
- [VBScript](#) <sup>67</sup>
- [Visual Basic](#) <sup>69</sup>

### 5.4.1 C#

L'exemple ci-dessous montre comment utiliser le code C# pour générer un fichier de sortie RTF à l'aide d'un fichier PXF et d'un fichier XML d'entrée. Assurez-vous que StyleVision Server est installé et mis sous licence et qu'il est disponible en tant qu'objet de serveur COM. L'enregistrement en tant qu'objet de serveur COM a généralement lieu pendant l'installation de StyleVision Server. Pour vérifier que l'enregistrement a réussi, voir [A propos de l'interface COM](#) <sup>60</sup>. Voir aussi [A propos de l'interface .NET](#) <sup>59</sup>.

```
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
                //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++

L'exemple ci-dessous montre comment utiliser le code C++ pour générer un fichier de sortie RTF à l'aide d'un fichier PXF et d'un fichier XML d'entrée. Assurez-vous que StyleVision Server est installé et mis sous licence et qu'il est disponible en tant qu'objet de serveur COM. L'enregistrement en tant qu'objet de serveur COM a généralement lieu pendant l'installation de StyleVision Server. Pour vérifier que l'enregistrement a réussi, voir [A propos de l'interface COM](#)<sup>60</sup>. Voir aussi [A propos de l'interface .NET](#)<sup>59</sup>.

```

// 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.
#ifdef _WIN64
// 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
        //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

L'exemple ci-dessous montre comment utiliser le code code Java pour générer un fichier de sortie RTF à l'aide d'un fichier PXF et d'un fichier XML d'entrée. Assurez-vous que StyleVision Server est installé et mis sous licence et qu'il est disponible en tant qu'objet de serveur. L'enregistrement en tant qu'un objet de serveur a généralement lieu pendant l'installation de StyleVision Server. L'exemple de programme ci-dessous peut être créé et exécuté utilisant un fichier batch appelé `buildAndRun.bat`, qui est situé dans le dossier `etc\Examples\Java` de votre dossier d'installation StyleVision Server.

Pour des informations sur l'interface, voir [À propos de l'interface Java](#)<sup>61</sup>.

**Note :** Nous recommandons que vous copiez le dossier des Exemples dans votre répertoire de base ou tout autre dossier idéal, puis que vous parcouriez le dossier Java pour accéder à `Program.java` et `buildAndRun.bat`.

#### 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();

            //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 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

L'exemple ci-dessous montre comment utiliser le code VB Script pour générer un fichier de sortie RTF à l'aide d'un fichier PXF et d'un fichier XML d'entrée. Assurez-vous que StyleVision Server est installé et mis sous licence et qu'il est disponible en tant qu'objet de serveur COM. L'enregistrement en tant qu'un objet de serveur COM a généralement lieu pendant l'installation de StyleVision Server. Pour vérifier que l'enregistrement a réussi, voir [À propos de l'interface COM](#) <sup>60</sup>.

### 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-enable 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
'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
' 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-enable default error promotion

```

## 5.4.5 Visual Basic

L'exemple suivant illustre comment utiliser le code Visual Basic pour générer un fichier de sortie RTF utilisant un fichier PXF et un fichier XML d'entrée. Assurez-vous que StyleVision Server est installé et mis sous licence et qu'il est disponible en tant qu'objet de serveur COM. L'enregistrement en tant qu'objet de serveur COM a généralement lieu pendant l'installation de StyleVision Server. Pour contrôler si l'enregistrement a réussi, voir [À propos de l'interface COM](#) <sup>60</sup>.

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

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

End Sub

```

## 5.5 Référence API

Cette section est une référence de l'utilisateur pour l'API StyleVision Server.

- [COM et .NET](#)<sup>71</sup>
- [Java](#)<sup>78</sup>

### 5.5.1 COM et .NET

L'API StyleVisionServer expose l'**interface** `IServer`, qui crée une nouvelle instance d'objet StyleVision Server, et fournit un accès à StyleVision Server.

L'interface `IServer` a les méthodes et les propriétés suivantes.

#### Méthodes

### ▼ AddParameter

Assigne une valeur à un paramètre défini dans le fichier PXF.

#### ▣ 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

Efface la liste des paramètres.

#### ▣ C#

```
void ClearParameterList()
```

#### ▣ C++

```
HRESULT ClearParameterList();
```

#### ▣ VB

```
Sub ClearParameterList()
```

### ▼ Generate

Génère un ou plusieurs fichiers de sortie (HTML, PDF, RTF, et/ou DOCX) en utilisant le fichier PXF spécifié avec `TransfPath`. Il transforme le fichier XML d'entrée (Working XML File dans le fichier PXF) en utilisant le document XSLT contenu dans le fichier PXF. Retourne `TRUE` en cas de succès ; `FALSE` autrement.

#### ▣ 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
```

## Propriétés

### ▼ APIMajorVersion

Obtient la version majeure de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

#### ▣ C#

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

#### ▣ C++

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

#### ▣ VB

```
ReadOnly Property APIMajorVersion As Integer
```

### ▼ APIMinorVersion

Obtient la version mineure de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

#### ▣ C#

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

#### ▣ C++

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

#### ▣ VB

```
ReadOnly Property APIMinorVersion As Integer
```

### ▼ APIServicePackVersion

Obtient la version du service pack de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

#### ▣ C#

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

#### ▣ C++

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

#### ▣ VB

```
ReadOnly Property APIServicePackVersion As Integer
```

### ▼ InputXML

Définit le chemin et le nom du fichier XML à traiter (le Working XML File dans le fichier PXF).

#### ▣ C#

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

## ▣ C++

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

## ▣ VB

```
Property InputXML As String
```

## ▼ Is64Bit

Retourne `TRUE` si le moteur StyleVision Server est un exécutable 64-bit.

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
ReadOnly Property Is64Bit As Boolean
```

## ▼ LastExecutionMessage

Obtient le message reçu pendant la dernière commande `Generate`.

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
ReadOnly Property LastExecutionMessage As String
```

## ▼ MajorVersion

Obtient la version majeure de StyleVision Server.

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
ReadOnly Property MajorVersion As Integer
```

## ▼ MinorVersion

Obtient la version mineure de StyleVision Server.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property MinorVersion As Integer
```

▼ OutputDOCX

Définit le chemin et le nom du fichier de sortie DOCX.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputDOCX As String
```

▼ OutputFO

Définit le chemin et le nom du fichier de sortie FO.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputFO As String
```

▼ OutputHTML

Définit le chemin et le nom du fichier de sortie HTML.

▣ C#

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

▣ C++

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

▣ VB

Property OutputHTML As String

#### ▼ OutputPDF

Définit le chemin et le nom du fichier de sortie PDF.

##### ▣ C#

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

##### ▣ C++

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

##### ▣ VB

```
Property OutputPDF As String
```

#### ▼ OutputRTF

Définit le chemin et le nom du fichier de sortie RTF.

##### ▣ C#

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

##### ▣ C++

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

##### ▣ VB

```
Property OutputRTF As String
```

#### ▼ OutputText

Définit le chemin et le nom du fichier de sortie texte.

##### ▣ C#

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

##### ▣ C++

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

##### ▣ VB

```
Property OutputText As String
```

#### ▼ ProductName

Obtient le nom du produit : "StyleVision Server"

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
ReadOnly Property ProductName As String
```

## ▼ ProductNameAndVersion

Obtient le nom complet du produit, y compris le numéro de version : "StyleVision Server 2014r2 sp1 (x64)".

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
ReadOnly Property ProductNameAndVersion As String
```

## ▼ ServerPath

Obtient ou définit le chemin vers l'exécutable StyleVision Server.

## ▣ C#

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

## ▣ C++

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

## ▣ VB

```
Property ServerPath As String
```

## ▼ ServicePackVersion

Obtient la version du pack de service de StyleVision Server (par exemple : 1 pour Altova StyleVision Server 2014 r2 sp1 (x64).)

## ▣ C#

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

## ▣ C++

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

▣ VB

```
ReadOnly Property ServicePackVersion As Integer
```

▼ WhereClause

Définit une clause SQL `WHERE` qui détermine les lignes d'une source de schéma DB-XML à traiter.

▣ C#

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

▣ C++

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

▣ VB

```
Property WhereClause As String
```

▼ WorkingDirectory

Obtient ou définit le répertoire actuel pour des tâches en cours. Les chemins relatifs sont évalués par rapport au répertoire de travail.

▣ 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

Le paquet `com.altova.stylevisionserver` consiste des classes suivantes :

- `public class StyleVisionServer` (*décrit ci-dessous*)
- `public class StyleVisionServerException` extends `Exception`

### **StyleVisionServer class**

La classe `StyleVisionServer` crée une nouvelle instance d'objet `StyleVision Server`, et fournit l'accès à `StyleVision Server`. Les méthodes de l'interface `StyleVisionServer` sont décrites ci-dessous.

## Méthodes de la classe `styleVisionServer`

Les méthodes de la classe `StyleVisionServer` sont listées ci-dessous par ordre alphabétique.

### ▼ `addParameter`

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

Ajoute le nom et la valeur d'un nouveau paramètre. Chaque paramètre et sa valeur est spécifiée dans un appel séparer à la méthode. Les paramètres doivent être déclarés dans le document XSLT.

#### Paramètres :

name: Détient le nom du paramètre en tant qu'un string.

value: Détient la valeur of du paramètre en tant qu'un string.

### ▼ `clearParameterList`

```
public void clearParameterList()
```

Efface la liste des paramètres.

### ▼ `generate`

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

Traite le fichier PXF spécifié dans `transfPath`. Lance `StyleVisionServerException`.

#### Paramètres :

`transfPath`: Une URL absolue donnant l'emplacement du fichier PXF.

#### Retourne :

`true()` si l'exécution est réussie

`false()` si l'exécution échoue

Dans le cas d'une erreur, utiliser `getLastExecutionMessage()`

### ▼ `getAPIMajorVersion`

```
public int getAPIMajorVersion()
```

Obtient la version majeure de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

### ▼ `getAPIMinorVersion`

```
public int getAPIMinorVersion()
```

Obtient la version mineure de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

### ▼ `getAPIServicePackVersion`

```
public int getAPIServicePackVersion()
```

Obtient la version du pack service de l'API StyleVision Server. Elle peut être différente de la version de produit si l'API est connectée à un autre serveur.

### ▼ `getLastExecutionMessage`

```
public String getLastExecutionMessage()
```

Obtient le message reçu pendant la dernière commande `generate`.

### ▼ `getMajorVersion`

```
public int getMajorVersion()
```

Obtient la version majeure de l'application.

▼ getMinorVersion

```
public int getMinorVersion()
```

Obtient la version mineure de l'application.

▼ getProductName

```
public String getProductName()
```

Obtient le nom de produit.

▼ getProductNameAndVersion

```
public String getProductNameAndVersion()
```

Obtient le nom complet et le numéro de version du produit.

▼ getServicePackVersion

```
public int getServicePackVersion()
```

Obtient la version du pack de service de StyleVision Server.

▼ is64bit

```
public boolean is64bit()
```

Contrôle si l'exécutable est 64-bit.  
Retourne :  
true() pour StyleVision Server (x64), false() sinon.

▼ setInputXML

```
public void setInputXML(String path)
```

Définit le fichier XML à traiter. Doit être le chemin du Working XML File qui est spécifié dans le fichier PXF.  
Paramètres :  
path: Contient le chemin du Working XML file dans le fichier PXF.

▼ setOutputDOCX

```
public void setOutputDOCX(String path)
```

Définit le chemin et le nom du fichier DOCX à générer.  
Paramètres :  
path: Le chemin et le nom du fichier DOCX à générer.

▼ setOutputFO

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

Définit le chemin et le nom du fichier FO à générer.  
Paramètres :  
path: Le chemin et le nom du fichier FO à générer.

▼ setOutputHTML

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

Définit le chemin et le nom du fichier HTML à générer.

Paramètres :

path: Le chemin et le nom du fichier HTML à générer.

#### ▼ setOutputPDF

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

Définit le chemin et le nom du fichier PDF à générer.

Paramètres :

path: Le chemin et le nom du fichier PDF à générer.

#### ▼ setOutputRTF

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

Définit le chemin et le nom du fichier RTF à générer.

Paramètres :

path: Le chemin et le nom du fichier RTF à générer.

#### ▼ setOutputText

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

définit le chemin et le nom du fichier texte à générer.

Paramètres :

path: Le chemin et le nom du fichier texte à générer.

#### ▼ setServerPath

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

Définit le chemin de l'exécutable StyleVisionServer.

Paramètres :

serverFile: Le chemin de l'exécutable StyleVisionServer.

#### ▼ setWhereClause

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

Définit une clause SQL WHERE qui détermine les lignes d'une source de schéma DB-XML à traiter.

Paramètres :

whereClause: La clause SQL WHERE qui détermine les lignes d'une source de schéma DB-XML à traiter.

#### ▼ setWorkingDirectory

```
public void setWorkingDirectory(String workingDirectory)
```

Définit un répertoire par défaut. Les chemins relatifs sont résolus relativement à ce répertoire.

Paramètres :

workingDirectory: Le chemin du répertoire (de travail) par défaut.

## 6 Gestionnaire de schéma

XML Gestionnaire de schéma est un outil qui propose un moyen centralisé d'installer et de gérer des schémas XML (DTD pour XML et Schémas XML) pour une utilisation sur toutes les applications « XML-Schema-aware » d'Altova, y compris StyleVision Server

- Sur Windows, Gestionnaire de schéma a une interface utilisateur graphique (*voir la capture d'écran ci-dessous*) et est aussi disponible dans la ligne de commande. (Les applications desktop d'Altova sont disponibles sur Windows uniquement ; *voir la liste ci-dessous*.)
- Sur Linux et Mac Gestionnaire de schéma, l'outil est disponible uniquement dans la ligne de commande. (Les applications serveur d'Altova sont disponibles sur Windows, Linux et macOS ; *voir la liste ci-dessous*.)



*Application d'Altova qui fonctionnent avec Schema Manager*

Applications desktop (Windows uniquement)	Applications de serveur (Windows, Linux, macOS)
XMLSpy ( toutes éditions )	RaptorXML Server, RaptorXML+XBRL Server

MapForce ( toutes éditions )	StyleVision Server
StyleVision ( toutes éditions )	
Authentic Desktop Enterprise Edition	

## Installation et désinstallation de Gestionnaire de schéma

Gestionnaire de schéma est installé automatiquement quand vous installez d'abord une nouvelle version de l'Altova Mission Kit ou toute application « XML-schema-aware » d'Altova (voir la table ci-dessus).

De même, il est supprimé automatiquement lorsque vous désinstallez la dernière application XML-schema-aware d'Altova depuis votre ordinateur.

## Fonctions <% SCHEMA-MANAGER%>

Gestionnaire de schéma propose les fonctions suivantes :

- Affiche les schémas XML installés sur votre ordinateur et contrôle si de nouvelles versions sont disponibles pour le téléchargement.
- Télécharge des versions plus récentes des schémas XML indépendamment du cycle de release des produits Altova. (Altova stocke des schémas en ligne et vous pouvez les télécharger via Gestionnaire de schéma.)
- Installer ou désinstaller une des multiples versions d'un schéma donné (ou toutes les versions, si nécessaire).
- Un schéma XML peut avoir des dépendances sur d'autres schémas. Lorsque vous installez ou désinstallez un schéma particulier, Gestionnaire de schéma vous informe sur d'autres schémas dépendants et les installera ou désinstallera également automatiquement.
- Gestionnaire de schéma utilise le mécanisme du [catalogue XML](#) pour mapper les références de schéma aux fichiers locaux. Dans le cas de larges schémas XML, le traitement sera plus rapide que si les schémas étaient à un emplacement à distance.
- Tous les schémas majeurs sont disponibles via Gestionnaire de schéma et sont régulièrement mis à jour pour les dernières versions. Ceci vous fournit une ressource unique pour gérer tous vos schémas et les mettre à disposition de toutes les applications « XML-schema-aware » d'Altova.
- Les changements réalisés dans Gestionnaire de schéma prennent effet pour tous les produits d'Altova sur cet appareil.

## Comment cela fonctionne ?

Altova stocke tous les schémas XML utilisés dans les produits Altova en ligne. Ce référentiel est mis à jour lorsque de nouvelles versions de schémas sont publiées. Gestionnaire de schéma affiche des informations sur les derniers schémas disponibles lorsqu'ils sont appelés dans son formulaire GUI de même que sur CLI. Vous pouvez ensuite installer, mettre à jour ou désinstaller les schémas via Gestionnaire de schéma.

Gestionnaire de schéma installe également les schémas d'une autre manière. Sur le site web d'Altova (<https://www.altova.com/fr/schema-manager>), vous pouvez sélectionner un schéma et ses Schémas dépendants que vous souhaitez installer. Le site web préparera un fichier de type `.altova_xmlschemas` pour le téléchargement qui contient des informations sur la sélection de schéma. Lorsque vous double-cliquez sur ce fichier ou le passez à Gestionnaire de schéma via CLI comme argument de la commande `installer`<sup>96</sup>, Gestionnaire de schéma installera les schémas que vous avez sélectionnés.

Cache local : suivre vos schémas

Toutes les informations sur les schémas installés sont suivies dans un répertoire cache centralisé sur votre ordinateur, situé ici :

<i>Windows</i>	C:\ProgramData\Altova\pkgs\cache
<i>Linux</i>	/var/opt/Altova/pkgs/cache
<i>macOS</i>	/var/Altova/pkgs

Ce répertoire cache est mis à jour régulièrement avec le dernier statut des schémas dans l'emplacement de stockage en ligne d'Altova. Ces mises à jour sont réalisées aux moments suivants :

- À chaque fois que vous lancez Gestionnaire de schéma.
- Lorsque vous exécutez StyleVision Server pour la première fois dans un jour donné du calendrier.
- Si StyleVision Server est ouvert plus de 24 heures, le cache est mis à jour toutes les 24 heures.
- Vous pouvez aussi mettre à jour le cache en exécutant la commande de [mise à jour](#)<sup>99</sup> dans l'interface de ligne de commande.

Pour cette raison, le cache permet à Gestionnaire de schéma de suivre continuellement vos schémas installés par rapport aux schémas disponibles en ligne sur le site web d'Altova.

**Ne modifiez pas le cache manuellement !**

Le répertoire de cache local est entretenu automatiquement sur la base des schémas que vous installez ou désinstallez. Il ne devrait pas être altéré ou supprimé manuellement. Si vous êtes amené à réinitialiser Gestionnaire de schéma à son état original "intact", alors, sur l'interface de la ligne de commande (CLI) : (i) exécutez la commande [reset](#)<sup>97</sup>, et (ii) exécutez la commande [initialize](#)<sup>95</sup>. (En alternative, exécutez la commande `reset` avec l'option `--i`.)

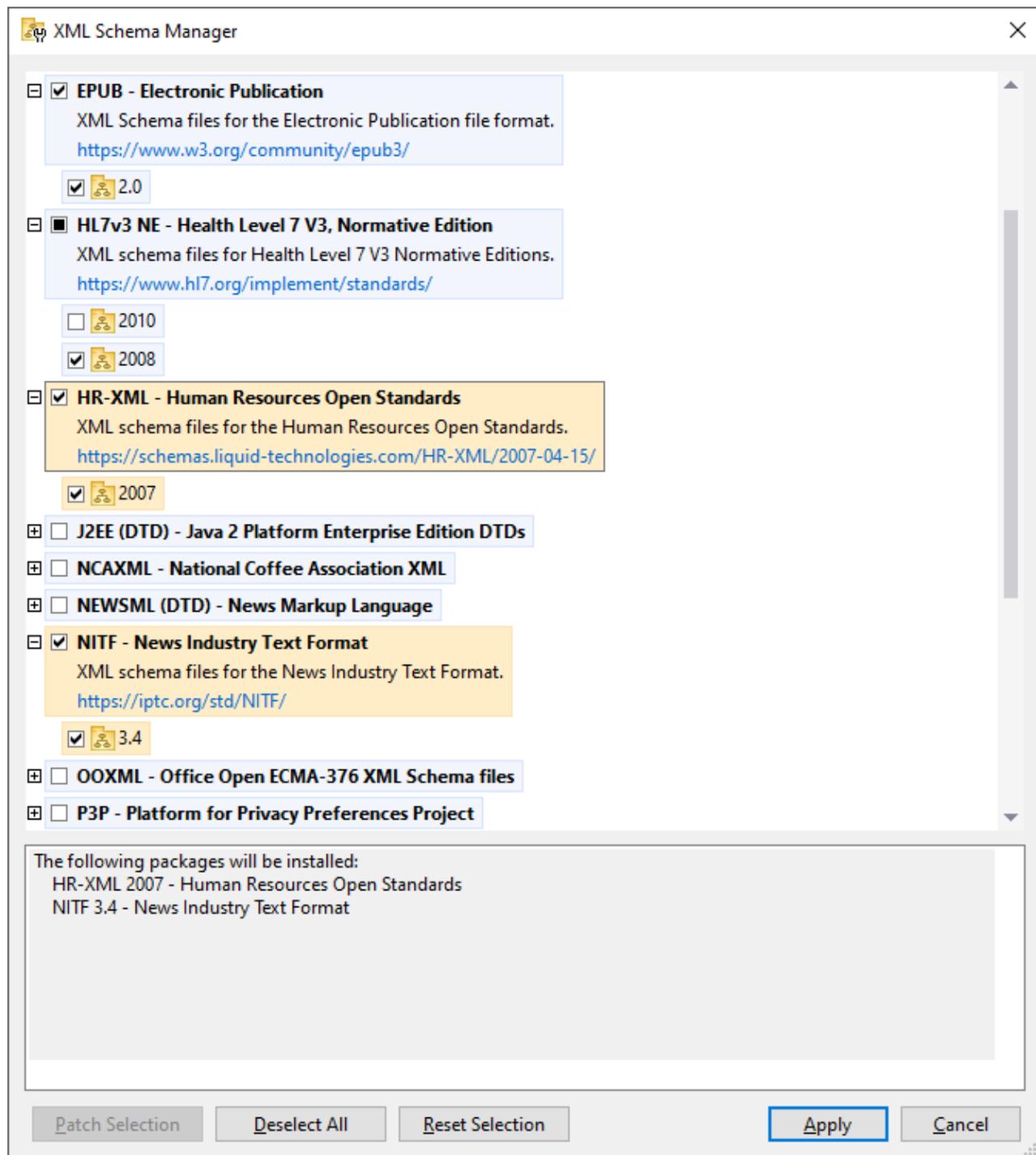
## 6.1 Exécuter Schema Manager

### Interface utilisateur graphique

Vous pouvez accéder à la GUI de Gestionnaire de schéma des manières suivantes :

- *Durant l'installation de StyleVision Server:* Vers la fin de la procédure d'installation, sélectionnez la case à cocher *Invoke Altova XML-Schema Manager* pour accéder directement à la GUI de Gestionnaire de schéma. Ceci vous permettra d'installer les schémas au cours de la procédure d'installation de votre application Altova.
- Via le fichier `.altova_taxonomies` téléchargé de [Altova website](#): Double-cliquez sur le fichier téléchargé pour exécuter la GUI de Gestionnaire de schéma, qui sera configurée pour installer les schémas que vous avez sélectionnés (le site web) pour installation.

Une fois que la GUI de Gestionnaire de schéma (*capture d'écran ci-dessous*) a été ouverte, les schémas déjà installés seront affichés tels sélectionnés. Si vous voulez installer un schéma additionnel, sélectionnez-le. Si vous voulez désinstaller un schéma déjà installé, désélectionnez-le. Une fois que vous avez faits vos sélections et/ou désélections, vous êtes prêts pour appliquer vos changements. Les schémas qui seront installés ou désinstallés seront mis en surbrillance et un message sur les modifications à venir sera posté dans le volet Messages au niveau inférieur de la fenêtre Gestionnaire de schéma (*voir la capture d'écran*).



### Interface de ligne de commande

Vous pouvez exécuter Gestionnaire de schéma depuis une interface de ligne de commande en sélectionnant son fichier exécutable, `xmlschemamanager.exe`.

Le fichier `xmlschemamanager.exe` est situé dans le dossier suivant :

- *Sur Windows* : `C:\ProgramData\Altova\SharedBetweenVersions`
- *Sur Linux ou macOS (application serveur uniquement)* : `%INSTALLDIR%/bin`, où `%INSTALLDIR%` est le répertoire d'installation du programme.

Vous pouvez alors utiliser toute commande dans la [section de référence de la commande CLI](#) <sup>94</sup>.

Pour afficher l'aide pour la commande, exécutez l'étape suivante :

- *Sur Windows* : `xmlschemamanager.exe --help`
- *Sur Linux ou macOS (application serveur uniquement)* : `sudo ./xmlschemamanager --help`

## 6.2 Catégories de statut

Gestionnaire de schéma catégorise les schémas sous sa gestion comme suit :

- *Schémas installés.* Ceux-ci sont affichés dans la GUI avec leurs cases à cocher sélectionnées (dans la capture d'écran ci-dessous, les versions cochées et bleues des schémas EPUB et HL7v3 NE sont des schémas installés). Si toutes les versions de schéma sont sélectionnées, alors la marque de sélection est une coche. Si au moins une version de schéma est décochée, alors la coche de sélection est un carré coloré plein. Vous pouvez décocher un schéma installé pour le **désinstaller** ; (dans la capture d'écran ci-dessous, le DocBook DTD est installé et a été désélectionné, le préparant ainsi pour la désinstallation).
- *Schémas désinstallés disponibles.* Ils sont affichés dans la GUI avec leurs cases à cocher non sélectionnées. Vous pouvez sélectionner les schémas que vous souhaitez **installer**.



- Les *Schémas pouvant être mises à niveau* sont ceux qui ont été revus par leurs émetteurs depuis qu'ils ont été installés. Ils sont indiqués dans la GUI par une icône . Vous pouvez **retoucher** le schéma installé avec la révision disponible.

### Points à noter

- Dans la capture d'écran ci-dessus, les deux schémas CBCR sont cochés. Celui avec un arrière-plan bleu est déjà installé. Celui avec un arrière-plan jaune est désinstallé et a été sélectionné pour l'installation. Notez que le schéma HL7v3 NE 2010 n'est pas installé et n'a pas été sélectionné pour l'installation.
- Un arrière-plan jaune signifie que le schéma sera modifié d'une manière ou d'une autre quand le bouton **Appliquer** est cliqué. Si un schéma est décoché et a un arrière-plan jaune, cela signifie qu'il sera

désinstallé quand le bouton **Appliquer** est cliqué. Dans la capture d'écran ci-dessus, le DocBook DTD a un tel statut.

- Lorsque vous exécutez Gestionnaire de schéma depuis la ligne de commande, la commande [list](#)<sup>96</sup> est utilisée avec différentes options pour recenser les différentes catégories de schémas :

<code>xmlschemamanager.exe list</code>	Recense tous les schémas installés et disponibles ; ceux pouvant être mis à niveau sont également indiqués.
<code>xmlschemamanager.exe list -i</code>	Recense les schémas installés uniquement ; ceux pouvant être mis à niveau sont également indiqués
<code>xmlschemamanager.exe list -u</code>	Recense les schémas pouvant être mis à niveau

**Note :** Sur Linux et macOS, use `sudo ./xmlschemamanager list`

## 6.3 Retoucher ou Installer un schéma

### Retoucher un schéma installé

Occasionnellement, des schémas XML peuvent recevoir des patches (mises à niveau ou révisions) de leurs émetteurs. Lorsque Gestionnaire de schéma détecte que des patches sont disponibles, ceux-ci sont indiqués dans les listes de schéma de Gestionnaire de schéma et vous pouvez installer les patches rapidement.

#### Dans la GUI

Les patches sont indiqués par l'icône . (Voir aussi la rubrique précédente sur les [catégories de statut](#)<sup>89</sup>.) Si les patches sont disponibles, le bouton **Patch Selection** sera activé. Cliquez dessus pour sélectionner et préparer tous les patches pour installation. Dans la GUI, l'icône de chaque schéma sera patchée de  à , et le volet des Messages en bas du dialogue recense les patches qui doivent être appliqués. Lorsque vous êtes prêt pour installer des patches sélectionnés, cliquez sur **Appliquer**. Tous les correctifs seront appliqués ensemble. Notez que si vous décochez un schéma marqué pour une correction, vous désinstallerez de fait ce schéma.

#### Sur le CLI

Pour appliquer un patch dans l'interface de ligne de commande :

1. Exécuter la commande `list -u`<sup>96</sup> . Cela liste tout schéma lorsque des mises à niveau sont disponibles.
2. Exécutez la commande `upgrade`<sup>99</sup> pour installer les patches.

### Installer un schéma disponible

Vous pouvez installer des schémas en utilisant soit la GUI Gestionnaire de schéma ou en envoyant à Gestionnaire de schéma les instructions d'install via la ligne de commande.

**Note** : Si le schéma actuel référence d'autres schémas, les schémas référencées sont aussi installés.

#### Dans la GUI

Pour installer des schémas utilisant la GUI Gestionnaire de schéma GUI, sélectionnez les schémas que vous voulez installer et cliquez sur **Appliquer**.

Vous pouvez aussi sélectionner les schémas que vous voulez installer sur le [site web d'Altova](#) et générer un fichier téléchargeable `.altova_xmlschemas`. Lorsque vous double-cliquez sur ce fichier, il ouvrira Gestionnaire de schéma avec les schémas que vous vouliez présélectionner. La seule chose qui vous reste à faire, c'est cliquer sur **Appliquer**.

#### Sur le CLI

Pour installer des schémas via la ligne de commande, exécutez la commande `install`<sup>96</sup> :

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

où `schéma` est le schéma (ou les schémas) que vous voulez installer ou un fichier `.altova_xmlschemas`. Un schéma est référencé par un identifiant de format `<name>--<version>`. (Les identifiants de schémas sont affichés quand vous exécutez la commande `list`<sup>96</sup>.) Vous pouvez saisir autant de schémas que vous le souhaitez. Pour plus de détails, voir la description de la commande `install`<sup>96</sup>.

**Note** : sur Linux ou macOS, utilisez la commande `sudo ./xmlschemamanager`.

#### Installer un schéma requis

Lorsque vous exécutez une commande activée par XML dans StyleVision Server, et que StyleVision Server découvre qu'un schéma dont il a besoin pour exécuter la commande n'est pas présente ou est incomplète, Gestionnaire de schéma affichera l'information sur les schémas manquants. Vous pouvez ensuite installer directement tout schéma manquant via Gestionnaire de schéma.

Dans la GUI de Schema Manager, vous pouvez consulter tous les schémas précédemment installés à tout moment en exécutant Gestionnaire de schéma depuis **Outils | Gestionnaire de schéma**.

## 6.4 Désinstaller un schéma, Réinitialiser

### Désinstaller un schéma

Vous pouvez désinstaller des schémas en utilisant soit la GUI Gestionnaire de schéma ou en envoyant à Gestionnaire de schéma les instructions d'installation via la ligne de commande.

**Note** : si la Police que vous voulez désinstaller référence d'autres Schémas, alors les Schémas référencées sont également désinstallées.

#### Dans la GUI

Pour désinstaller les schémas utilisant la GUI Gestionnaire de schéma, effacez leurs cases à cocher et cliquez sur **Appliquer**. Les schémas sélectionnés et leurs schémas référencées seront désinstallés.

Pour désinstaller tous les schémas, cliquez sur **Désélectionner tout** et cliquez sur **Appliquer**.

#### Sur le CLI

Pour désinstaller des schémas via la ligne de commande, exécutez la commande [désinstaller](#)<sup>98</sup> :

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

où chaque argument `schéma` est le schéma que vous voulez désinstaller ou un fichier `.altova_xmlschemas`. Un schéma est spécifié par un identifiant qui a un format `<name>-<version>`. (Les identifiants de schémas sont affichés quand vous exécutez la commande [list](#)<sup>96</sup>.) Vous pouvez saisir autant de schémas que vous le souhaitez. Pour plus de détails, voir la description de la commande [désinstaller](#)<sup>98</sup>.

**Note** : sur Linux ou macOS, utilisez la commande `sudo ./xmlschemamanager`.

### Réinitialiser Gestionnaire de schéma

Vous pouvez réinitialiser Gestionnaire de schéma. Ceci supprime toutes les schémas installés et le répertoire de mise sous cache.

- Dans la GUI, cliquez sur **Reset Selection**.
- Dans la CLI, exécutez la commande [reset](#)<sup>97</sup>.

Une fois avoir exécuté cette commande, vous devrez exécuter la commande [initialize](#)<sup>95</sup>, pour pouvoir recréer le répertoire de mise sous cache. En alternative, exécutez la commande [reset](#)<sup>97</sup> avec l'option `-i`.

Notez que [reset -i](#)<sup>97</sup> restaure l'installation originale du produit, il est recommandé d'exécuter la commande [update](#)<sup>99</sup> après avoir réalisé la réinitialisation. En alternative, exécutez la commande [reset](#)<sup>97</sup> avec les options `-i` and `-u`.

## 6.5 Interface de ligne de commande (CLI)

Pour appeler Gestionnaire de schéma dans la ligne de commande, vous devez connaître le chemin de l'exécutable. Par défaut, l'exécutable Gestionnaire de schéma est installé dans le chemin suivant :

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

**Note** : sur les systèmes Linux et macOS, une fois que vous avez changé le répertoire à celui contenant l'exécutable, vous pouvez appeler l'exécutable avec `sudo ./xmlschemamanager`. Le préfixe `./` indique que l'exécutable est le répertoire actuel. Le préfixe `sudo` indique que la commande doit être exécutée avec des privilèges root.

### Syntaxe de ligne de commande

La syntaxe générale pour utiliser la ligne de commande est la suivante :

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

Dans l'extrait ci-dessus, la barre verticale `|` sépare un ensemble d'items mutuellement exclusifs. Les crochets `[]` indiquent des items optionnels. De manière générale, vous pouvez saisir le chemin d'exécutable suivi soit par les options `--h`, `--help`, ou `--version` ou par une commande. Chaque commande peut contenir des options et des arguments. La liste des commandes est décrite dans les sections suivantes.

### 6.5.1 help

Cette commande propose une aide contextuelle pour les commandes liées à l'exécutable Gestionnaire de schéma.

#### Syntaxe

```
<exec> help [command]
```

Où `[command]` est un argument optionnel qui spécifie un nom de commande valide.

Veillez noter les points suivants :

- Vous pouvez invoquer de l'aide en saisissant la commande suivie par `-h` ou `--help`, par exemple :  
`<exec> list-h`
- Si vous tapez `-h` or `--help` directement après la commande exécutable et avant une commande, vous recevrez une aide générale (pas d'aide pour la commande), par exemple : `<exec> -h list`

#### Exemple

La commande suivante affiche une aide concernant la commande `list` :

```
xmlschemamanager help list
```

## 6.5.2 info

Cette commande affiche des informations détaillées pour chacun des schémas fournis en tant qu'argument de Schéma. Cette information inclut le titre, la version, description, l'éditeur et tout schéma soumis et tout schéma référencé, et informe si le schéma a été installé ou non.

### Syntaxe

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

- L'argument **schéma** est le nom d'un schéma ou une partie du nom de schéma. (Pour afficher une ID de pack de schéma et des informations détaillées sur son statut d'installation, vous devriez utiliser la commande [list](#) <sup>96</sup>.)
- Utiliser `<exec> info -h` pour afficher l'aide de la commande.

### Exemple

La commande suivante affiche l'information sur les derniers schémas **DocBook-DTD** et **NITF** :

```
xmlschemamanager info doc nitf
```

## 6.5.3 initialize

Cette commande initialise l'environnement Gestionnaire de schéma. Elle crée un répertoire de cache où les informations concernant tous les schémas sont stockées localement. L'initialisation est réalisée automatiquement la première fois qu'une application schema-cognizant d'Altova est installée. Vous n'aurez pas besoin d'exécuter cette commande dans des circonstances normales, mais vous devrez l'exécuter généralement après la commande **reset**.

### Syntaxe

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

#### Options

La commande **initialize** accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante initialise Gestionnaire de schéma:

```
xmlschemamanager initialize
```

### 6.5.4 install

Cette commande installe un ou plusieurs schémas.

#### Syntaxe

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

Pour installer de multiples schémas, ajoutez l'argument `schéma` de nombreuses fois.

L'argument `schéma` est l'un des suivants :

- Un identifiant de schéma (avoir un format de `<name>-<version>`, par exemple : `cbcr-2.0`). Pour trouver les identifiants de schémas que vous voulez, exécutez la commande [list](#)<sup>96</sup>. Vous pouvez aussi utiliser des identifiants abrégés s'ils sont uniques, par exemple `docbook`. Si vous utilisez un identifiant abrégé, alors la dernière version de ce schéma sera installée.
- Le chemin vers un fichier `.altova_xmlschemas` téléchargé depuis le site web d'Altova. Pour information sur ces fichiers, voir [Introduction à SchemaManager : Comment cela fonctionne-t-il ?](#)<sup>82</sup>.

#### Options

La commande `install` accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante installe le schéma CBCR 2.0 (Country-By-Country Reporting) et le dernier DocBook DTD:

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

### 6.5.5 list

Cette commande recense les schémas sous la gestion de Gestionnaire de schéma. La liste affiche comme suit

- Tous les schémas disponibles
- Les schémas contenant le string dans leur nom soumis comme argument de `schéma`
- Seuls les schémas installés
- Seuls les schémas qui peuvent être mis à niveau

## Syntaxe

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

Si aucun argument de `schéma` n'est soumis, alors toutes les schémas disponibles sont recensés. Autrement, les schémas sont recensés par des options soumises (*voir l'exemple ci-dessous*). Notez que vous pouvez soumettre l'argument de `schéma` de nombreuses fois.

### Options

La commande `list` accepte les options suivantes :

<code>--installed, --i</code>	Recenser uniquement la liste des schémas installés. Le réglage par défaut est <b>faux</b> .
<code>--upgradeable, --u</code>	Recenser uniquement les schémas lorsque des mises à niveau (patches) sont disponibles. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemples

- Pour exécuter tous les schémas disponibles, exécutez : `xmlschemamanager list`
- Pour recenser les schémas installés, exécutez : `xmlschemamanager list -i`
- Pour recenser tous les schémas qui contiennent soit "doc", soit "nitf" dans leur nom, exécutez `xmlschemamanager list doc` :

## 6.5.6 reset

Cette commande supprime tous les schémas installés et le répertoire de mise sous cache. Vous réinitialiserez complètement votre environnement de schéma. Une fois avoir exécuté cette commande, vous devrez exécuter la commande `initialize`<sup>95</sup>, pour pouvoir recréer le répertoire de mise sous cache. En alternative, exécuter la commande `reset` avec l'option `-i`. Puisque `reset -i` restaure l'installation originale du produit, nous vous recommandons que vous exécutiez la commande `update`<sup>99</sup> après avoir réalisé la réinitialisation et l'initialisation. En alternative, exécutez la commande `reset` avec les options `-i` and `-u`

## Syntaxe

```
<exec> reset [options]
```

### Options

La commande `reset` accepte les options suivantes :

<code>--init, --i</code>	Initialiser Gestionnaire de schéma après le reset. Le réglage par défaut est <b>faux</b> .
<code>--update, --u</code>	Mettre à jour la liste de schémas disponibles dans le cache. Le réglage par défaut est <b>faux</b> .
<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemples

- Pour réinitialiser Gestionnaire de schéma, exécuter : `xmlschemamanager reset`
- Pour réinitialiser Gestionnaire de schéma et l'initialiser, exécutez : `xmlschemamanager reset -i`
- Pour réinitialiser Gestionnaire de schéma, initialiser-le et mettez à jour sa liste de schéma, exécutez : `xmlschemamanager reset -i-u`

## 6.5.7 uninstall

Cette commande désinstalle un ou plusieurs schémas. Par défaut, tout schéma référencé par la taxonomie actuelle sera également désinstallé. Pour désinstaller uniquement le schéma actuel et garder les schémas référencés, définir l'option `--k`.

### Syntaxe

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

Pour désinstaller de multiples schémas, ajoutez l'argument `schéma` de nombreuses fois.

L'argument `schéma` est l'un des suivants :

- Un identifiant de schéma (avoir un format de `<name>-<version>`, par exemple : `cocr-2.0`). Pour trouver les identifiants de schéma qui sont installés, exécutez la commande `list -i`<sup>96</sup>. Vous pouvez aussi utiliser un nom de schéma abrégé s'il est unique, par exemple `docbook`. Si vous utilisez un nom abrégé, alors tous les schémas qui contiennent une abréviation dans leur nom seront désinstallés.
- Le chemin vers un fichier `.altova_xmlschemas` téléchargé depuis le site web d'Altova. Pour information sur ces fichiers, voir [Introduction à SchemaManager : Comment cela fonctionne-t-il ?](#)<sup>82</sup>.

### Options

La commande `désinstaller` accepte les options suivantes :

<code>--keep-references, --k</code>	Définir cette option pour garder les schémas référencés. Le réglage par défaut est <b>faux</b> .
-------------------------------------	--

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante désinstalle les schémas CBCR 2.0 et EPUB 2.0 et leurs dépendances :

```
xmlschemamanager uninstall cbc-2.0 epub-2.0
```

La commande suivante désinstalle la taxonomie `eba-2.10` mais pas les schémas qu'elle référence :

```
xmlschemamanager uninstall --k cbc-2.0
```

## 6.5.8 update

Cette commande requête la liste des schémas disponibles depuis l'emplacement de stockage en ligne et met à jour le répertoire de mise sous cache local. Vous devriez exécuter cette commande sauf si vous avez réalisé un [reset](#)<sup>97</sup> et [initialize](#)<sup>95</sup>.

### Syntaxe

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

#### Options

La commande **mise à jour** accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante met à jour le cache local avec la liste des derniers schémas :

```
xmlschemamanager update
```

## 6.5.9 upgrade

Cette commande met à niveau toutes les schémas installés qui peuvent être mis à niveau à la dernière version *patchée* disponible. Vous pouvez identifier des schémas à mettre à niveau en exécutant la commande [list -u](#)<sup>96</sup>.

**Note** : La commande `mettre à niveau` supprime une Police dépréciée si aucune version plus récente n'est disponible.

## Syntaxe

```
<exec> upgrade [options]
```

### Options

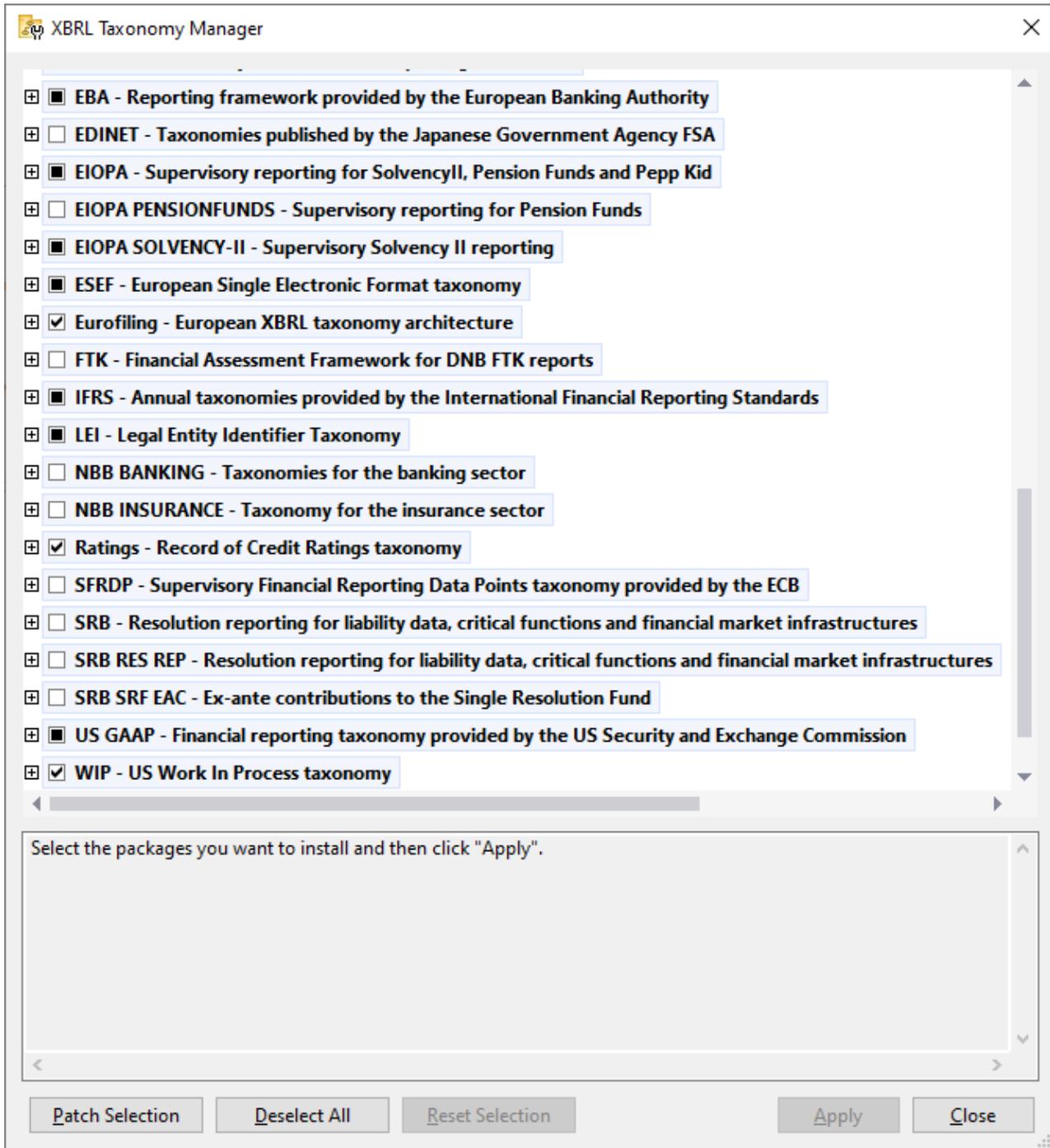
La commande `mise à niveau` accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## 7 Gestionnaire de taxonomie

XBRL Gestionnaire de taxonomie est un outil qui propose un moyen centralisé d'installer et de gérer des taxonomies XBRL pour une utilisation sur toutes les applications XBRL Altova, y compris StyleVision Server

- Sur Windows, Gestionnaire de taxonomie a une interface utilisateur graphique (*voir la capture d'écran ci-dessous*) et est aussi disponible dans la ligne de commande. (Les applications desktop d'Altova sont disponibles sur Windows uniquement ; *voir la liste ci-dessous*.)
- Sur Linux et Mac Gestionnaire de taxonomie, l'outil est disponible uniquement dans la ligne de commande. (Les applications serveur d'Altova sont disponibles sur Windows, Linux et macOS ; *voir la liste ci-dessous*.)



Les applications activées par XBRL d'Altova

Applications desktop (Windows uniquement)	Applications de serveur (Windows, Linux, macOS)
---	---

Altova XBRL Add-ins pour Excel (EBA, Solvency II, II)	MapForce Server (Standard et Advanced Edition).
MapForce Enterprise Edition	RaptorXML+XBRL Server
StyleVision Enterprise Edition	StyleVision Server
XMLSpy Enterprise Edition	

## Installation et désinstallation de Gestionnaire de taxonomie

Gestionnaire de taxonomie est installé automatiquement quand vous installez d'abord une nouvelle version de l'Altova Mission Kit Enterprise Edition ou toute application activée par XBRL d'Altova (*voir la table ci-dessus*).

De même, il est supprimé automatiquement lorsque vous désinstallez la dernière application activée pour XBRL Altova depuis votre ordinateur.

## Fonctions <% TAXON-MANAGER%>

Gestionnaire de taxonomie propose les fonctions suivantes :

- Consulter les taxonomies XBRL installées sur votre ordinateur, et contrôlez si de nouvelles versions sont disponibles pour le téléchargement.
- Télécharger des versions plus récentes des taxonomies XBRL indépendamment du cycle de release des produits Altova. (Altova stocke des taxonomies en ligne et vous pouvez les télécharger via Gestionnaire de taxonomie.)
- Installer ou désinstaller une des versions d'une taxonomie donnée (ou toutes les versions si nécessaire).
- Une taxonomie XBRL peut avoir des dépendances sur d'autres taxonomies. Lorsque vous installez ou désinstallez une taxonomie particulière, Gestionnaire de taxonomie vous informe sur d'autres taxonomies dépendantes et les installera ou désinstallera également automatiquement.
- Gestionnaire de taxonomie utilise le mécanisme du [catalogue XML](#) pour mapper les références de schéma aux fichiers locaux. Dans le cas de taxonomies XBRL larges, le traitement sera plus rapide que si les taxonomies étaient à un emplacement à distance.
- Toutes les taxonomies majeures sont disponibles via Gestionnaire de taxonomie et sont régulièrement mises à jour pour les dernières versions. Ceci vous fournit une ressource unique pour gérer toutes vos taxonomies et les mettre à disposition de toutes les applications activées par XBRL d'Altova.
- Les changements réalisés dans Gestionnaire de taxonomie prennent effet pour tous les produits d'Altova sur cet appareil.

## Taxonomies XBRL personnalisées

Si vous devez travailler avec des taxonomies XBRL personnalisées qui ne sont pas incluses dans Gestionnaire de taxonomie, vous pouvez ajouter ces taxonomies à l'ensemble des packs personnalisés référencés par <% appName%>. Pour ce faire, suivez les étapes suivantes :

- *Dans les applications desktop d'Altova* : Sélectionnez la commande de menu **Outils | Options** et allez à la section *XBRL | Packages de taxonomie*. Recherchez le package ZIP de votre taxonomie XBRL personnalisée. Pour plus d'information, voir la description de cette commande dans votre documentation de produit desktop.
- *Dans les applications de serveur Altova* : Lorsque vous exécutez des commandes de la ligne de commande qui appuie les taxonomies personnalisées, fournissez l'option `--taxonomy-package of --`

`taxonomy-package-config-file`. Par exemple : Dans RaptorXML+XBRL Server, ces options sont appuyées par les commandes de validation XBRL telles que `valxbrl` ou `valxbrltaxonomy` ; dans MapForce, elles sont prises en charge par la commande d'exécution.

## Comment cela fonctionne ?

Altova stocke toutes les taxonomies XBRL utilisées dans les produits Altova en ligne. Ce référentiel est mis à jour lorsque de nouvelles versions des taxonomies sont publiées. Gestionnaire de taxonomie affiche des informations sur les dernières taxonomies disponibles lorsqu'elles sont appelées dans son formulaire GUI de même que sur CLI. Vous pouvez ensuite installer, mettre à jour ou désinstaller les taxonomies via Gestionnaire de taxonomie.

Gestionnaire de taxonomie installe également les taxonomies d'une autre manière. Sur le site web d'Altova (<https://www.altova.com/fr/taxonomy-manager>), vous pouvez sélectionner une taxonomie et ses taxonomies dépendantes que vous souhaitez installer. Le site web préparera un fichier de type `.altova_taxonomies` pour le téléchargement qui contient des informations sur la sélection de taxonomie. Lorsque vous double-cliquez sur ce fichier ou le passez à Gestionnaire de taxonomie via CLI comme argument de la commande `installer`<sup>116</sup>, Gestionnaire de taxonomie installera les taxonomies que vous avez sélectionnées.

### Cache local : suivre vos taxonomies

Toutes les informations sur les taxonomies installées sont suivies dans un répertoire cache centralisé sur votre ordinateur, situé ici :

<i>Windows</i>	C:\ProgramData\Altova\pkgs\cache
<i>Linux</i>	/var/opt/Altova/pkgs/cache
<i>macOS</i>	/var/Altova/pkgs

Ce répertoire cache est mis à jour régulièrement avec le dernier statut des taxonomies dans l'emplacement de stockage en ligne d'Altova. Ces mises à jour sont réalisées aux moments suivants :

- À chaque fois que vous lancez Gestionnaire de taxonomie.
- Lorsque vous exécutez StyleVision Server pour la première fois dans un jour donné du calendrier.
- Si StyleVision Server est ouvert plus de 24 heures, le cache est mis à jour toutes les 24 heures.
- Vous pouvez aussi mettre à jour le cache en exécutant la commande de `mise à jour`<sup>119</sup> dans l'interface de ligne de commande.

Pour cette raison, le cache permet à Gestionnaire de taxonomie de suivre continuellement vos taxonomies installées par rapport aux taxonomies disponibles en ligne sur le site web d'Altova.

### Ne modifiez pas le cache manuellement !

Le répertoire de cache local est entretenu automatiquement sur la base des taxonomies que vous installez ou désinstallez ; il ne doit pas être modifié ou supprimé manuellement. Il ne devrait pas être altéré ou supprimé manuellement. Si vous êtes amené à réinitialiser Gestionnaire de taxonomie à son état original "intact", alors, sur l'interface de la ligne de commande (CLI) : (i) exécutez la commande `reset`<sup>117</sup>, et (ii) exécutez la commande `initialize`<sup>115</sup>. (En alternative, exécutez la commande `reset` avec l'option `--i`.)

## Proxy HTTP

Vous pouvez utiliser un proxy HTTP pour les connexions Gestionnaire de taxonomie. Les paramètres Proxy du système seront utilisés.

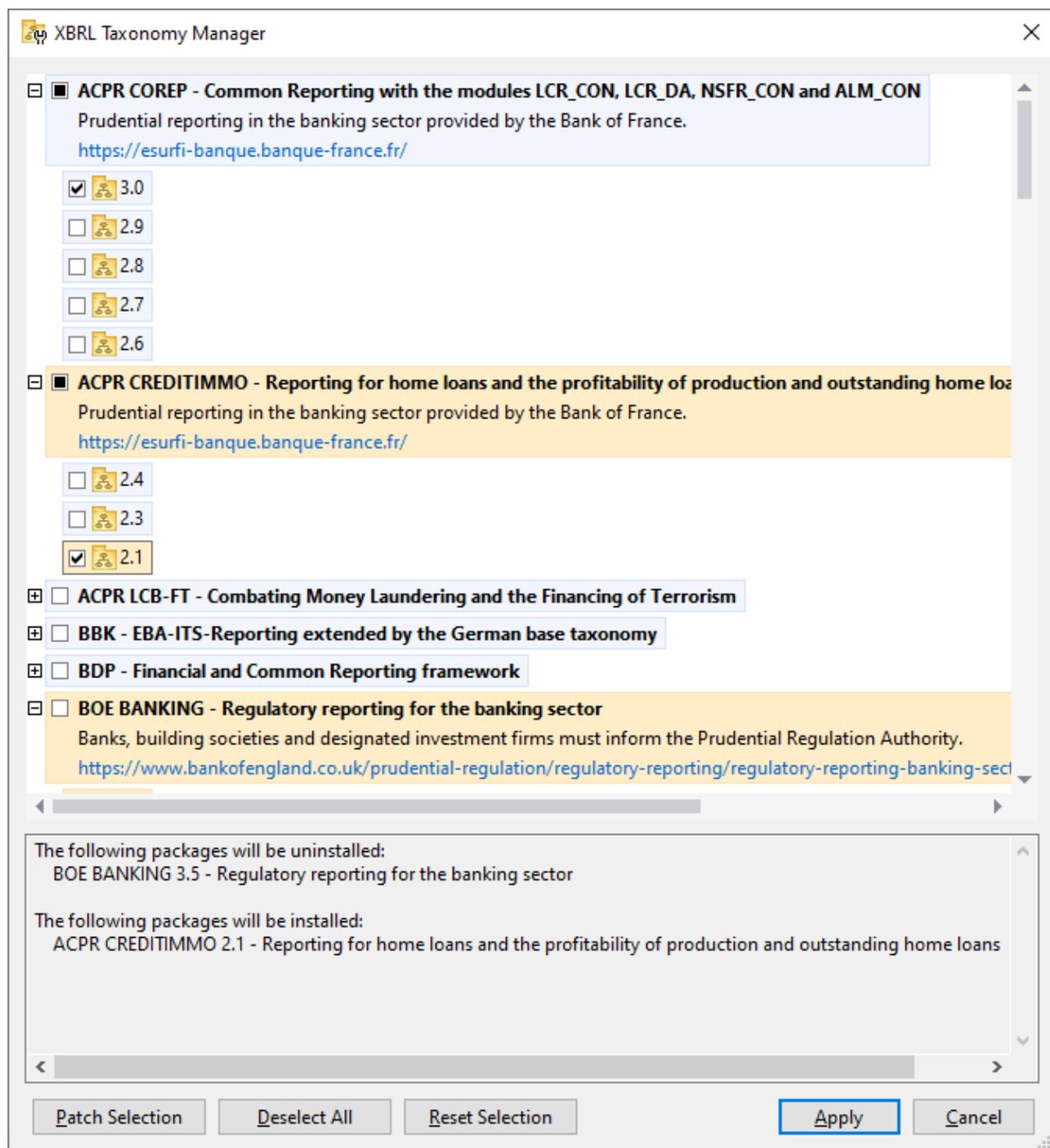
## 7.1 Exécuter le Gestionnaire de taxonomie

### Interface utilisateur graphique

Vous pouvez accéder à la GUI de Gestionnaire de taxonomie des manières suivantes :

- *Durant l'installation de StyleVision Server:* Vers la fin de la procédure d'installation, sélectionnez la case à cocher *Invoke Altova Taxonomy Manager* pour accéder au Gestionnaire de taxonomie XBRL directement. Ceci vous permettra d'installer les taxonomies au cours de la procédure d'installation de votre application Altova.
- Via le fichier `.altova_taxonomies` téléchargé du [Centre de téléchargement de la taxonomie XBRL d'Altova](#): Double-cliquez sur le fichier téléchargé pour exécuter la GUI de Gestionnaire de taxonomie, qui sera configurée pour installer les taxonomies que vous avez sélectionnées (le site web) pour installation.

Une fois que Gestionnaire de taxonomie GUI (*capture d'écran ci-dessous*) a été ouvert, les taxonomies déjà installées seront affichées telles sélectionnées. Si vous voulez installer une taxonomie supplémentaire, sélectionnez-la. Si vous voulez désinstaller une taxonomie déjà installée, désélectionnez-la. Une fois que vous avez faits vos sélections et/ou désélections, vous êtes prêts pour appliquer vos changements. Les taxonomies qui seront installées ou désinstallées seront mises en surbrillance et un message sur les modifications à venir sera posté dans le volet Messages au niveau inférieur de la fenêtre Gestionnaire de taxonomie (*voir la capture d'écran*).



## Interface de ligne de commande

Vous pouvez exécuter Gestionnaire de taxonomie depuis une interface de ligne de commande en sélectionnant son fichier exécutable, `taxonomymanager.exe`.

Le fichier `taxonomymanager.exe` est situé dans le dossier suivant :

- *Sur Windows* : `C:\ProgramData\Altova\SharedBetweenVersions`
- *Sur Linux ou macOS (application serveur uniquement)* : `%INSTALLDIR%/bin`, où `%INSTALLDIR%` est le répertoire d'installation du programme.

Vous pouvez alors utiliser toute commande de la section de référence de la commande CLI.

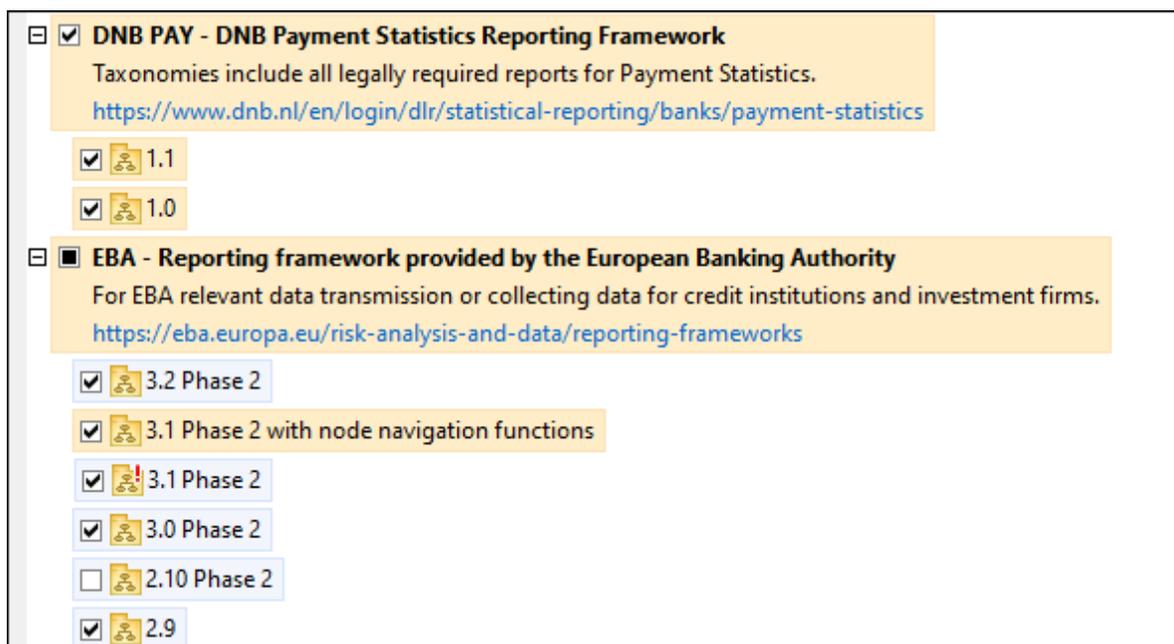
Pour afficher l'aide pour la commande, exécutez l'étape suivante :

- *Sur Windows* : `taxonomymanager.exe --help`
- *Sur Linux ou macOS (application serveur uniquement)* : `sudo ./taxonomymanager --help`

## 7.2 Catégories de statut

Gestionnaire de taxonomie catégorise les taxonomies sous sa gestion comme suit :

- *Installer taxonomies.* Celles-ci sont affichées dans la GUI avec leurs cases à cocher sélectionnées (dans la capture d'écran ci-dessous, les versions cochées ci-de la taxonomie DNB et EBA sont des taxonomies installées). Si toutes les versions de taxonomie sont sélectionnées, alors la marque de sélection est une coche. Si au moins une version est décochée, alors la coche de sélection est un carré coloré plein. Vous pouvez décocher une taxonomie installée pour la **désinstaller**.
- *Désinstaller taxonomies disponibles.* Elles sont affichées dans la GUI avec leurs cases à cocher non sélectionnées. Vous pouvez sélectionner les taxonomies que vous souhaitez **installer**.



- Les *taxonomies pouvant être mises à niveau* sont celles qui ont été revues par leurs émetteurs depuis qu'elles ont été installées. Elles sont indiquées par leur GUI par une icône  (voir la capture d'écran ci-dessus). Vous pouvez **retoucher** la taxonomie installée avec la révision disponible.

### Points à noter

- Dans la capture d'écran ci-dessus, les taxonomies DNB et quelques unes des taxonomies EBA sont cochées. Celles avec un arrière-plan bleu sont déjà installées. Celles avec un arrière-plan jaune sont désinstallées et ont été sélectionnées pour l'installation. Notez que (i) la taxonomie EBA 2.10 Phase 2 n'est pas installée et n'a pas été sélectionnée pour l'installation, (ii) la taxonomie EBA 3.1 Phase 2 a été installée, mais elle a été retouchée par son émetteur depuis qu'elle a été installée et le correctif n'a pas encore été installé.
- Lorsque vous exécutez Gestionnaire de taxonomie depuis la ligne de commande, la commande `list`<sup>116</sup> est utilisée avec différentes options pour recenser les différentes catégories de taxonomies :

<code>taxonomymanager.exe list</code>	Recense toutes les taxonomies installées et disponibles ; celles
---------------------------------------	--

	pouvant être mises à niveau sont également indiquées.
<code>taxonomymanager.exe list -i</code>	Recense les taxonomies installées uniquement ; celles pouvant être mises à niveau sont également indiquées
<code>taxonomymanager.exe list -u</code>	Recense les taxonomies pouvant être mises à niveau

**Note :** Sur Linux et macOS, use `sudo ./taxonomymanager list`

## 7.3 Retoucher ou Installer une taxonomie

### Retoucher une taxonomie installée

Occasionnellement, des taxonomies XBRL peuvent recevoir des patchs (mises à niveau ou révisions) depuis leurs émetteurs. Lorsque Gestionnaire de taxonomie détecte que des patchs sont disponibles, ceux-ci sont indiqués dans les listes de taxonomie Gestionnaire de taxonomie et vous pouvez installer des patchs rapidement.

#### Dans la GUI

Les patchs sont indiqués par l'icône . (Voir aussi la rubrique précédente sur les [catégories de statut](#)<sup>109</sup>.) Si les patchs sont disponibles, le bouton **Patch Selection** sera activé. Cliquez dessus pour sélectionner et préparer tous les patchs pour installation. Dans la GUI, l'icône de chaque taxonomie sera patchée de  à , et le volet des Messages en bas du dialogue recense les patchs qui doivent être appliqués. Lorsque vous êtes prêt pour installer des patchs sélectionnés, cliquez sur **Appliquer**. Tous les correctifs seront appliqués ensemble. Notez que si vous décochez une taxonomie marquée pour une correction, réellement vous désinstallerez cette taxonomie.

#### Sur le CLI

Pour appliquer un patch dans l'interface de ligne de commande :

1. Exécuter la commande `list -u`<sup>116</sup> . Cela liste des taxonomies lorsque des mises à niveau de patch sont disponibles.
2. Exécuter la commande `upgrade`<sup>119</sup> pour installer les patches.

### Installer une taxonomie disponible

Vous pouvez installer des taxonomies en utilisant soit la GUI Gestionnaire de taxonomie ou en envoyant Gestionnaire de taxonomie les instructions d'installation via la ligne de commande.

**Note** : Si la taxonomie actuelle référence d'autres taxonomies, les taxonomies référencées sont aussi installées.

#### Dans la GUI

Pour installer une taxonomie utilisant la GUI Gestionnaire de taxonomie GUI, sélectionnez les taxonomies que vous voulez installer et cliquez sur **Appliquer**.

Vous pouvez aussi sélectionner les taxonomies que vous voulez installer sur le [site web d'Altova](#) et générer un fichier téléchargeable `.altova_taxonomies`. Lorsque vous double-cliquez sur ce fichier, il ouvrira Gestionnaire de taxonomie avec les taxonomies que vous vouliez présélectionner. La seule chose qui vous reste à faire, c'est cliquer sur **Appliquer**.

#### Sur le CLI

Pour installer des taxonomies via la ligne de commande, exécutez la commande `installer`<sup>116</sup> :

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

où **Taxonomie** est la taxonomie (ou les taxonomies) que vous voulez installer ou un fichier `.altova_taxonomies`. Une taxonomie est référencée par un identifiant de format `<name>-<version>`. (Les

identifiants de taxonomies sont affichés quand vous exécutez la commande [list](#)<sup>116</sup>.) Vous pouvez saisir autant de taxonomies que vous le souhaitez. Pour plus de détails, voir la description de la commande [install](#)<sup>116</sup>.

**Note** : sur Linux ou macOS, utilisez la commande `sudo ./taxonomymanager`.

#### *Installer une taxonomie requise*

Lorsque vous exécutez une commande activée par XBRL dans StyleVision Server, et que StyleVision Server découvre qu'une taxonomie dont elle a besoin pour exécuter la commande n'est pas présente ou est incomplète, Gestionnaire de taxonomie sera affiché l'information sur la taxonomie manquante. Vous pouvez ensuite installer directement toute taxonomie manquante via Gestionnaire de taxonomie.

Dans le Gestionnaire de taxonomie GUI, vous pouvez consulter toutes les taxonomies précédemment installées à tout moment en exécutant Gestionnaire de taxonomie de **Outils | Gestionnaire de taxonomie**.

## 7.4 Désinstaller une taxonomie, Réinitialiser

### Désinstaller une taxonomie

Vous pouvez désinstaller des taxonomies en utilisant soit la GUI Gestionnaire de taxonomie ou en envoyant Gestionnaire de taxonomie les instructions d'installation via la ligne de commande.

**Note** : si la taxonomie que vous voulez désinstaller référence d'autres taxonomies, alors les taxonomies référencées sont également désinstallées.

#### Dans la GUI

Pour désinstaller les taxonomies utilisant la GUI Gestionnaire de taxonomie, effacez leurs cases à cocher et cliquez sur **Appliquer**. Les taxonomies sélectionnées et leurs taxonomies référencées seront désinstallées.

Pour désinstaller les taxonomies, cliquez sur **Désélectionner tout** et cliquez sur **Appliquer**.

#### Sur le CLI

Pour désinstaller des taxonomies via la ligne de commande, exécutez la commande `désinstaller` :

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

où chaque argument `Taxonomie` est la taxonomie que vous voulez désinstaller ou un fichier `.altova_taxonomies`. Une taxonomie est spécifiée par un identifiant qui a un format `<name>-<version>`. (Les identifiants de taxonomies sont affichés quand vous exécutez la commande `list`<sup>116</sup>.) Vous pouvez saisir autant de taxonomies que vous le souhaitez. Pour plus de détails, voir la description de la commande `désinstaller`<sup>118</sup>.

**Note** : sur Linux ou macOS, utilisez la commande `sudo ./taxonomymanager`.

### Réinitialiser Gestionnaire de taxonomie

Vous pouvez réinitialiser Gestionnaire de taxonomie.

- Dans la GUI, cliquez sur **Reset Selection**. Ceci réinitialise la GUI pour montrer quelles taxonomies sont actuellement installées. Toute sélection ou désélection que l'utilisateur a effectué dans la session actuelle sera annulée.
- Dans la CLI, exécutez la commande `reset`<sup>117</sup>. Ceci supprime toutes les taxonomies installées et le répertoire de mise sous cache.

Une fois avoir exécuté cette commande, vous devrez exécuter la commande `initialize`<sup>115</sup>, pour pouvoir recréer le répertoire de mise sous cache. En alternative, exécutez la commande `reset`<sup>117</sup> avec l'option `-i`.

Notez que `reset -i`<sup>117</sup> restaure l'installation originale du produit, il est recommandé d'exécuter la commande `update`<sup>119</sup> après avoir réalisé la réinitialisation. En alternative, exécutez la commande `reset`<sup>117</sup> avec les options `-i` and `-u`.

## 7.5 Interface de ligne de commande (CLI)

Pour appeler Gestionnaire de taxonomie dans la ligne de commande, vous devez connaître le chemin de l'exécutable. Par défaut, l'exécutable Gestionnaire de taxonomie est installé dans le chemin suivant :

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

**Note** : sur les systèmes Linux et macOS, une fois que vous avez changé le répertoire à celui contenant l'exécutable, vous pouvez appeler l'exécutable avec `sudo ./taxonomymanager`. Le préfixe `./` indique que l'exécutable est le répertoire actuel. Le préfixe `sudo` indique que la commande doit être exécutée avec des privilèges root.

### Syntaxe de ligne de commande

La syntaxe générale pour utiliser la ligne de commande est la suivante :

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

Dans l'extrait ci-dessus, la barre verticale `|` sépare un ensemble d'items mutuellement exclusifs. Les crochets `[]` indiquent des items optionnels. De manière générale, vous pouvez saisir le chemin d'exécutable suivi soit par les options `--h`, `--help`, ou `--version` ou par une commande. Chaque commande peut contenir des options et des arguments. La liste des commandes est décrite dans les sections suivantes.

### 7.5.1 help

Cette commande propose une aide contextuelle pour les commandes liées à l'exécutable Gestionnaire de taxonomie.

#### Syntaxe

```
<exec> help [command]
```

Où `[command]` est un argument optionnel qui spécifie un nom de commande valide.

Veillez noter les points suivants :

- Vous pouvez invoquer de l'aide une commande en saisissant La section commande suivie par `-h` ou `--help`, par exemple : `<exec> list-h`
- Si vous tapez `-h` or `--help` directement après la commande exécutable et avant une commande, vous recevrez une aide générale (pas d'aide pour la commande), par exemple : `<exec> -h list`

#### Exemple

La commande suivante affiche une aide concernant la commande `list` :

```
./taxonomymanager help list
```

## 7.5.2 info

Cette commande affiche des informations détaillées pour chacune des taxonomies fournies en tant qu'argument de `Taxonomie`. Cette information inclut le titre, la version, description, l'éditeur et toute taxonomie soumise et toute taxonomie dépendante, et mentionne si la taxonomie a été installée ou non.

### Syntaxe

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

- L'argument `Taxonomie` est le nom de la taxonomie ou une partie du nom de la taxonomie. (Pour afficher une ID de pack de taxonomie et des informations détaillées sur son statut d'installation, vous devriez utiliser la commande [list](#) <sup>116</sup>.)
- Utiliser `<exec> info -h` pour afficher l'aide de la commande.

### Exemple

La commande suivante affiche des informations détaillées concernant les taxonomies `eba-2.10` et `us-gaap-2020.0` :

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

## 7.5.3 initialize

Cette commande initialise l'environnement Gestionnaire de taxonomie. Elle crée un répertoire de cache où les informations concernant toutes les taxonomies sont stockées localement. L'initialisation est réalisée automatiquement la première fois qu'une application activée par XBRL est installée. Vous n'aurez pas besoin d'exécuter cette commande dans des circonstances normales, mais vous devrez l'exécuter généralement après la commande `reset`.

### Syntaxe

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

#### Options

La commande `initialize` accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante initialise Gestionnaire de taxonomie:

```
taxonomymanager initialize
```

## 7.5.4 install

Cette commande installe une ou plusieurs taxonomies.

### Syntaxe

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

Pour installer de multiples taxonomies, ajoutez l'argument **Taxonomy** de nombreuses fois.

L'argument **Taxonomy** est l'un des suivants :

- Un identifiant de taxonomie (avoir un format de `<name>-<version>`, par exemple : `eba-2.10`). Pour trouver les identifiants de taxonomie que vous voulez, exécutez la commande [list](#)<sup>116</sup>. Vous pouvez aussi utiliser des identifiants abrégés s'ils sont uniques, par exemple `eba`. Si vous utilisez un identifiant abrégé, alors la dernière version de cette taxonomie sera installée.
- Le chemin vers un fichier `.altova_taxonomies` téléchargé depuis le site web d'Altova. Pour information sur ces fichiers, voir [Introduction à TaxonomyManager : Comment cela fonctionne-t-il ?](#)<sup>101</sup>.

### Options

La commande `install` accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante installe les dernières taxonomies `eba` (European Banking Authority) et `us-gaap` (US Generally Accepted Accounting Principles) :

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

## 7.5.5 list

Cette commande recense les taxonomies sous la gestion de Gestionnaire de taxonomie. La liste affiche comme suit

- Toutes les taxonomies disponibles
- Les taxonomies contenant le string dans leur nom soumis comme argument `Taxonomy`
- Seules les taxonomies installées
- Seules les taxonomies qui peuvent être mises à niveau

## Syntaxe

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

Si aucun argument `Taxonomy` n'est soumis, alors toutes les taxonomies disponibles sont recensées. Autrement, les taxonomies sont recensées par des options soumises (*voir l'exemple ci-dessous*). Notez que vous pouvez soumettre l'argument `Taxonomy` de nombreuses fois.

### Options

La commande `list` accepte les options suivantes :

<code>--installed, --i</code>	Uniquement la liste des taxonomies installées. Le réglage par défaut est <b>faux</b> .
<code>--upgradeable, --u</code>	Recenser uniquement les taxonomies lorsque des mises à niveau (patches) sont disponibles. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemples

- Pour exécuter toutes les taxonomies disponibles, exécutez : `taxonomymanager list`
- Pour recenser les taxonomies installées, exécutez : `taxonomymanager list -i`
- Pour recenser tous les taxonomies qui contiennent soit "eba", soit "us-gaap" dans leur nom, exécutez `taxonomymanager list eba us-gaap` :

## 7.5.6 reset

Cette commande supprime toutes les taxonomies installées et le répertoire de mise sous cache. Vous réinitialiserez complètement votre environnement de taxonomie. Une fois avoir exécuté cette commande, vous devez exécuter la commande `initialize`<sup>115</sup>, pour pouvoir recréer le répertoire de mise sous cache. En alternative, exécuter la commande `reset` avec l'option `-i`. Puisque `reset -i` restaure l'installation originale du produit, nous vous recommandons que vous exécutiez la commande `update`<sup>119</sup> après avoir réalisé la réinitialisation et l'initialisation. En alternative, exécutez la commande `reset` avec les options `-i`<sup>115</sup> et `-u`<sup>115</sup>.

## Syntaxe

```
<exec> reset [options]
```

### Options

La commande `reset` accepte les options suivantes :

<code>--init, --i</code>	Initialiser Gestionnaire de taxonomie après le reset. Le réglage par défaut est <b>faux</b> .
<code>--update, --u</code>	Mettre à jour la liste de taxonomies disponibles dans le cache. Le réglage par défaut est <b>faux</b> .
<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemples

- Pour réinitialiser Gestionnaire de taxonomie, exécuter : `taxonomymanager reset`
- Pour réinitialiser Gestionnaire de taxonomie et l'initialiser, exécutez : `taxonomymanager reset -i`
- Pour réinitialiser Gestionnaire de taxonomie, initialiser-la et mettez à jour sa liste de taxonomie, exécutez : `taxonomymanager reset -i-u`

## 7.5.7 uninstall

Cette commande désinstalle une ou plusieurs taxonomies. Par défaut, toute taxonomie référencée par la taxonomie actuelle sera également désinstallée. Pour désinstaller uniquement la taxonomie actuelle et garder les taxonomies référencées, définir l'option `--k`.

### Syntaxe

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

Pour désinstaller de multiples taxonomies, ajoutez l'argument `Taxonomy` de nombreuses fois.

L'argument `Taxonomy` est l'un des suivants :

- Un identifiant de taxonomie (avoir un format de `<name>-<version>`, par exemple : `eba-2.10`). Pour trouver les identifiants de taxonomie qui sont installées, exécutez la commande `list -i`<sup>116</sup>. Vous pouvez aussi utiliser un nom de taxonomie abrégé s'il est unique, par exemple `eba`. Si vous utilisez un nom abrégé, alors toutes les taxonomies qui contiennent une abréviation dans leur nom seront désinstallées.
- Le chemin vers un fichier `.altova_taxonomies` téléchargé depuis le site web d'Altova. Pour information sur ces fichiers, voir [Introduction à TaxonomyManager : Comment cela fonctionne-t-il ?](#)<sup>101</sup>.

### Options

La commande `désinstaller` accepte les options suivantes :

<code>--keep-references, --k</code>	Définir cette option pour garder les taxonomies référencées. Le réglage par défaut est <b>faux</b> .
-------------------------------------	--

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante désinstalle les taxonomies `eba-2.10` et `us-gaap-2020.0` et leurs dépendances :

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

La commande suivante désinstalle la taxonomie `eba-2.10` mais pas les taxonomies qu'elle référence :

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

## 7.5.8 update

Cette commande requête la listes des taxonomies disponibles depuis le magasin en ligne et met à jour le répertoire de mise sous cache local. Vous devriez exécuter cette commande sauf si vous avez réalisé un [reset](#)<sup>117</sup> et [initialize](#)<sup>115</sup>.

### Syntaxe

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

#### Options

La commande **mise à jour** accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

## Exemple

La commande suivante met à jour le cache local avec la liste des dernières taxonomies :

```
taxonomymanager update
```

## 7.5.9 upgrade

Cette commande met à niveau toutes les taxonomies éligibles qui peuvent être mises à niveau à la dernière version *patch* disponible. Vous pouvez identifier des taxonomies à mettre à niveau en exécutant la commande [list -u](#)<sup>116</sup>.

**Note** : La commande `mettre à niveau` supprime une taxonomie dépréciée si aucune version plus récente n'est disponible.

## Syntaxe

```
<exec> upgrade [options]
```

### Options

La commande `mise à niveau` accepte les options suivantes :

<code>--silent, --s</code>	Afficher uniquement des messages d'erreur. Le réglage par défaut est <b>faux</b> .
<code>--verbose, --v</code>	Afficher des informations détaillées lors de l'exécution. Le réglage par défaut est <b>faux</b> .
<code>--help, --h</code>	Afficher l'aide pour la commande.

# Index

## A

**Altova ServiceController, 15**  
**Assigning a license to StyleVision Server on macOS, 29**  
**Attribuer une licence à StyleVision Server sur Linux, 23**  
**Attribuer une licence à StyleVision Server sur Windows, 17**

## C

**Configuration,**  
  sur Linux, 19  
  sur macOS, 26  
  sur Windows, 11  
**Configuration de StyleVision Server, 10**  
**Connexions de base de données sur Linux, 24**  
**Connexions de base de données sur macOS, 30**

## D

**Démarrer LicenseServer sur Linux, 22**  
**Démarrer LicenseServer sur macOS, 28**  
**Démarrer LicenseServer sur Windows, 15**  
**Démarrer StyleVision Server sur Linux, 22**  
**Démarrer StyleVision Server sur macOS, 28**  
**Démarrer StyleVision Server sur Windows, 15**  
**Désinstallation, 11**  
**Désinstaller, 11**

## E

**Enregistrer StyleVision Server avec LicenseServer sur Linux, 22**  
**Exigences FOP, 34**

## F

**FlowForce Server,**  
  et StyleVision Server, 8

## G

**Gérer la licence de StyleVision Server, 10**  
**Gestionnaire de schéma,**  
  aperçu CLI, 94  
  aperçu de, 82  
  Commande Aide CLI, 94  
  Commande de mise à jour CLI, 99  
  Commande de mise à niveau CLI, 99  
  Commande désinstallation CLI, 98  
  Commande Info CLI, 95  
  Commande Initialiser CLI, 95  
  Commande Installer CLI, 96  
  Commande Liste CLI, 96  
  Commande Réinitialiser CLI, 97  
  comment exécuter, 86  
  corriger un schéma, 91  
  désinstaller un schéma, 93  
  installer un schéma, 91  
  mettre à niveau un schéma, 91  
  recenser les schémas par statut dans, 89  
  réinitialiser, 93  
  statut de schémas dans, 89  
**Gestionnaire de taxonomie,**  
  aperçu CLI, 114  
  aperçu de, 101  
  Commande Aide CLI, 114  
  Commande de mise à jour CLI, 119  
  Commande de mise à niveau CLI, 119  
  Commande désinstallation CLI, 118  
  Commande Info CLI, 115  
  Commande Initialiser CLI, 115  
  Commande Installer CLI, 116  
  Commande Liste CLI, 116  
  Commande Réinitialiser CLI, 117  
  comment exécuter, 106  
  corriger une taxonomie, 111  
  désinstaller une taxonomie, 113

**Gestionnaire de taxonomie,**

- installation d'une taxonomie, 111
- mettre à niveau une taxonomie, 111
- recenser les taxonomies par statut dans, 109
- réinitialiser, 113
- statut de taxonomies dans, 109

**I****Inscrire StyleVision Server avec LicenseServer sur macOS, 29****Inscrire StyleVision Server avec LicenseServer sur Windows, 16****installation de LicenseServer sur Linux, 21****Installation de StyleVision Server, 10****Installation sur Linux, 19****Installation sur Windows, 11****Installer LicenseServer sur macOS, 28****Installer LicenseServer sur Windows, 14****Installer sur macOS, 26****Installer sur Windows Server Core, 12****J****JRE, 34****L****Licence pour StyleVision Server,**

- attribuer à Linux, 23
- Attribuer à Windows, 17

**License for StyleVision Server,**

- assigning on macOS, 29

**Linux,**

- installation sur, 19

**M****macOS,**

- installation sur, 26

**Migrer StyleVision Server vers un nouvel appareil, 33****Mise à niveau de StyleVision Server sur Windows, 32****O****OpenJDK, 34****P****Paramètres d'environnement sur Linux, 24****Paramètres d'environnement sur macOS, 30****S****StyleVision Server,**

- dans le flux de travail FlowForce, 8
- fonctionnement, 7
- fonctions, 7
- migrer vers un nouvel appareil, 33

**StyleVision Server DLL,**

- enregistrement de, 17

**V****Versions de LicenseServer, 14, 21, 28****W****Windows,**

- installation sur, 11
- mise à niveau de StyleVision Server sur, 32