

Altova StyleVision Server 2021

Manuel de l'utilisateur et de référence

Altova StyleVision Server 2021

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: 2015-2021

© 2015-2021 Altova GmbH

Table des matières

1	Introduction	6
2	Fonctionnement	8
2.1	Dans le flux de travail FlowForce.....	9
2.2	En tant que serveur autonome.....	10
3	Configuration de StyleVision Server	11
3.1	Cœurs de processeur et licences.....	12
3.2	Configuration sur Windows.....	13
3.2.1	Installation sur Windows.....	13
3.2.2	Obtention d'une licence sur Windows.....	16
3.3	Configuration sur Linux.....	20
3.3.1	Installation sur Linux.....	20
3.3.2	Obtention d'une licence sur Linux.....	23
3.3.3	Notes à propos de l'environnement.....	25
3.4	Configuration sur macOS.....	27
3.4.1	Installation sur macOS.....	27
3.4.2	Obtention d'une licence sur macOS.....	29
3.4.3	Notes à propos de l'environnement.....	31
4	Ligne de commande StyleVision Server	32
4.1	assignlicense (uniquement Windows).....	34
4.2	exportresourcestrings.....	36
4.3	generate.....	38
4.4	help.....	41
4.5	licenseserver.....	42
4.6	pdfdata.....	44

4.7	setdeflang.....	46
4.8	setfopath.....	47
4.9	verifylicense (uniquement Windows).....	49
4.10	version.....	51

5 StyleVision Server API 52

5.1	À propos de l'interface .NET	53
5.2	À propos de l'interface COM.....	54
5.3	À propos de l'interface Java.....	55
5.4	Exemples de code.....	56
5.4.1	C#	56
5.4.2	C++	57
5.4.3	VBScript.....	59
5.4.4	Visual Basic.....	60
5.4.5	Java.....	61
5.5	Référence API.....	63
5.5.1	COM et .NET.....	63
5.5.2	Java.....	70

6 Gestionnaire de taxonomie XBRL 74

6.1	Exécuter Gestionnaire de taxonomie XBRL.....	78
6.2	Installer une taxonomie.....	79
6.3	Consulter les taxonomies installées.....	82
6.4	Désinstaller une taxonomie.....	83
6.5	Interface de ligne de commande.....	84
6.5.1	help.....	85
6.5.2	info.....	86
6.5.3	initialize.....	86
6.5.4	install.....	87
6.5.5	list	88
6.5.6	reset.....	89
6.5.7	setdeflang.....	90
6.5.8	uninstall.....	91

6.5.9	update.....	91
6.5.10	upgrade.....	92

Index

93

1 Introduction

Altova StyleVision Server est une implémentation du moteur d'exécution intégré de [StyleVision d'Altova](#). Il fonctionne en tant que module du [FlowForce Server](#)⁹ d'Altova et est aussi disponible en tant que [produit serveur autonome](#)¹⁰.

StyleVision Server exécute des paquets de transformation qui ont été déployés sur un [FlowForce Server](#)⁹. Ces transformations sont lancées par [FlowForce Server](#) sur la base d'une variété de déclencheurs de temps, de fichiers ou de déclencheurs à distance programmables. De plus la fonction StyleVision Server peut être invoquée par le biais de la [ligne de commande](#)³².

Exigences du système, de l'installation et de la licence

StyleVision Server est pris en charge sur les systèmes d'exploitation suivants :

▼ Windows

Windows 7 SP1 avec mise à jour de la plateforme, Windows 8, Windows 10

▼ Windows Server

Windows Server 2008 R2 SP1 avec mise à jour de la plateforme ou plus récent

▼ Linux

- CentOS 7 ou plus récent
- RedHat 7 ou plus récent
- Debian 8 ou plus récent
- Ubuntu 16.04 LTS ou plus récent

Les bibliothèques suivantes sont exigées en tant que condition préalable pour installer et exécuter l'application. Si les paquets ci-dessous ne sont pas encore disponibles sur votre appareil Linux, vous pourrez exécuter la commande `command yum` (ou `apt-get` le cas échéant) pour les installer.

Exigé par	CentOS, RedHat	Debian	Ubuntu
LicenseServer	krb5-libs	libgssapi-krb5-2	libgssapi-krb5-2
StyleVision Server	qt5-qtbase-gui, krb5-libs	libqt5gui5, libgssapi-krb5-2	libqt5gui5, libgssapi-krb5-2

Note : Si vous prévoyez d'utiliser les fonctions Graphiques d'Altova, au moins une police doit être installée sur votre système afin de garantir que les graphiques seront rendus correctement. Afin de recenser les polices installées, utiliser, par exemple, la commande `fc-list` depuis la [bibliothèque Fontconfig](#).

▶ macOS

macOS 10.13 ou plus récent

StyleVision Server est disponible pour 32-bit et 64-bit sur les appareils Windows. Pour plus de détails concernant l'installation et la licence, voir les sections de configuration pour [Windows](#)¹³, [Linux](#)²⁰ et [macOS](#)²⁷.

Note: Si les parties remplissables d'un PDF remplissable sont manquantes lorsque le PDF est ouvert sur un système macOS, une cause probable est que Java 6 n'est pas installé sur la machine. Si cela est le cas, vous pouvez installer Java 6 depuis https://support.apple.com/kb/dl1572?locale=en_US. Si un version plus récente que Java 6 a déjà été installé sur votre système, alors l'installation de la version Java 6 plus ancienne n'aura pas d'effet sur votre travail avec la nouvelle version, qui sera la version par défaut du système.

Dernière mise à jour : 25 February 2021

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 :

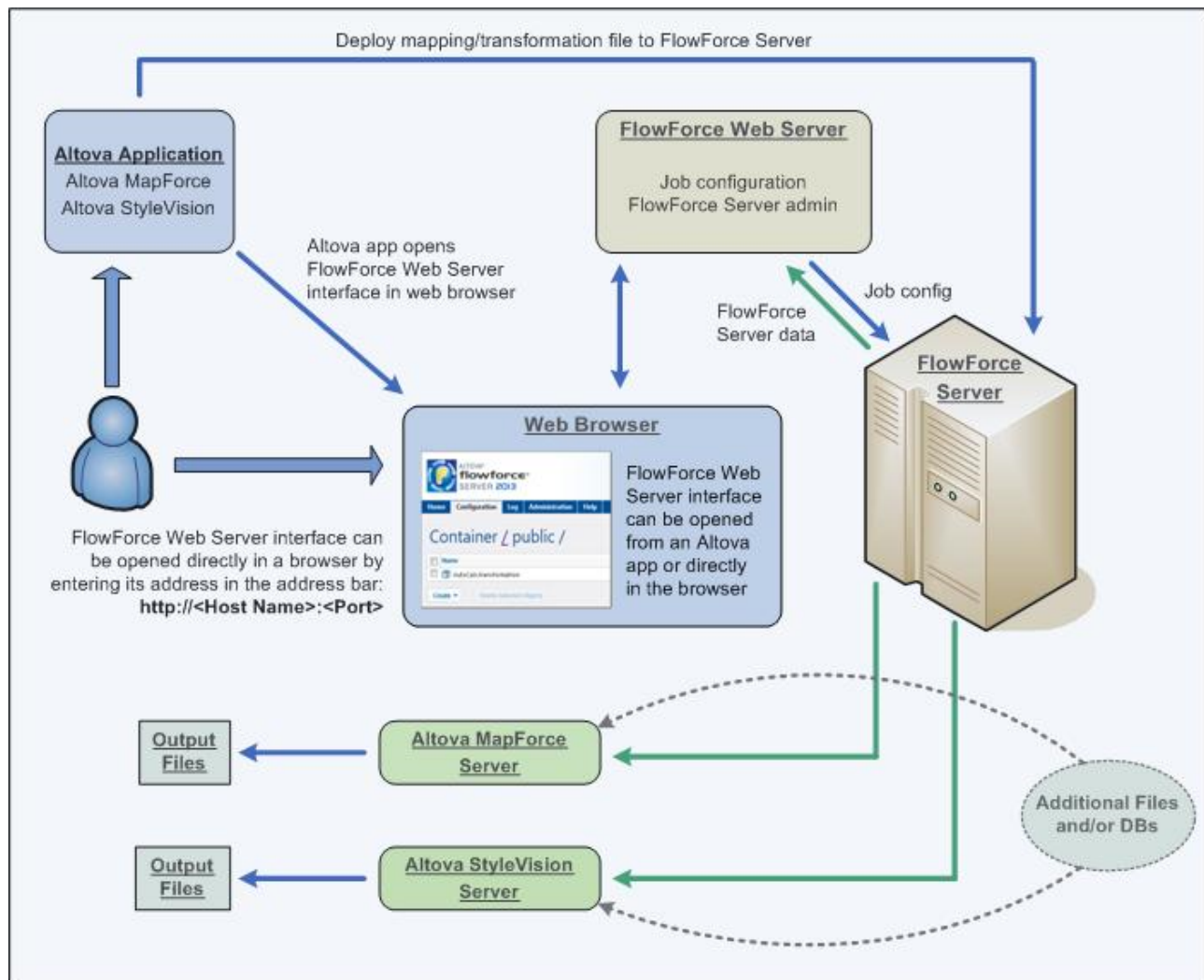
- En tant que partie prenante du [flux de travail FlowForce Altova](#)⁹. Pour plus d'information concernant [Altova FlowForce](#), consulter le [site web Altova](#).
- [En tant qu'un produit de serveur autonome](#)⁹ qui est accédé par le biais de l'interface de ligne de commande (CLI).

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](#) ³².

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](#)³².

3 Configuration de StyleVision Server

Cette section décrit des procédures pour configurer StyleVision Server. Elle décrit les points suivants :

- Informations concernant des [cœurs de processeur et licences](#) ¹²
- Installation et mise sous licence de StyleVision Server [sur les systèmes Windows](#) ¹³
- Installation et mise sous licence de StyleVision Server [sur les systèmes Linux](#) ²⁰
- Installation et mise sous licence de StyleVision Server [sur les systèmes macOS](#) ²⁷

3.1 Cœurs de processeur et licences

La mise sous licence des produits de serveur Altova se base sur le nombre de **cœurs de processeur physiques** disponibles sur l'appareil du produit (contrairement au nombre de cœurs logiques). Par exemple, un processus dual a deux cœurs, un processeur quadri-cœurs en a quatre, un processeur hexa-cœur en a six, etc. Le nombre de cœurs mis sous licence pour un produit doit être supérieur ou égal au nombre des cœurs disponibles dans ce serveur, que le serveur soit un appareil physique ou virtuel. Par exemple, si un serveur dispose de huit cœurs, vous devez acheter une licence à 8-cœurs. Vous pouvez aussi combiner les licences pour obtenir le décompte des cœurs. Donc des licence à 4-cœurs peuvent être utilisées pour un serveur à 8-cœurs au lieu d'une licence à 8-cœurs.

Si vous utilisez un serveur d'ordinateur avec un grand nombre de cœurs CPU mais que vous n'avez qu'un faible volume à traiter, vous pouvez aussi créer une machine virtuelle à laquelle un petit nombre de cœurs est attribué, et acheter une licence pour ce nombre. Il va sans dire qu'un tel déploiement présentera une vitesse de traitement plus faible que si tous les cœurs disponibles de l'ordinateur sont utilisés.

Note: Chaque licence de produit de serveur Altova peut être utilisé pour un seul appareil client (l'appareil sur lequel le produit de serveur Altova est installé) à la fois, même si la licence dispose de capacités de mise sous licence non-utilisées. Par exemple, si une licence à 10-cœurs est utilisée pour un appareil client qui a 6 cœurs CPU, les 4 cœurs restant de capacité de licence ne peuvent pas être utilisés simultanément pour un autre appareil client.

Exécution traitement unique

Si un produit de serveur Altova permet une exécution à traitement unique, une option pour *Exécution à traitement unique* sera disponible. Dans ces cas, si une licence de produit de serveur Altova est disponible pour un seul cœur est disponible dans le parc de licences, un appareil avec plusieurs cœurs peut être attribué à cette licence monocœur. Dans ce cas, l'appareil exécutera ce produit sur un seul cœur. Le traitement sera donc plus lent parce que le traitement multiple (qui est possible sur plusieurs cœurs) ne sera pas disponible. Le produit sera exécuté en mode traitement unique sur cette machine.

Pour attribuer une licence monocœur à un appareil multi-cœur, dans LicenseServer, cocher la case *limiter à l'exécution à traitement unique* pour ce produit.

Estimation des exigences de cœur

Il existe plusieurs facteurs externes qui influent sur les volumes de données et sur les délais de traitement que votre serveur est en mesure de gérer (par exemple : le matériel, la charge actuelle du CPU et l'attribution de mémoire d'autres applications en cours sur le serveur). Pour pouvoir mesurer la performance aussi précisément que possible, tester les applications dans votre environnement avec des volumes de données, et dans des conditions qui se rapprochent au maximum des véritables situations de votre entreprise.

3.2 Configuration sur Windows

Cette section décrit l'[installation](#)¹³ et l'obtention de la [licence](#)¹⁶ de StyleVision Server sur les systèmes Windows.

[Installation sur Windows](#)¹³

- [Exigences du système](#)¹³
- [Installer StyleVision Server](#)¹³
- [Altova LicenseServer](#)¹³
- [Versions du LicenseServer](#)¹³
- [Licence d'essai](#)¹³
- [Emplacement du dossier d'application](#)¹³

[Obtention de la licence sur Windows](#)¹⁶

- [Lancer ServiceController](#)¹⁶
- [Lancer LicenseServer](#)¹⁶
- [Enregistrer StyleVision Server](#)¹⁶
- [Attribuer une licence](#)¹⁶

Note: À partir de la version 2021, une version 32-bit de StyleVision Server ne peut pas être installée par-dessus une version 64-bit, ou une version 64-bit sur une version 32-bit. Vous devez soit (i) supprimer l'ancienne version avant d'installer la nouvelle, ou (ii) mettre à niveau une version plus récente qui est la même version de bit que votre installation plus ancienne.

3.2.1 Installation sur Windows

StyleVision Server est disponible pour une installation sur les systèmes Windows. La procédure d'installation et de configuration est décrite ci-dessous.

▼ Exigences du système

▼ [Windows](#)

Windows 7 SP1 avec mise à jour de la plateforme, Windows 8, Windows 10

▼ [Windows Server](#)

Windows Server 2008 R2 SP1 avec mise à jour de la plateforme ou plus récent

▼ Installer StyleVision Server

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

- En tant que produit de serveur autonome séparé appelé StyleVision Server. Pour installer StyleVision Server, téléchargez et exécutez le programme d'installation StyleVision Server. Suivez les instructions sur l'écran.
- [En tant que partie intégrante du paquet d'installation FlowForce Server](#). Pour installer StyleVision Server en tant que partie intégrante du paquet [FlowForce Server](#), téléchargez et exécutez le

programme d'installation FlowForce Server. Suivez les instructions sur l'écran et veillez à cocher l'option de l'installation de StyleVision Server.

Les programmes d'installation de StyleVision Server et de [FlowForce Server](http://www.altova.com/fr/download.html) sont disponibles depuis le Centre de téléchargement Altova (<http://www.altova.com/fr/download.html>).

Une fois l'installation effectuée, le programme d'exécution de StyleVision Server sera situé par défaut sous :

```
<ProgramFilesFolder>\Altova\StyleVisionServer2021\bin\StyleVisionServer.exe
```

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

Installation sur Windows Server Core

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

1. Télécharger le programme d'installation StyleVision Server exécutable depuis le site web de Altova. Ce fichier est appelé `stylevisionserver<version>.exe`. Assurez-vous de choisir le programme exécutable correspondant 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 décompresse le fichier `.msi` vers le même dossier que le programme d'installation exécutable.
3. Copiez le fichier décompressé `.msi` sur l'appareil Windows Server Core.
4. Si vous mettez à jour une version antérieure de StyleVision Server, désactivez StyleVision Server avant d'exécuter la prochaine étape.
5. Utilisez le fichier `.msi` pour l'installation en exécutant la commande `msiexec /i stylevisionserver.msi`. Ceci lance l'installation sur Windows Server Core.

Gardez le(s) fichier(s) `.msi` file/s à un emplacement sûr. Vous en aurez besoin plus tard pour désinstaller, réparer ou modifier votre/vos installation/s.

Pour tester la valeur retour de votre installation, exécutez un script similaire au suivant. 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 journal du processus d'installation, exécutez :

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

Pour modifier l'installation, exécutez :

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

Pour réparer l'installation, exécutez :

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

Pour désinstaller StyleVision Server, exécutez :

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

Pour désinstaller StyleVision Server silencieusement et faire rapport du résultat détaillé dans le fichier journal :

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

Note : sur Windows Server Core, la fonctionnalité des graphiques et du code barre de StyleVision Server ne sera pas disponible. Pour installer les taxonomies, utilisez le Taxonomy Package Manager via la ligne de commande. Voir le manuel StyleVision Server pour toute information liée à la procédure.

▼ Altova LicenseServer

- Pour que StyleVision Server puisse fonctionner, vous devez obtenir une licence pour votre réseau par le biais de Altova LicenseServer.
- Lors de l'installation de StyleVision Server ou de [FlowForce Server](#) sur des systèmes Windows, une option est disponible vous permettant de télécharger et d'installer Altova LicenseServer avec StyleVision Server ou le [FlowForce Server](#).
- Si Altova LicenseServer est déjà installé sur votre réseau, il ne sera pas nécessaire d'en installer un autre, sauf si une version plus récente d'Altova LicenseServer est nécessaire. (*Voir le point suivant, [versions de LicenseServer](#).*)
- Au cours du processus d'installation de StyleVision Server ou de [FlowForce Server](#), cochez ou décochez l'option d'installation d'Altova LicenseServer selon le cas. Notez les points suivants :
 - a. Si vous n'avez pas encore installé Altova LicenseServer, ne touchez pas aux paramètres par défaut. L'Assistant installera la dernière version d'Altova LicenseServer sur l'ordinateur sur lequel vous exécutez l'assistant.
 - b. Si vous n'avez pas encore installé Altova LicenseServer et que vous souhaitez installer Altova LicenseServer sur un autre ordinateur, cliquer pour supprimer la case à cocher **Installer Altova LicenseServer sur cet appareil**, puis choisir **S'enregistrer plus tard**. Dans ce cas, vous devrez installer Altova LicenseServer et enregistrer StyleVision Server séparément.
 - c. Si Altova LicenseServer a déjà été installé sur votre ordinateur mais que sa version est moins élevée que celle prescrite par l'assistant à l'installation, ne touchez pas aux paramètres par défaut. Dans ce cas, l'assistant à l'installation mettra automatiquement à niveau votre version LicenseServer avec celle indiquée dans le dialogue. Veuillez noter que l'enregistrement existant et les informations de licence seront préservées après la mise à niveau.
 - d. Si Altova LicenseServer a déjà été installé sur votre ordinateur ou sur votre réseau, et s'il a la même version que celle indiquée par l'assistant, procédez comme suit :
 - i. cliquer pour supprimer la case à cocher **Installer Altova LicenseServer sur cet appareil**.
 - ii. sous **Enregistrer ce produit avec**, choisir l'instance d'Altova LicenseServer sur laquelle vous souhaitez enregistrer StyleVision Server, ou choisir **S'enregistrer plus tard**. Notez que vous pouvez toujours sélectionner **S'enregistrer plus tard** si vous voulez ignorer les associations de LicenseServer et poursuivre l'installation de StyleVision Server.

Voir la section [Obtention de la licence sur Windows](#) ¹⁶ pour plus d'informations à propos de l'enregistrement de la licence StyleVision Server avec Altova LicenseServer.

▼ Versions du LicenseServer

- Une licence doit être délivrée pour les produits de serveur Altova, soit avec la version LicenseServer convenant à la version du StyleVision Server installée, soit avec une version ultérieure de LicenseServer.
- La version LicenseServer qui convient pour une version particulière de StyleVision Server est affichée pendant l'installation de StyleVision Server. Vous pouvez installer cette version de LicenseServer avec StyleVision Server, ou bien vous pouvez installer LicenseServer séparément.

- Avant d'installer une version plus récente de LicenseServer, toute version plus ancienne doit être désinstallée. Le programme d'installation LicenseServer effectuera cette procédure automatiquement si elle détecte une version plus ancienne.
- Les versions LicenseServer sont rétrocompatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- Si vous installez une nouvelle version de StyleVision Server et si votre version installée de LicenseServer est plus ancienne que le LicenseServer approprié, installer la dernière version disponible du site web Altova.
- Au moment de la désinstallation de LicenseServer, toutes les informations d'enregistrement et d'obtention de la licence contenues dans la version plus ancienne de LicenseServer seront enregistrées dans une base de données sur la machine de votre serveur. Ces données seront importées automatiquement dans la version la plus récente lorsque la version la plus récente est installée.
- Le numéro de version du LicenseServer actuellement installé est indiqué au bas de la page de configuration de LicenseServer (tous les onglets).

Version actuelle : 3.6

▼ Licence d'essai

Au cours du processus d'installation, vous serez invité à demander une licence d'essai de 30 jours pour StyleVision Server. Une fois la demande renvoyée, la licence d'essai vous sera envoyée sur l'adresse électronique que vous aurez indiquée pendant l'inscription.

▼ Emplacement du dossier d'application

L'application sera installée dans le dossier suivant :

Windows 7, 8, 10	C:\Program Files\Altova\
Version 32 bit sur SE 64-bit	C:\Program Files (x86)\Altova\

3.2.2 Obtention d'une licence sur Windows

StyleVision Server doit être mis sous licence avec un Altova LicenseServer. L'obtention de la licence est un processus en deux étapes :

1. **Enregistrer StyleVision Server** avec LicenseServer. L'enregistrement s'effectue depuis StyleVision Server.
2. **Attribuer une licence** à StyleVision Server depuis LicenseServer. Télécharger la dernière version de LicenseServer depuis le [site Internet d'Altova](#), et l'installer sur votre appareil local ou un appareil sur votre réseau.

Les étapes à suivre sont indiquées ci-dessous en bref, voir le [manuel d'utilisateur LicenseServer](#) sur le [site Internet d'Altova](#).

▼ Lancer ServiceController

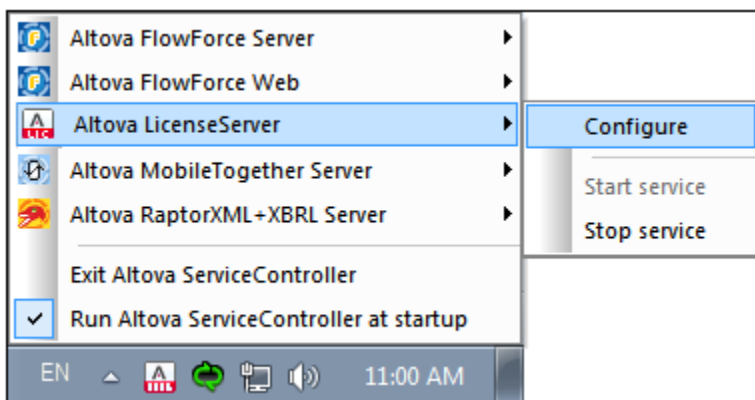
Altova ServiceController est démarré pour lancer Altova LicenseServer.

Altova ServiceController (ServiceController en abrégé) est une application permettant de lancer, d'arrêter et de configurer confortablement les services Altova **sur les systèmes Windows**.

ServiceController est installé avec l'Altova LicenseServer et avec les *produits de serveur Altova installés en tant que services*(FlowForce Server, RaptorXML(+XBRL) Server et MobileTogether Server). Il peut être démarré en cliquant sur **Start | Altova LicenseServer | Altova ServiceController**. (Cette commande est également disponible dans les dossiers de menu **Start** des *produits de serveur Altova qui sont installés en tant que services*(FlowForce Server, RaptorXML(+XBRL) Server et MobileTogether Server).) Une fois que ServiceController a été démarré, il peut être accédé depuis la zone de notification (*capture d'écran ci-dessous*).

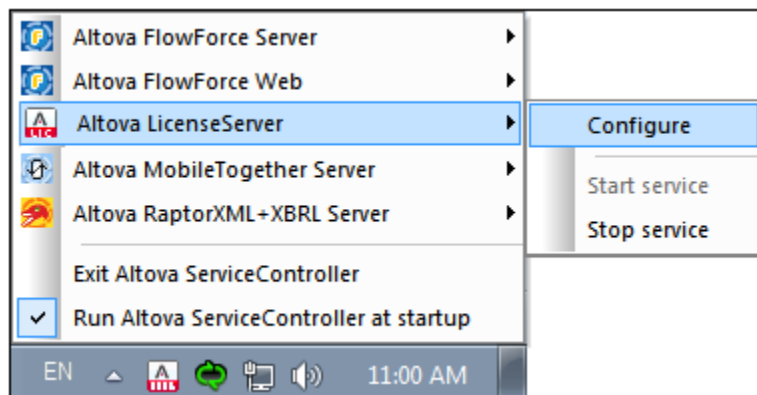


Pour spécifier que ServiceController démarre automatiquement lors de la connexion au système, cliquer sur l'icône **ServiceController** dans la zone de notification pour afficher le menu **ServiceController** (*capture d'écran ci-dessous*) et basculer sur la commande **Run Altova ServiceController at 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 capture d'écran ci-dessous*), cliquer sur **Exit Altova ServiceController**.



▼ Lancer LicenseServer

Pour démarrer le LicenseServer, cliquer sur l'icône **ServiceController** dans la zone de notification, survoler au-dessus d'**Altova LicenseServer** dans le menu qui s'ouvre (*voir capture d'écran ci-dessous*) et choisir **Start Service** depuis le sous-menu LicenseServer. Si LicenseServer est déjà en cours d'exécution, l'option *Start Service* sera désactivée.



▼ Enregistrer StyleVision Server

Pour enregistrer StyleVision Server depuis l'interface de ligne de commande, utiliser la commande `licenseserver` :

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

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

```
StyleVisionServer licenseserver localhost
```

Si StyleVision Server a été installé dans le cadre d'une installation [FlowForce Server](#), l'enregistrement du FlowForce Server avec LicenseServer entraînera automatiquement aussi l'enregistrement de StyleVision Server. En principe : (i) Lancer Altova FlowForce Web en tant que service via ServiceController (*voir point précédent*) ; (ii) Saisir votre mot de passe pour accéder à la page de Configuration ; (iii) Choisir le nom ou l'adresse du LicenseServer et cliquer sur **Register with LicenseServer**. Pour plus d'informations, voir Enregistrer FlowForce Server.

Une fois l'enregistrement effectué avec succès, rendez-vous à l'onglet Client Management de la page de configuration de LicenseServer pour attribuer une licence à StyleVision Server.

▼ Attribuer une licence

Une fois l'enregistrement de StyleVision Server effectué avec succès, il sera compris dans l'onglet Client Management de la page de configuration du LicenseServer. Rendez-vous dans cette section 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 de 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 (l'appareil sur lequel le produit de serveur Altova est installé), même si la licence a une capacité de licence n'est pas utilisée. 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.

3.3 Configuration sur Linux

Cette section décrit l'[installation](#)²⁰ et l'obtention de la [licence](#)²³ de StyleVision Server sur des systèmes Linux (Debian, Ubuntu, CentOS, RedHat).

[Installation sur Linux](#)²⁰

- [Exigences du système](#)²⁰
- [Désinstaller les anciennes versions des produits de serveurs Altova](#)²⁰
- [Télécharger le paquet Linux](#)²⁰
- [Installer StyleVision Server](#)²⁰
- [Altova LicenseServer](#)²⁰
- [Versions de LicenseServer](#)²⁰

[Obtention de la licence sur Linux](#)²³

- [Lancer LicenseServer](#)²³
- [Enregistrer StyleVision Server](#)²³
- [Attribuer une licence](#)²⁴

[Notes à propos de l'environnement](#)²⁵

3.3.1 Installation sur Linux

StyleVision Server est disponible pour une installation sur les systèmes Linux. La procédure d'installation et de configuration est décrite ci-dessous.

▼ Exigences du système

▼ *Linux*

- CentOS 7 ou plus récent
- RedHat 7 ou plus récent
- Debian 8 ou plus récent
- Ubuntu 16.04 LTS ou plus récent

Les bibliothèques suivantes sont exigées en tant que condition préalable pour installer et exécuter l'application. Si les paquets ci-dessous ne sont pas encore disponibles sur votre appareil Linux, vous pourrez exécuter la commande `command yum` (ou `apt-get` le cas échéant) pour les installer.

Exigé par	CentOS, RedHat	Debian	Ubuntu
LicenseServer	krb5-libs	libgssapi-krb5-2	libgssapi-krb5-2
StyleVision Server	qt5-qtbase-gui, krb5-libs	libqt5gui5, libgssapi-krb5-2	libqt5gui5, libgssapi-krb5-2

Note : Si vous prévoyez d'utiliser les fonctions Graphiques d'Altova, au moins une police doit être installée sur votre système afin de garantir que les graphiques seront rendus correctement. Afin de recenser les polices installées, utiliser, par exemple, la commande `fc-list` depuis la [bibliothèque Fontconfig](#).

▼ Intégration du serveur FlowForce

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

```
cp /opt/Altova/StyleVisionServer2021/etc/*.tool /opt/Altova/FlowForceServer2021/tools
```

Cette commande copie le fichier **.tool** depuis le répertoire **/etc** de StyleVision Server vers le répertoire **/tools** de FlowForce Server. Le fichier **.tool** est nécessité par FlowForce Server. Il contient le chemin vers le programme exécutable StyleVision Server. Si vous installez FlowForce Server avant d'installer StyleVision Server, vous n'aurez pas besoin d'exécuter cette commande.

▼ Désinstaller les anciennes versions des produits de serveurs Altova

Suivez les étapes suivantes pour procéder à une désinstallation. Dans l'interface de ligne de commande Linux (CLI), vous pouvez vérifier quels sont les produits de serveur Altova installés avec les commandes suivantes :

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

Si StyleVision Server n'est pas installé, poursuivez l'installation comme documenté ci-dessous dans *Installer StyleVision Server*.

Si StyleVision Server est installé et vous souhaitez installer une version plus récente de StyleVision Server, désinstaller l'ancienne version avec la commande ::

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

Si vous devez désinstaller une ancienne version de Altova LicenseServer, procédez comme suit :

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

▼ Télécharger le paquet Linux

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

Distribution	Extension du paquet
Debian	.deb
Ubuntu	.deb
CentOS	.rpm
RedHat	.rpm

Après le téléchargement du paquet Linux, le copier dans un répertoire du système Linux. Puisqu'il vous faudra Altova LicenseServer pour pouvoir exécuter StyleVision Server, vous pouvez éventuellement télécharger LicenseServer depuis le [site web Altova](#) au moment de télécharger StyleVision Server, au lieu de le télécharger plus tard.

Dans les systèmes Debian et Ubuntu, il se peut que StyleVision Server apparaisse encore dans la liste des produits installés après qu'il ait été installé. Dans ce cas, exécuter la commande `purge` pour supprimer StyleVision Server de la liste. Vous pouvez aussi utiliser la commande `purge` à la place de la commande `remove` recensée ci-dessus.

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

▼ Installer StyleVision Server

Dans une fenêtre de terminal, passez au répertoire dans lequel vous avez copié le paquet Linux. Par exemple, si vous l'avez copié dans un répertoire d'utilisateur appelé `MyAltova` (situé par exemple dans le répertoire `/home/User`), passez à ce répertoire comme suit :

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

Installer StyleVision Server avec la commande suivante :

```
[Debian]: sudo dpkg --install stylevisionserver-2021-debian.deb
[Ubuntu]: sudo dpkg --install stylevisionserver-2021-ubuntu.deb
[CentOS]: sudo rpm -ivh stylevisionserver-2021-1.x86_64.rpm
[RedHat]: sudo rpm -ivh stylevisionserver-2021-1.x86_64.rpm
```

Le paquet StyleVision Server sera installé dans le dossier :

```
/opt/Altova/StyleVisionServer2021
```

▼ Altova LicenseServer

Pour faire fonctionner tout produit Altova Server, y compris StyleVision Server, ce produit de serveur doit posséder une licence via Altova LicenseServer sur votre réseau.

Sur des systèmes Linux, Altova LicenseServer devra être installé séparément. Télécharger LicenseServer depuis le [site web Altova](#) et copier le paquet dans un répertoire sur le système Linux. L'installer de la même manière que StyleVision Server (*voir étape précédente*).

```
[Debian]: sudo dpkg --install licenseserver-3.6-debian.deb
[Ubuntu]: sudo dpkg --install licenseserver-3.6-ubuntu.deb
[CentOS]: sudo rpm -ivh licenseserver-3.6-1.x86_64.rpm
[RedHat]: sudo rpm -ivh licenseserver-3.6-1.x86_64.rpm
```

Le paquet LicenseServer sera installé dans :

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

Pour plus d'informations concernant l'enregistrement de StyleVision Server avec Altova LicenseServer et l'obtention d'une licence, voir la section, [Obtention de la licence sur Linux](#)²³. Voir aussi la [documentation LicenseServer](#) pour plus de détails

▼ Versions de LicenseServer

- Une licence doit être délivrée pour les produits de serveur Altova, soit avec la version LicenseServer convenant à la version du StyleVision Server installée, soit avec une version

ultérieure de LicenseServer.

- La version LicenseServer qui convient pour une version particulière de StyleVision Server est affichée pendant l'installation de StyleVision Server. Vous pouvez installer cette version de LicenseServer avec StyleVision Server, ou bien vous pouvez installer LicenseServer séparément.
- Avant d'installer une version plus récente de LicenseServer, toute version plus ancienne doit être désinstallée. Le programme d'installation LicenseServer effectuera cette procédure automatiquement si elle détecte une version plus ancienne.
- Les versions LicenseServer sont rétrocompatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- Si vous installez une nouvelle version de StyleVision Server et si votre version installée de LicenseServer est plus ancienne que le LicenseServer approprié, installer la dernière version disponible du site web Altova.
- Au moment de la désinstallation de LicenseServer, toutes les informations d'enregistrement et d'obtention de la licence contenues dans la version plus ancienne de LicenseServer seront enregistrées dans une base de données sur la machine de votre serveur. Ces données seront importées automatiquement dans la version la plus récente lorsque la version la plus récente est installée.
- Le numéro de version du LicenseServer actuellement installé est indiqué au bas de la page de configuration de LicenseServer (tous les onglets).

Version actuelle : 3.6

3.3.2 Obtention d'une licence sur Linux

StyleVision Server doit être mis sous licence avec un Altova LicenseServer. L'obtention de la licence est un processus en deux étapes :

1. **Enregistrer StyleVision Server** avec LicenseServer. L'enregistrement s'effectue depuis StyleVision Server.
2. **Attribuer une licence** à StyleVision Server depuis LicenseServer. Télécharger la dernière version de LicenseServer depuis le [site Internet d'Altova](#), et l'installer sur votre appareil local ou un appareil sur votre réseau.

Les étapes à suivre sont indiquées ci-dessous en bref, voir le [manuel d'utilisateur LicenseServer](#) sur le [site Internet d'Altova](#).

▼ Lancer LicenseServer

Pour enregistrer et délivrer correctement une licence pour StyleVision Server avec LicenseServer, celui-ci doit fonctionner en tant que démon sur le réseau. Lancer LicenseServer en tant que démon avec la commande suivante :

[< Debian 8]	<code>sudo /etc/init.d/licenseserver start</code>
[≥ Debian 8]	<code>sudo systemctl start licenseserver</code>
[< CentOS 7]	<code>sudo initctl start licenseserver</code>
[≥ CentOS 7]	<code>sudo systemctl start licenseserver</code>
[< Ubuntu 15]	<code>sudo initctl start licenseserver</code>

[≥ Ubuntu 15]	<code>sudo systemctl start licenseserver</code>
[RedHat]	<code>sudo initctl start licenseserver</code>

Si, à tout moment, vous souhaitez arrêter le LicenseServer, remplacer `start` avec `stop` dans la commande ci-dessus. Par exemple :

```
sudo /etc/init.d/licenseserver stop
```

▼ Enregistrer StyleVision Server

Pour enregistrer StyleVision Server depuis l'interface à ligne de commande, utiliser la commande `licenseserver` :

```
sudo /opt/Altova/StyleVisionServer2021/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/StyleVisionServer2021/bin/stylevisionserver licenseserver  
localhost
```

Dans la commande ci-dessus, `localhost` est le nom du serveur sur lequel LicenseServer est installé. Veuillez aussi noter que l'emplacement du programme d'exécution StyleVision Server est :

```
/opt/Altova/StyleVisionServer2021/bin/
```

Une fois l'enregistrement effectué avec succès, rendez-vous sur l'onglet Client Management de la page de configuration de LicenseServer pour attribuer une licence à StyleVision Server.

▼ Attribuer une licence

Une fois l'enregistrement de StyleVision Server effectué avec succès, il sera compris dans l'onglet Client Management de la page de configuration du LicenseServer. Rendez-vous dans cette section 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 de 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 (l'appareil sur lequel le produit de serveur Altova est installé), même si la licence a une capacité de licence n'est pas utilisée. 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.

3.3.3 Notes à propos de l'environnement

Dossiers

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

☐ Installation root

`/opt/Altova/StyleVisionServer2021/`

☐ Fichiers de licence

`/var/opt/Altova/StyleVisionServer`

☐ Paramètres d'environnement

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

Le fichier de paramètres d'environnement (généralement appelé `jdbc.sh`) est exécuté lors du démarrage du système. Les définitions s'y trouvant doivent être spécifiques à votre environnement. Le chemin d'exemple ci-dessus sert uniquement de guide général.

Note : Le fichier de paramètres d'environnement définit les variables pour **tous les utilisateurs** dans le système. Soyez donc prudent lors de la modification des paramètres. Par exemple, si vous modifiez un chemin de classe dans ce fichier, les modifications seront appliquées sur tout le système. Si vous souhaitez effectuer des changements uniquement pour StyleVision Server, vous devrez envisager l'utilisation d'un fichier d'unité (vous trouverez des explications dans la section *Connexions JDBC* ci-dessous).

Connexions de base de données

Sur Linu, 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 sauf pour Microsoft Access
- Connexions natives — Actuellement disponibles pour les bases de données SQLite et PostgreSQL

Si vous utilisez JDBC, veuillez noter les points suivants :

- Java Runtime Environment ou SDK doit être installé.
- Les pilotes JDBC de la base de données cible doivent être installés.
- Les variables d'environnement suivantes doivent être configurées correctement pour votre environnement:
 - `CLASSPATH`: pour trouver les fichiers jar qui se connectent à la base de données JDBC ; les fichiers jar peuvent être saisis soit dans (i) un script exécutable (comme `jdbc.sh`) qui est exécuté au lancement du système, ou (ii) un fichier d'unité qui est exécuté lorsque StyleVision Server est lancé en tant que service. L'utilisation d'un fichier d'unité pour spécifier l'emplacement des fichiers jar présente l'avantage de localiser les fichiers requis par les connexions JDBC StyleVision Server

sans que vous ayez à modifier la configuration existante du système. Un fichier d'unité est listé ci-dessous.

- o PATH: pour trouver le JRE, mais n'est pas forcément nécessaire, dépend de l'installation
- o JAVA_HOME: si nécessaire, selon l'installation.

Liste des 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 soulignées en bleu sont spécifiques à l'environnement et devront être ajustées pour se conformer à votre 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
/jdbc/oracle/xmlparserv2.jar:/usr/local/jdbc/postgre/postgresql-9.0-
801.jdbc4.jar:/usr/local/jdbc/mssql/sqljdbc4.jar:/usr/local/jdbc/ibseries/lib/jt400.ja
r:/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/
jconn7/jconn4.jar:/usr/local/jdbc/db2/db2jcc.jar:/usr/local/jdbc/db2/db2jcc_license_c
u.jar:./:
```

3.4 Configuration sur macOS

Cette section décrit l'[installation](#)¹³ et l'obtention de la [licence](#)¹⁶ de StyleVision Server sur les systèmes macOS.

[Installation sur macOS](#)²⁷

- [Exigences du système](#)²⁷
- [Désinstaller les anciennes versions des produits de serveurs Altova](#)²⁷
- [Télécharger le paquet macOS](#)²⁷
- [Installer StyleVision Server](#)²⁷
- [Altova LicenseServer](#)²⁷
- [Versions de LicenseServer](#)²⁷

[Obtention de la licence sur macOS](#)²⁹

- [Lancer LicenseServer](#)²⁹
- [Enregistrer StyleVision Server](#)²⁹
- [Attribuer une licence](#)²⁹

[Notes à propos de l'environnement](#)³¹

3.4.1 Installation sur macOS

StyleVision Server est disponible pour une installation sur les systèmes macOS. La procédure d'installation et de configuration est décrite ci-dessous.

▼ Exigences du système

▶ macOS

macOS 10.13 ou plus récent

▼ Intégration du serveur FlowForce

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

```
cp /usr/local/Altova/StyleVisionServer2021/etc/*.tool /usr/local/Altova/FlowForceServer2021/tools
```

Cette commande copie le fichier **.tool** depuis le répertoire **/etc** de StyleVision Server vers le répertoire **/tools** de FlowForce Server. Le fichier **.tool** est requis par FlowForce Server ; il contient le chemin vers le programme exécutable StyleVision Server. Il n'est pas nécessaire d'exécuter cette commande si vous installez FlowForce Server avant d'installer StyleVision Server.

▼ Désinstaller les anciennes versions des produits de serveurs Altova

Dans le dossier d'Application dans le Finder, cliquez avec le bouton de droite sur l'icône de StyleVision Server et choisissez **Déplacer dans la corbeille**. L'application sera déplacée dans la Corbeille. Néanmoins, vous devrez supprimer l'application du dossier `usr`. Utilisez la commande :

```
sudo rm -rf /usr/local/Altova/StyleVisionServer2021/
```

Si vous souhaitez désinstaller une ancienne version d'Altova LicenseServer, vous devrez d'abord stopper son exécution en tant que service. Utilisez la commande suivante :

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

Pour vérifier si le service a été interrompu, ouvrir le Moniteur d'activité dans Finder et s'assurer que LicenseServer ne se trouve pas dans la liste. Ensuite poursuivre le processus de désinstallation tel que décrit ci-dessus pour StyleVision Server.

▼ Télécharger le fichier image disque

Télécharger le fichier image disque (.dmg) depuis le site Internet Altova (<http://www.altova.com/fr/download.html>).

▼ Installer StyleVision Server

Cliquer pour ouvrir l'image disque (.dmg) téléchargée. Le programme d'installation de StyleVision Server apparaît en tant que nouveau disque virtuel sur votre ordinateur. Sur ce nouveau disque virtuel, double-cliquer le paquet d'installation (.pkg). Suivre les étapes de l'assistant d'installation. Les explications sont simples à comprendre et contiennent une étape au cours de laquelle vous devrez confirmer les accords de licence avant de pouvoir continuer. Pour éjecter le disque après l'installation, cliquer avec le bouton de droite et sélectionner **Éjecter**.

Le paquet StyleVision Server sera installé dans le dossier :

```
/usr/local/Altova/StyleVisionServer2021 (fichiers binaires)  
/var/Altova/StyleVisionServer (fichiers de données : base de données et journaux)
```

▼ Altova LicenseServer

Pour faire fonctionner tout produit Altova Server, y compris StyleVision Server, ce produit de serveur doit posséder une licence via un Altova LicenseServer sur votre réseau.

Le paquet d'installation Altova LicenseServer est disponible sur le disque virtuel que vous avez monté dans l'étape précédente. Pour installer Altova LicenseServer, double-cliquer sur le paquet d'installation compris dans le disque virtuel et suivre les instructions sur l'écran. Vous devrez accepter l'accord de licence pour poursuivre l'installation.

Altova LicenseServer peut également être téléchargé et installé séparément depuis le site Internet Altova (<http://www.altova.com/fr/download.html>).

Le paquet LicenseServer sera installé dans le dossier :

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

Pour plus d'informations concernant l'enregistrement de StyleVision Server avec Altova LicenseServer et l'obtention d'une licence, voir la section, [Obtention d'une licence sur macOS](#) ²⁸.

▼ LicenseServer versions

- Une licence doit être délivrée pour les produits de serveur Altova, soit avec la version LicenseServer convenant à la version du StyleVision Server installée, soit avec une version ultérieure de LicenseServer.
- La version LicenseServer qui convient pour une version particulière de StyleVision Server est affichée pendant l'installation de StyleVision Server. Vous pouvez installer cette version de LicenseServer avec StyleVision Server, ou bien vous pouvez installer LicenseServer séparément.
- Avant d'installer une version plus récente de LicenseServer, toute version plus ancienne doit être désinstallée. Le programme d'installation LicenseServer effectuera cette procédure automatiquement si elle détecte une version plus ancienne.
- Les versions LicenseServer sont rétrocompatibles. Elles fonctionneront avec des versions plus anciennes de StyleVision Server.
- Si vous installez une nouvelle version de StyleVision Server et si votre version installée de LicenseServer est plus ancienne que le LicenseServer approprié, installer la dernière version disponible du site web Altova.
- Au moment de la désinstallation de LicenseServer, toutes les informations d'enregistrement et d'obtention de la licence contenues dans la version plus ancienne de LicenseServer seront enregistrées dans une base de données sur la machine de votre serveur. Ces données seront importées automatiquement dans la version la plus récente lorsque la version la plus récente est installée.
- Le numéro de version du LicenseServer actuellement installé est indiqué au bas de la page de configuration de LicenseServer (tous les onglets).

Version actuelle : 3.6

3.4.2 Obtention d'une licence sur macOS

StyleVision Server doit être mis sous licence avec un Altova LicenseServer. L'obtention de la licence est un processus en deux étapes :

1. **Enregistrer StyleVision Server** avec LicenseServer. L'enregistrement s'effectue depuis StyleVision Server.
2. **Attribuer une licence** à StyleVision Server depuis LicenseServer. Télécharger la dernière version de LicenseServer depuis le [site Internet d'Altova](#), et l'installer sur votre appareil local ou un appareil sur votre réseau.

Les étapes à suivre sont indiquées ci-dessous en bref, voir le [manuel d'utilisateur LicenseServer](#) sur le [site Internet d'Altova](#).

▼ Lancer LicenseServer

Pour enregistrer et délivrer correctement une licence pour StyleVision Server avec LicenseServer, celui-ci doit fonctionner en tant que démon. Lancer LicenseServer en tant que démon avec la commande suivante :

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

Si, à tout moment, vous souhaitez arrêter le LicenseServer, remplacer `load` avec `unload` dans la commande ci-dessus :

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

▼ Enregistrer StyleVision Server

Pour enregistrer StyleVision Server depuis l'interface à ligne de commande, utiliser la commande

```
licenseserver :  
sudo /usr/local/Altova/StyleVisionServer2021/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/StyleVisionServer2021/bin/StyleVisionServer licenseserver  
localhost
```

Dans la commande ci-dessus, `localhost` est le nom du serveur sur lequel LicenseServer est installé.

Veillez aussi noter que l'emplacement du programme d'exécution StyleVision Server est :

```
/usr/Altova/StyleVisionServer2021/bin/
```

Une fois l'enregistrement effectué avec succès, rendez-vous sur l'onglet Client Management de la page de configuration de LicenseServer pour attribuer une licence à StyleVision Server.

▼ Attribuer une licence

Une fois l'enregistrement de StyleVision Server, effectué avec succès, il sera compris dans l'onglet Client Management de la page de configuration du LicenseServer. Rendez-vous dans cette section 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 de 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 (l'appareil sur lequel le produit de serveur Altova est installé), même si la licence a une capacité de licence n'est pas utilisée. 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.

3.4.3 Notes à propos de l'environnement

Dossiers

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

☐ Installation root

`/usr/local/Altova/StyleVisionServer2021/`

☐ Fichiers de licence

`/var/Altova/StyleVisionServer`

☐ Paramètres d'environnement

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

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

Note : Ces variables d'environnement sont uniquement définies pour le processus StyleVision Server et n'ont aucun impact 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 sauf pour Microsoft Access
- Connexions natives — Actuellement disponibles pour les bases de données SQLite et PostgreSQL

Si vous utilisez JDBC, veuillez noter les points suivants :

- Java Runtime Environment ou SDK doit être installé.
- Les pilotes JDBC de la base de données cible doivent être installés.
- Les variables d'environnement suivantes doivent être configurées correctement pour votre environnement:
 - `CLASSPATH`: pour trouver les fichiers jar ; le chemin de classe est défini dans le fichier `plist`.
 - `PATH`: pour trouver le JRE, mais n'est pas forcément nécessaire, dépend de l'installation
 - `JAVA_HOME`: si nécessaire, selon l'installation

Note

Sur les serveurs macOS, les seules connexions de bases de données prises en charge sont JDBC.

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/StyleVisionServer2021/bin/stylevisionserver</code>
<i>Mac</i>	<code>/usr/local/Altova/StyleVisionServer2021/bin/stylevisionserver</code>
<i>Windows</i>	<code><ProgramFilesFolder>\Altova\StyleVisionServer2021\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](#)³⁴ : Charge une licence sur LicenseServer et attribue cette licence à StyleVision Server.
- [exportresourcestrings](#)³⁶ : Exporte tous les strings de ressource d'application vers un fichier XML.
- [generate](#)³⁸ : Génère un ou plusieurs documents depuis un fichier XML d'entrée et une feuille de style XSLT dans le fichier PXF d'entrée.

- [help](#)⁴¹ : Affiche des informations concernant la commande soumise dans l'argument (ou concernant toutes les commandes si aucun argument n'est soumis).
- [licenseserver](#)⁴² : Enregistre StyleVision Server avec un LicenseServer sur le réseau local.
- [pdfdata](#)⁴⁴ : Génère des données de formulaires depuis un fichier PDF vers un fichier FDF ou XML.
- [setdeflang](#)⁴⁶ : Définit la langue par défaut de StyleVision Server.
- [setfopath](#)⁴⁷ : Sélectionne un processeur FO alternatif pour une génération PDF ultérieure.
- [verifylicense](#)⁴⁹ : Contrôle si StyleVision Server actuel détient une licence ; en option, contrôle si une clé de licence donnée est attribuée.
- [version](#)⁵¹ : Affiche la version de StyleVision Server.

4.1 assignlicense (uniquement Windows)

Syntaxe et description

La commande `assignlicense` est disponible uniquement sur Windows (par sur les systèmes Linux ou Mac). Elle charge un fichier de licence sur l'Altova LicenseServer avec lequel StyleVision Server est enregistré (voir la commande `licenseserver`), et attribue la licence sur 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/AltovaLicenseServer/>).

▼ 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.

4.2 exportresourcestrings

Syntaxe et description

La commande `exportresourcestrings` sort un fichier XML contenant les strings de ressource de l'application de StyleVision Server dans le langage spécifié. Les langues d'exportation disponibles sont anglais (`en`), allemand (`de`), 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 du *langage d'exportation*. Les langues d'exportation autorisées sont (avec leur code respectif entre parenthèses) : anglais (`en`), allemand, (`de`), espagnol (`es`), français (`fr`) et japonais (`ja`).
- L'argument `XMLOutputFile` spécifie le chemin et le nom du fichier XML de sortie.

Vous trouverez ci-dessous une description de la localisation des strings.

▼ 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 `exportresourcestrings` :

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

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

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 (anglais, allemand, espagnol, japonais et français) sont déjà disponibles sous le dossier `C:\Program Files (x86)\Altova\StyleVisionServer2021\bin` et n'ont donc pas besoin d'être créées.

Créer une version localisée comme suit :

1. Générer un fichier XML contenant les 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 : anglais (`en`), allemand (`de`), espagnol (`es`), japonais (`ja`) et français, conformément à l'argument `LanguageCode` utilisé avec cette 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. Prendre contact avec [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 la part de [Altova Support](#), enregistrer le DLL dans le dossier `C:\Program Files (x86)\Altova\StyleVisionServer2021\bin`. Votre fichier DLL aura un nom de la forme `StyleVisionServer2021_lc.dll`. La partie `_lc` du nom contient le code de langue. Par exemple, dans `StyleVisionServer2021_de.dll`, la partie `de` est le code de langue pour l'allemand ("Deutsch").
5. Exécuter la commande `setdeflang` pour définir votre fichier localisé DLL en tant que l'application StyleVision Server à utiliser. Pour l'argument de la commande `setdeflang`, utiliser le code de langue qui fait partie du nom DLL.

Note : propose une prise en charge pour cinq langues : anglais, allemand, espagnol, japonais et français. Vous n'aurez donc pas à créer une version localisée de ces langues. Pour définir une de ces cinq langues en tant que langue par défaut, utiliser la commande `setdeflang` de StyleVision Server.

4.3 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 `StyleVisionServer2021`. 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.

▼ 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.

▼ 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.

▼ 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.

4.4 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`.

4.5 licenseserver

Syntaxe et description

La commande `licenseserver` enregistre StyleVision Server auprès du 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, voir la documentation (<https://www.altova.com/fr/manual/AltovaLicenseServer/>).

▼ 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 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.

▼ 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.

4.6 pdfdata

Syntaxe et description

La commande `pdfdata` génère un fichier FDF ou un fichier XML depuis le fichier PDF qui est soumis en tant que l'argument `InputPDF`.

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

- L'argument `InputPDF` spécifie le chemin vers le fichier PDF, à partir duquel le fichier FDF ou XML de sortie sera généré. Si le fichier PDF n'a pas de données de formulaire, le fichier généré ne contiendra aucunes données de formulaire.
- Utiliser l'option `--outfdf` pour spécifier l'emplacement du fichier PDF généré ou l'option `--outxml` pour spécifier l'emplacement du fichier XML généré.

Pour plus d'informations concernant des FDF et concevoir 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 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 `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 ci-dessus créent, respectivement, et le fichier FDF et un fichier XML depuis la même entrée PDF.

Options

▼ outfdf

`--outfdf = FilePath`

Le chemin vers le fichier FDF généré.

▼ outxml

`--outxml = FilePath`

Le chemin vers le fichier XML généré.

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.

4.7 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

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

4.8 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`³⁸ 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 `StyleVisionServer2021`.

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 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

Après avoir exécuté la commande `setfopath`, vous pouvez utiliser la commande `generate`³⁸ 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.

4.9 verifylicense (uniquement Windows)

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 a déjà été attribuée au produit. Cette commande est uniquement pris en charge sur les systèmes Windows. Elle n'est pas prise en charge sur les systèmes Linux ou Mac.

```
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/AltovaLicenseServer/>).

▼ 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

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 [I]

```
--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.

4.10 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](#) ⁵³
- [À propos de l'interface COM](#) ⁵⁴
- [À propos de l'interface Java](#) ⁵⁵
- [Exemples de code](#) ⁵⁶
- [Référence API](#) ⁶³

5.1 À propos de l'interface .NET

L'interface .NET est conçue en tant qu'un wrapper autour de l'interface COM. Elle est fournie en tant qu'un primary interop assembly signé par Altova et 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). Le `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 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 le `Altova.StyleVisionServer.dll` est disponible sur l'interface .NET, la fonction API StyleVision Server devient disponible dans votre projet .NET.

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

1. Le projet .NET ouvert dans Visual Studio, cliquer sur **Project | Add Reference**. Le dialogue Add Reference s'ouvre.
2. Dans l'onglet Browse, chercher le dossier : `<StyleVisionServer application folder>/bin`, choisir `Altova.StyleVisionServer.dll` et cliquer sur **OK**.

Vous pouvez consulter la structure de l'assembly `Altova.StyleVisionServer` avec le Visual Studio Object Browser (pour afficher l'Object Browser, cliquer 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

Pour accéder l'API StyleVision Server à partir du 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

Les deux fichiers sont disponibles dans le dossier `bin` du dossier d'installation StyleVision Server. Vous pouvez soit référencer le fichier depuis leur emplacement d'origine ou de les copier à un autre emplacement, selon la configuration de votre projet.

Ajouter des références de bibliothèque dans Eclipse

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

1. Le projet ouvert dans Eclipse, sur le menu **Project**, cliquer sur **Properties**, puis sélectionnez le chemin d'accès Java.
2. Dans l'onglet Bibliothèques, cliquer sur **Add External JARs**, puis chercher le fichier `StyleVisionServer.jar` situé dans le dossier d'installation StyleVision Server.
3. Sous *JARs and class folders on the build path*, agrandir le record `StyleVisionServer.jar`, puis double-cliquer le record `Javadoc location: (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. Cliquer sur **OK**. La référence à la bibliothèque StyleVision Server et l'archive Javadoc est ajoutée au fichier `.classpath` du projet.

5.4 Exemples de code

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

- [C++](#) ⁵⁷
- [C#](#) ⁵⁶
- [VBScript](#) ⁵⁹
- [Visual Basic](#) ⁶⁰
- [Java](#) ⁶¹

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'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](#) ⁵⁴.

```
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\\
\\StyleVisionServer2021\\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 suivant illustre comment utiliser le code C++ 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](#) ⁵⁴.

```

// 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\\
\\StyleVisionServer2021\\bin\\StyleVisionServer.exe";
        //pSVS->ServerPath = "C:\\Program Files\\Altova\\StyleVisionServer2021\\
\\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 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](#)⁵⁴.

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

```

5.4.4 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](#)⁵⁴.

```
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

```

5.4.5 Java

L'exemple suivant illustre comment utiliser le code Java 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. L'enregistrement en tant qu'objet de serveur a généralement lieu

pendant l'installation de StyleVision Server. Pour contrôler si l'enregistrement a réussi, voir [À propos de l'interface Java](#) ⁵⁵.

```

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

            //Set a working directory - used for output and for intermediate files
            objSVS.setWorkingDirectory( "C:\\temp" );

            //Default path to the StyleVision Server executable is the installation path
            (same dir with the StyleVisionServer.jar)
            //In case you copied the JAR file to a new location, you need to explicitly set
            the path to the .exe file
            //objSVS.setServerPath( "C:\\Program Files (x86)\\Altova\\
\\StyleVisionServer2021\\bin\\StyleVisionServer.exe" );

            //Prepare the name of the working XML
            objSVS.setInputXML( "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( "C:\\Program Files (x86)\\Altova\\StyleVisionServer2021\\
etc\\Examples\\ExpReport.pxf" ) )
                System.out.println( "Success - finished execution" );
            else
                System.out.println( objSVS.getLastExecutionMessage() );
        }
        catch ( Exception e )
        {
            e.printStackTrace();
        }
    }
}

```

5.5 Référence API

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

- [COM et .NET](#) ⁶³
- [Java](#) ⁷⁰

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
```

▼ 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.

▼ `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 taxonomie XBRL

Gestionnaire de taxonomie XBRL 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 XBRL a une interface utilisateur graphique et est aussi disponible dans la ligne de commande. Sur Linux et Mac*, l'outil est disponible uniquement dans la ligne de commande.

* Les systèmes d'exploitation Linux et macOS sont uniquement applicables si vous exécutez Gestionnaire de taxonomie XBRL sur les systèmes d'exploitation en conjonction avec les application de serveur multi-plateforme Altova comme MapForce Server, StyleVision Server, ou RaptorXML+XBRL Server.

Gestionnaire de taxonomie XBRL 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. Toutes les taxonomies sont entretenues par Altova sur un stockage en ligne accessible à Gestionnaire de taxonomie XBRL, et vous pouvez les télécharger dès qu'elles sont disponibles.
- Installer ou désinstaller une des versions d'une taxonomie donnée (ou toutes les versions si nécessaire).
- Une seule taxonomie XBRL représente un "package" mais elle peut avoir des dépendances sur d'autres taxonomies. Que vous choisissiez d'installer ou de désinstaller une taxonomie particulière, toute taxonomie dépendante est détectée et est installée ou supprimée automatiquement. L'interface d'utilisateur graphique (ou la ligne de commande le cas échéant) vous informe lorsque des dépendances sont ajoutées ou supprimées.
- Les taxonomies XBRL entretenues par Gestionnaire de taxonomie XBRL bénéficient du mécanisme [XML catalog](#) qui permet la résolution des références URI dans des documents d'instance ou de schéma des références depuis des fichiers locaux, contrairement à une extraction d'Internet. Cela est extrêmement important dans le cas des grandes taxonomies XBRL où la résolution de schéma depuis les URIs à distance n'est pas pratique ou même recommandé, surtout pour des raisons de performance.

Gestionnaire de taxonomie XBRL offre un moyen d'administrer une des taxonomies XBRL nécessaire pour une utilisation dans un des [applications activées pour XBRL Altova](#)⁷⁵. Ceux-ci contiennent les taxonomies European Banking Authority Reporting Framework, les taxonomies US-GAAP Financial Reporting et diverses autres taxonomies XBRL spécifiques au pays ou au domaine. Pour consulter la liste complète, exécutez soit Gestionnaire de taxonomie XBRL soit exécuter la commande `list` dans la ligne de commande .

Taxonomies XBRL personnalisées

Si vous souhaitez travailler avec des taxonomies XBRL personnalisées qui ne sont pas incluses avec Gestionnaire de taxonomie XBRL, vous pouvez les activer comme suit :

- En partant de votre application de desktop activée pour XBRL Altova, exécuter la commande de menu **Outils | Options**, puis rendez-vous sur la page de paramètres **XBRL | Packages de taxonomie**. à partir de la page des paramètres, cherchez la page des paramètres, chercher le package .zip de votre taxonomie XBRL parsonnalisée.
- Depuis votre application de serveur activée pour XBRL Altova, fournir les options `--taxonomy-package` ou `--taxonomy-package-config-file` lorsque vous exécutez des commandes ou des méthodes API qui les prennent en charge. Par exemple, ces options sont prises en charge par des

commandes de validation XBRL comme `valxbrl` ou `valxbrltaxonomy` dans RaptorXML+XBRL Server, ou par la commande `run` (ou la méthode API équivalente) dans MapForce Server.

Applications activées pour XBRL Altova

Les applications Altova suivantes sont activées pour XBRL et bénéficient donc des fonctions fournies par Gestionnaire de taxonomie XBRL:

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

Les modifications dans les taxonomies XBRL en utilisant Gestionnaire de taxonomie XBRL prennent effet pour toutes les applications recensées ci-dessus si elles sont installées sur le même ordinateur.

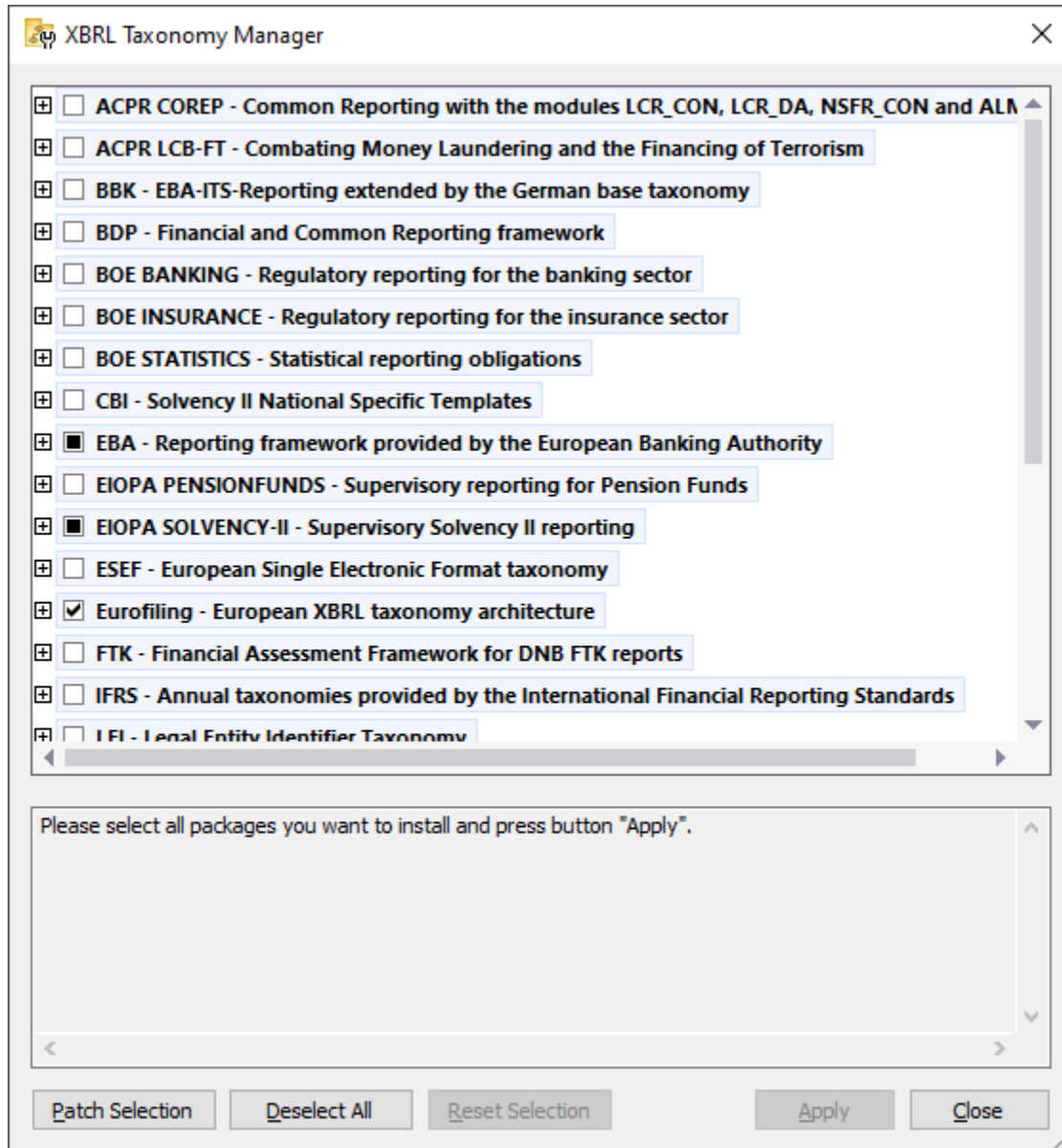
Installation

Gestionnaire de taxonomie XBRL est installé automatiquement dès que vous installez une application activée pour XBRL Altova ou l'**Altova Mission Kit Enterprise Edition**. De même, il est supprimé automatiquement lorsque vous désinstallez la dernière application activée pour XBRL Altova depuis votre ordinateur ou l'**Altova Mission Kit Enterprise Edition**.

Fonctionnement

Altova entretient un stockage de taxonomie en ligne où toutes les taxonomies XBRL utilisées dans les produits Altova sont stockés. Ce stockage de taxonomie est mis à jour sur une base périodique, par exemple, juste après que les taxonomies pertinentes sont publiées par leurs organisations émettrices.

Lorsque vous exécutez Gestionnaire de taxonomie XBRL dans l'interface d'utilisateur graphique, des informations concernant les dernières taxonomies disponibles sont affichées dans un dialogue, où vous pouvez consulter, installer, mettre à niveau ou les désinstaller. Vous pouvez aussi effectuer les mêmes actions dans l'interface de ligne de commande.



Vous pouvez aussi installer des taxonomies en exécutant des fichiers `.altova_taxonomies` téléchargés depuis le site web Altova (<https://www.altova.com/taxonomy-manager>). De plus, lorsque vous entreprenez une action qui exige le chargement des taxonomies XBRL, StyleVision Server vous invite à installer les taxonomies XBRL manquantes.

Quelque soit le moyen par lequel les taxonomies ont été installées, toutes les informations concernant des taxonomies installées sont conservées dans un emplacement centralisé sur votre ordinateur, aussi connu en tant que le répertoire de cache local. Celui-ci contient des informations concernant des package Altova (sauf pour les fichiers de taxonomie, qui sont installés sur demande). Le répertoire de cache local se trouve dans le chemin suivant :

<i>Linux</i>	<code>/var/opt/Altova/pkggs</code>
--------------	------------------------------------

<i>macOS</i>	/var/Altova/pkg
<i>Windows</i>	C:\ProgramData\Altova\pkg

Le répertoire de cache local est mis à jour automatiquement de temps en temps, afin de distribuer le dernier état du stockage en ligne dans l'ordinateur local. Plus spécifiquement, le cache est mis à jour comme suit :

- Lorsque vous exécutez le Gestionnaire de Taxonomie XBRL.
- Lorsque vous exécutez StyleVision Server pour la première fois dans le même jour civil.
- Si StyleVision Server est déjà en cours d'exécution, le répertoire de cache est mis à jour toutes les 24 heures.
- Vous pouvez aussi mettre à jour le cache local depuis le stockage en ligne sur demande, en exécutant la commande de mise à jour dans l'interface de ligne de commande.

Lorsque vous installez ou désinstallez des taxonomies, le répertoire de cache local est mis à jour automatiquement avec des informations concernant les taxonomies disponibles et installées, ainsi que les fichiers de taxonomie eux-même.

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. Si vous souhaitez réinitialiser Gestionnaire de taxonomie XBRL à son état "pristine" (pur) d'origine, exécuter la commande `reset` de l'interface de ligne de commande, puis exécuter la commande `initialize`. (En alternative, exécuter la commande `reset` avec l'option `-i`.)

6.1 Exécuter Gestionnaire de taxonomie XBRL

Vous pouvez exécuter Gestionnaire de taxonomie XBRL en utilisant les approches suivantes.

Interface graphique de l'utilisateur (uniquement Windows)

Si StyleVision Server est exécuté sur Windows, vous pouvez exécuter Gestionnaire de taxonomie XBRL comme suit :

- Exécuter (double-cliquer) un fichier avec l'extension `.altova_taxonomies` téléchargé depuis le site web de Altova.
- À partir du Panneau de commandes Windows, cliquez avec la touche de droite sur l'entrée **Altova Taxonomie Manager** et sélectionnez **Modifier** ou **Désinstaller** depuis le menu contextuel.

De plus, la case à cocher **Invoke Altova Taxonomy Manager** est disponible sur la dernière page de l'assistant de l'installation, une fois après avoir terminé l'installation de StyleVision Server.

Interface de ligne de commande (Windows)

Pour exécuter Gestionnaire de taxonomie XBRL depuis une interface de la ligne de commande :

1. Ouvrir une invite de commande et passer au répertoire **C:\ProgramData\Altova\SharedBetweenVersions**.
2. Pour afficher l'aide dans la ligne de commande, exécuter :

```
taxonomymanager.exe --help
```

Interface de ligne de commande (Linux, macOS)

Pour exécuter Gestionnaire de taxonomie XBRL depuis une interface de la ligne de commande :

1. Ouvrir une fenêtre de terminal et modifier le répertoire en `%INSTALLDIR%/bin`. `%INSTALLDIR%` est le répertoire d'installation du programme.
2. Pour afficher l'aide dans la ligne de commande, exécuter :

```
sudo ./taxonomymanager --help
```

6.2 Installer une taxonomie

Vous pouvez installer une taxonomie en utilisant les approches suivantes.

Veillez noter les points suivants :

- L'installation ou la désinstallation d'une taxonomie provenant de Gestionnaire de taxonomie XBRL prend effet pour tous les comptes d'utilisateurs sur le même ordinateur.
- L'installation ou la désinstallation d'une taxonomie provenant de Gestionnaire de taxonomie XBRL prend effet pour toutes les applications liées à Altova XBRL installées sur le même ordinateur.
- Si la taxonomie actuelle a des dépendances avec d'autres taxonomies, les taxonomies dépendantes sont aussi installées (ou désinstallées, selon).

Interface graphique de l'utilisateur (seulement Windows)

Pour installer une taxonomie :

1. Run Gestionnaire de taxonomie XBRL.
2. Sélectionnez la case située à côté des taxonomies ou versions de taxonomie que vous souhaitez installer et cliquer sur **Appliquer**.

En alternative, si vous avez téléchargé un fichier avec l'extension **.altova_taxonomies** depuis le site web de Altova (<https://www.altova.com/taxonomy-manager>), double-cliquez sur le fichier **.altova_taxonomies** pour l'exécuter. Gestionnaire de taxonomie XBRL s'ouvre lorsque vous exécutez le fichier **.altova_taxonomies**.

Interface de ligne de commande (Windows)

Pour installer une taxonomie, exécuter :

```
TaxonomyManager.exe install FILTER ...
```

Où **FILTER** signifie une des deux choses suivantes :

1. Un identificateur de taxonomie de format **<name>-<version>**, par exemple : **eba-2.10, us-gaap-2020.0**. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande `list`.
2. Un fichier **.altova_taxonomies** téléchargé depuis le site internet d'Altova.

Interface de ligne de commande (Linux, macOS)

Pour installer une taxonomie, exécuter :

```
sudo ./taxonomymanager install FILTER ...
```

Où **FILTER** signifie une des deux choses suivantes :

1. Un identificateur de taxonomie de format **<name>-<version>**, par exemple : **eba-2.10, us-gaap-2020.0**. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande `list`.
2. Un fichier **.altova_taxonomies** téléchargé depuis le site internet d'Altova.

Installer des taxonomies sur demande


Lorsque StyleVision Server détecte que certaines taxonomies XBRL doivent être installées sur votre ordinateur, vous serez invité à installer des taxonomies sur demande. Par exemple, si vous exécutez une commande qui exige le chargement de taxonomies XBRL, et si aucune taxonomie n'est encore installée, la commande sera annulée et un message d'erreur semblable aux suivants apparaît :

```
Unable to load a schema with target namespace [...] from [...].
Details:
I/O Error: File [...] is part of missing taxonomy [...]. Call 'taxonomymanager install
[...]'
```



Pour installer la taxonomie manquante, exécuter la commande `install` comme suggéré par le message d'erreur. Vous pouvez toujours consulter toutes les taxonomies installées précédemment en exécutant la commande `list`.

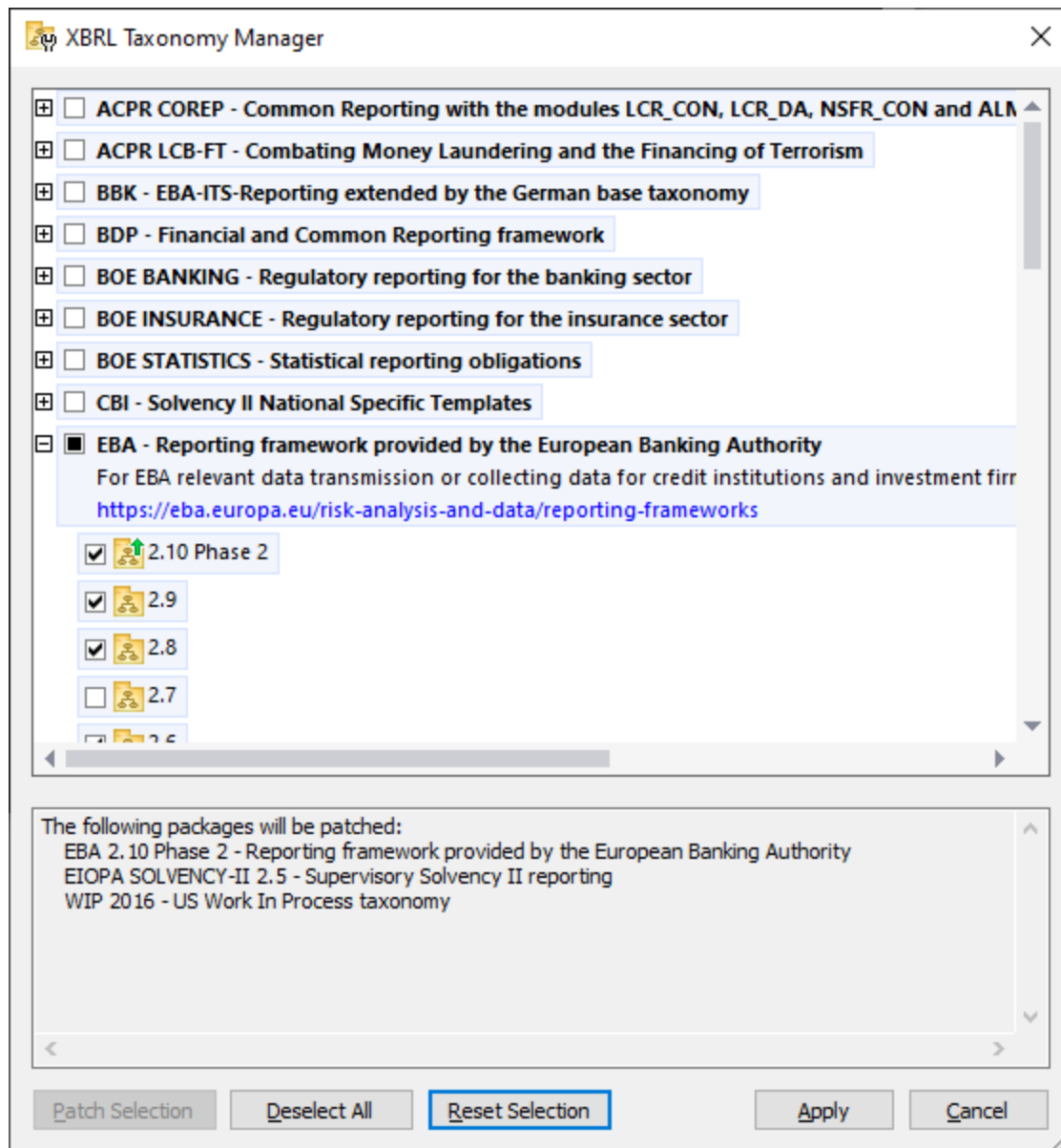
Appliquer des patches

Parfois, des taxonomies XBRL peuvent recevoir des patches depuis leurs émetteurs. Lorsque le Gestionnaire de Taxonomie XBRL détecte les patches disponibles, les choses suivantes se produisent :

- Si vous utilisez le Gestionnaire de Taxonomie XBRL par le biais de l'interface d'utilisateur graphique Windows, les taxonomies XBRL respectives sont affichées avec l'icône .
- Si vous utilisez la ligne de commande ou un système Linux/macOS, toutes les taxonomies XBRL qui ont des patches sont listées lorsque vous exécutez la commande `list -u`.

Pour appliquer un patch sur Windows :

1. Cliquer sur la touche **Sélection de Patch**. L'icône de chaque taxonomie XBRL qui qualifie des modifications de  à , et le dialogue vous informe des patches qui doivent être appliqués, par exemple :



Note: La touche **Sélection de Patch** est activée uniquement lorsque des patches sont disponibles pour une des taxonomies XBRL installées actuellement.

2. Cliquer sur **Appliquer**.

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

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

Pour plus d'informations, voir la référence à l'Interface de Ligne de Commande .

6.3 Consulter les taxonomies installées

Vous pouvez consulter les taxonomies installées actuellement en utilisant les approches ci-dessous.

Interface graphique de l'utilisateur (seulement Windows)

Pour consulter toutes les taxonomies installées depuis une interface utilisateur graphique, exécuter le Gestionnaire de taxonomie XBRL. Une case cochée à côté d'une taxonomie (ou une version de taxonomie) indique que cette taxonomie est installée.

Interface de ligne de commande (Windows)

Pour consulter toutes les taxonomies disponibles depuis une interface de ligne de commande, exécuter :

```
TaxonomyManager.exe list
```

Pour consulter uniquement les taxonomies installées, exécuter :

```
TaxonomyManager.exe list -i
```

Pour consulter uniquement les taxonomies lorsqu'une version plus récente est disponible, exécuter :

```
TaxonomyManager.exe list -u
```

Interface de ligne de commande (Linux, macOS)

Pour consulter tous les taxonomies disponibles, exécuter :

```
sudo ./taxonomymanager list
```

Pour consulter uniquement les taxonomies installées, exécuter :

```
sudo ./taxonomymanager list -i
```

Pour consulter uniquement les taxonomies lorsqu'une version plus récente est disponible, exécuter :

```
sudo ./taxonomymanager list -u
```

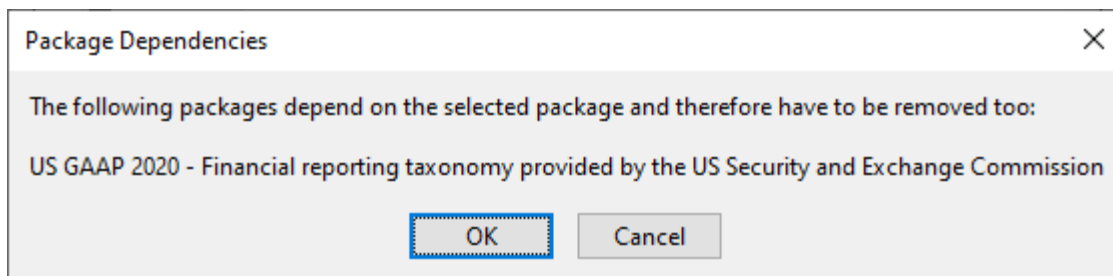
6.4 Désinstaller une taxonomie

Vous pouvez désinstaller une taxonomie en utilisant les approches suivantes.

Interface graphique de l'utilisateur (seulement Windows)

Pour désinstaller une taxonomie :

1. Run Gestionnaire de taxonomie XBRL.
2. Effacez la case à cocher à côté des taxonomies ou versions de taxonomie que vous voulez désinstaller. Si la taxonomie sélectionnée dépend d'autres packs de taxonomie, une boîte de dialogue s'ouvre, vous informant que les dépendances seront également effacées, par exemple :



3. Cliquez sur **Appliquer** pour retirer la taxonomie et ses dépendances.

Interface de ligne de commande (Windows)

Pour désinstaller une taxonomie, exécuter :

```
TaxonomyManager.exe uninstall FILTER...
```

Où `FILTER` signifie une des deux choses suivantes :

1. Un identificateur de taxonomie de format `<name>-<version>`, par exemple : **eba-2.10, us-gaap-2020.0**. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande `list`.
2. Un fichier **.altova_taxonomies** téléchargé depuis le site internet d'Altova.

Interface de ligne de commande (Linux, macOS)

Pour désinstaller une taxonomie, exécuter :

```
sudo ./taxonomymanager uninstall FILTER...
```

Où `FILTER` signifie une des deux choses suivantes :

1. Un identificateur de taxonomie de format `<name>-<version>`, par exemple : **eba-2.10, us-gaap-2020.0**. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande `list`.
2. Un fichier **.altova_taxonomies** téléchargé depuis le site internet d'Altova.

6.5 Interface de ligne de commande

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

<i>Linux*</i>	/opt/Altova/StyleVisionServer2021/bin/taxonomymanager
<i>macOS*</i>	/usr/local/Altova/StyleVisionServer2021/bin/taxonomymanager
<i>Windows</i>	C:\ProgramData\Altova\SharedBetweenVersions\TaxonomyManager.exe

* Les chemins Linux et macOS sont uniquement applicables si vous exécutez Gestionnaire de taxonomie XBRL sur ces systèmes d'exécution en conjonction avec les applications de serveur multi-plateforme Altova comme MapForce Server, StyleVision Server ou RaptorXML+XBRL Server.

Par convention, cette documentation omet le chemin complet de l'exécutable lors de la description d'une commande donnée, et utilise **<exec>** à la place du nom de l'exécutable, par exemple :

```
<exec> help
```

Où **<exec>** est le chemin ou le nom de l'exécutable.

Appeler Gestionnaire de taxonomie XBRL depuis le répertoire d'installation

Pour appeler l'exécutable sans devoir saisir le chemin complet, modifier le répertoire suivant dans celui en-dessous :

<i>Linux</i>	cd /opt/Altova/StyleVisionServer2021/bin
<i>macOS</i>	cd /usr/local/Altova/StyleVisionServer2021/bin
<i>Windows</i>	cd "C:\ProgramData\Altova\SharedBetweenVersions"

Vous pouvez exécuter une commande en appelant l'exécutable avec un chemin relatif, par exemple :

<i>Linux</i>	sudo ./taxonomymanager help
<i>macOS</i>	sudo ./taxonomymanager help
<i>Windows</i>	TaxonomyManager.exe help

Note : dans les systèmes Linux et macOS, le préfixe `./` indique que l'exécutable se trouve dans le répertoire actuel. Le préfixe `sudo` indique que la commande que la commande doit être exécutée avec des privilèges root.

Appeler Gestionnaire de taxonomie XBRL depuis n'importe quel répertoire

Pour appeler l'exécutable depuis n'importe quel répertoire, vous devez vous y référer en utilisant le chemin

absolu. En alternative, si vous souhaitez appeler le programme en saisissant uniquement le nom de l'exécutable, vous pouvez éditer la variable d'environnement PATH de votre système d'exploitation de manière à ce qu'il inclut le chemin complet vers le répertoire de l'exécutable. Pour découvrir comment modifier la variable d'environnement PATH, référez-vous à la documentation de votre système d'exploitation.

Notes:

- Une fois avoir modifié la variable d'environnement PATH, vous devrez éventuellement fermer la fenêtre de terminal et en ouvrir une nouvelle, pour que les modifications puissent prendre effet
- Sur Linux et macOS, l'utilisation de `sudo` ne prend pas en compte le PATH de l'utilisateur.

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 taxonomie XBRL.

Syntaxe

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

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

Remarques

Vous pouvez aussi invoquer de l'aide en saisissant une commande suivie par `-h` ou `--help`, par exemple :

```
<exec> list --h
```

Vous pouvez aussi invoquer une aide générale en saisissant `-h` ou `--help` directement après l'exécutable, par exemple :

```
<exec> --help
```

Exemple

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

<i>Linux, macOS</i>	<code>./taxonomymanager help list</code>
<i>Windows</i>	<code>TaxonomyManager.exe help list</code>

6.5.2 info

Cette commande affiche des informations détaillées pour chacune des taxonomies fournies en tant qu'argument. L'information détaillée contient le titre, la version, la description, l'éditeur et les références de dépendance.

Syntaxe

```
<exec> info [options] FILTER...
```

Pour spécifier plusieurs taxonomies, renouveler `FILTER` autant de fois que nécessaire.

Arguments

<code>FILTER</code>	Recenser uniquement les taxonomies qui contiennent ce string dans leur nom (par exemple, eba ou us-gaap-2020.0). Vous pouvez spécifier cet argument plusieurs fois. Pour consulter la liste complète des identifiants, exécuter la commande <code>list</code> .
---------------------	---

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
--------------------------	---

Exemple

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

<i>Linux, macOS</i>	<code>./taxonomymanager info eba-2.10 us-gaap-2020.0</code>
<i>Windows</i>	<code>TaxonomyManager.exe info eba-2.10 us-gaap-2020.0</code>

6.5.3 initialize

Cette commande initialise l'environnement Gestionnaire de taxonomie XBRL. Elle crée un répertoire de cache où les informations concernant toutes les taxonomies sont stockées localement. L'initialisation est effectuée automatiquement la première fois lorsque vous installez une application Altova qui inclut des prises en charge pour le Gestionnaire de taxonomie XBRL, il n'est donc pas nécessaire d'exécuter cette commande dans des circonstances habituelles. Généralement, vous devez exécuter cette commande après avoir exécuté la commande `reset`.

Syntaxe

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

L'alias de cette commande est `init`.

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--verbose, --v</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .

6.5.4 install

Cette commande installe un ou plusieurs packages.
Veuillez noter les points suivants :

- L'installation ou la désinstallation d'une taxonomie provenant de Gestionnaire de taxonomie XBRL prend effet pour tous les comptes d'utilisateurs sur le même ordinateur.
- L'installation ou la désinstallation d'une taxonomie provenant de Gestionnaire de taxonomie XBRL prend effet pour toutes les applications liées à Altova XBRL installées sur le même ordinateur.
- Si la taxonomie actuelle a des dépendances avec d'autres taxonomies, les taxonomies dépendantes sont aussi installées (ou désinstallées, selon).

Syntaxe

```
<exec> install [options] FILTER...
```

Pour spécifier plusieurs taxonomies à installer, renouveler `FILTER` autant de fois que nécessaire.

Arguments

FILTER	<p>Où <code>FILTER</code> signifie une des deux choses suivantes :</p> <ol style="list-style-type: none"> 1. Un identificateur de taxonomie de format <code><name>-<version></code>, par exemple : eba-2.10, us-gaap-2020.0. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande <code>list</code>. 2. Un fichier .altova_taxonomies téléchargé depuis le site internet d'Altova. <p>Vous pouvez aussi utiliser des identifiants abrégés s'ils sont uniques, par exemple eba. Si vous utiliser un identifiant abrégé, cela installera la dernière version disponible de cette taxonomie.</p>
--------	--

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--verbose, --v</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .

Exemple

La commande suivante installe les dernières taxonomies **eba** (European Banking Authority) et **us-gaap** (US Generally Accepted Accounting Principles) **eba-20.12.0** et **us-gaap-20.04.0** :

<i>Linux, macOS</i>	<code>./taxonomymanager install eba us-gaap</code>
<i>Windows</i>	<code>TaxonomyManager.exe install eba us-gaap</code>

6.5.5 list

Utiliser cette commande pour recenser les taxonomies dans la ligne de commande, d'une des manières suivantes :

- liste de tous les taxonomies disponibles
- liste des taxonomies spécifiques
- uniquement la liste des taxonomies installés
- uniquement la liste des taxonomies qui nécessitent une mise à niveau.

Syntaxe

```
<exec> list [options] [FILTER...]
```

Arguments

<code>FILTER</code>	Recenser uniquement les taxonomies qui contiennent ce string dans leur nom. Vous pouvez spécifier cet argument plusieurs fois
---------------------	---

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--installed, --i</code>	uniquement la liste des taxonomies installées La valeur par défaut est false .
<code>--upgradeable, --u</code>	Recenser uniquement les taxonomies lorsque des mises à niveau pour des versions patch plus récentes sont disponibles. La valeur par défaut est false .

Exemples

Pour recenser tous les taxonomies disponibles, exécuter :

<i>Linux, macOS</i>	<code>./taxonomymanager list</code>
<i>Windows</i>	<code>TaxonomyManager.exe list</code>

Pour recenser uniquement les taxonomies installées, exécuter :

<i>Linux, macOS</i>	<code>./taxonomymanager list -i</code>
<i>Windows</i>	<code>TaxonomyManager.exe list -i</code>

Pour recenser tous les taxonomies qui contiennent soit "eba" soit "us-gaap" dans leur nom, exécuter :

<i>Linux, macOS</i>	<code>./taxonomymanager list eba us-gaap</code>
<i>Windows</i>	<code>TaxonomyManager.exe list eba us-gaap</code>

6.5.6 reset

Cette commande supprime toutes les taxonomies installées et le répertoire de mise sous cache.

Attention : Cette commande supprime toutes les taxonomies installées et leur information.

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

Noter que `reset -i` restaure l'installation d'origine du produit, il est donc recommandé d'exécuter aussi la commande `update` également, après avoir effectué une réinitialisation. En alternative, exécuter la commande `reset` avec les options `-i` et `-u`

Syntaxe

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

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--i, --init</code>	Initialiser l'environnement du Gestionnaire de taxonomie XBRL après la réinitialisation. Les valeurs valides sont true et false . La valeur par défaut est false .

<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--update, --u</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .
<code>--verbose, --v</code>	Afficher des informations supplémentaires pendant l'exécution. La valeur par défaut est true .

Exemples

Pour réinitialiser le Gestionnaire de taxonomie XBRL, exécuter :

<i>Linux, macOS</i>	<code>./taxonomymanager reset</code>
<i>Windows</i>	<code>TaxonomyManager.exe reset</code>

6.5.7 setdeflang

Cette commande définit le langage de Gestionnaire de taxonomie XBRL.

Syntaxe

```
<exec> setdeflang language
```

Où `language` est un argument obligatoire fournissant le code de langage. L'alias de cette commande est `sdl`.

Arguments

<code>language</code>	Le langage à définir. Les valeurs valides sont l'anglais (<code>en</code>), le français (<code>fr</code>), l'allemand (<code>de</code>), le japonais (<code>ja</code>) et l'espagnol (<code>es</code>).
-----------------------	---

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
--------------------------	---

Exemples

Pour définir le langage en espagnol, exécuter :

<i>Linux, macOS</i>	<code>./taxonomymanager setdeflang es</code>
<i>Windows</i>	<code>TaxonomyManager.exe setdeflang es</code>

6.5.8 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> uninstall FILTER (...)
```

Pour spécifier plusieurs packages, renouveler `FILTER` autant de fois que nécessaire.

Arguments

FILTER	<p>Où <code>FILTER</code> signifie une des deux choses suivantes :</p> <ol style="list-style-type: none"> 1. Un identificateur de taxonomie de format <code><name>--<version></code>, par exemple : eba-2.10, us-gaap-2020.0. Pour consulter tous les identifiants et versions de taxonomie disponibles, exécuter la commande <code>list</code>. 2. Un fichier .altova_taxonomies téléchargé depuis le site internet d'Altova.
--------	---

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--keep-references, --k</code>	Si cette option est configurée, les taxonomies référencées ne sont pas désinstallées. La valeur par défaut est true .
<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--update, --u</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .

Exemple

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

<i>Linux, macOS</i>	<code>./taxonomymanager uninstall eba-2.10 us-gaap-2020.0</code>
<i>Windows</i>	<code>TaxonomyManager.exe uninstall eba-2.10 us-gaap-2020.0</code>

6.5.9 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. La mise à jour de cette information a lieu implicitement et vous ne devez pas exécuter cette commande à moins d'avoir effectué un `reset` et `intialize`.

Syntaxe

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

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--update, --u</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .

Exemple

La commande suivante met à jour l'information des taxonomies.

<i>Linux, macOS</i>	<code>./taxonomymanager update</code>
<i>Windows</i>	<code>TaxonomyManager.exe update</code>

6.5.10 upgrade

Cette commande met à niveau toutes les taxonomies éligibles à la dernière version *patch* disponible. Autrement dit, elle n'effectue que les mises à niveau au niveau patch d'une release spécifique. L'exécution de cette commande est uniquement utile si des taxonomies susceptibles d'être mises à niveau sont disponibles. Vous pouvez les identifier en exécutant la commande `list -u`.

Syntaxe

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

Options

<code>--help, --h</code>	Affiche l'aide concernant cette commande dans la ligne de commande.
<code>--silent, --s</code>	Affiche uniquement des messages d'erreur. La valeur par défaut est false .
<code>--verbose, --v</code>	Affiche plus d'informations pendant l'exécution. La valeur par défaut est false .

Index

C

Cœurs, 12

Cœurs de processeur, 12

Cœurs de processeur et licences, 12

Configuration,

sur Linux, 20

sur macOS, 27

sur Windows, 13

E

Exécution traitement unique, 12

F

FlowForce Server,

et StyleVision Server, 9

G

Gestionnaire de taxonomie XBRL, 74

I

Installation sur Linux, 20

Installation sur macOS, 27

Installation sur Windows, 13

L

Licence sur Linux, 23

Licence sur Mac OS X, 29

Licence sur Windows, 16

Linux,

installation sur, 20

licence sur, 23

M

Mac OS X,

licence sur, 29

macOS,

installation sur, 27

S

StyleVision Server,

dans le flux de travail FlowForce, 9

fonctionnement, 8

fonctions, 8

W

Windows,

installation sur, 13

licence sur, 16