

Altova StyleVision Server 2021

Manual del usuario y referencia

Altova StyleVision Server 2021 Manual del usuario y referencia

Todos los derechos reservados. Ningún fragmento de esta publicación podrá ser reproducido de manera alguna (ya sea de forma gráfica, electrónica o mecánica, fotocopiado, grabado o reproducido en sistemas de almacenamiento y recuperación de información) sin el consentimiento expreso por escrito de su autor/editor.

Los productos a los que se hace referencia en este documento pueden ser marcas registradas de sus respectivos propietarios. El autor y editor no afirman ser propietarios de dichas marcas registradas.

Durante la elaboración de este documento se tomaron todas las precauciones necesarias para prevenir errores. Sin embargo, el autor y editor no se responsabilizan de los errores u omisiones que pudiese contener el documento ni de los posibles daños o perjuicios derivados del uso del contenido de este documento o de los programas y código fuente que vengan con el documento. Bajo ninguna circunstancia se podrá considerar al autor y editor responsables de la pérdida de beneficios ni de cualquier otro daño y perjuicio derivado directa o indirectamente del uso de este documento.

Fecha de publicación: 2015-2021

© 2015-2021 Altova GmbH

Contenido

1	Introducción	6
2	Funcionamiento	8
2.1	Como módulo de FlowForce Server.....	9
2.2	Como producto servidor independiente.....	10
3	Configuración de StyleVision Server	11
3.1	Núcleos de procesador y licencias.....	12
3.2	Instalación y configuración en Windows.....	13
3.2.1	Instalación en Windows.....	13
3.2.2	Asignación de licencias en Windows.....	16
3.3	Instalación y configuración en Linux.....	20
3.3.1	Instalación en Linux.....	20
3.3.2	Asignación de licencias en Linux.....	23
3.3.3	Notas sobre configuración del entorno.....	25
3.4	Instalación y configuración en macOS.....	27
3.4.1	Instalación en macOS.....	27
3.4.2	Asignación de licencias en macOS.....	29
3.4.3	Notas sobre configuración del entorno.....	31
3.5	Notas sobre la configuración.....	33
4	Línea de comandos de StyleVision Server	34
4.1	assignlicense (Windows only).....	36
4.2	exportresourcestrings.....	38
4.3	generate.....	40
4.4	help	43
4.5	licenseserver.....	44

4.6	pdfdata.....	46
4.7	setdeflang.....	48
4.8	setfopath.....	49
4.9	verifylicense (Windows only).....	51
4.10	version.....	53

5 API de StyleVision Server 54

5.1	Información sobre la interfaz .NET.....	55
5.2	Información sobre la interfaz COM.....	56
5.3	Interfaz sobre la interfaz Java.....	57
5.4	Ejemplos de código.....	58
5.4.1	C++	58
5.4.2	C#	60
5.4.3	VBScript.....	61
5.4.4	Visual Basic.....	63
5.4.5	Java	64
5.5	Referencia de la API.....	67
5.5.1	COM y .NET	67
5.5.2	Java	74

6 Gestor de taxonomías XBRL 78

6.1	Ejecutar Gestor de taxonomías XBRL.....	82
6.2	Instalar una taxonomía.....	83
6.3	Ver las taxonomías instaladas.....	87
6.4	Desinstalar taxonomías.....	88
6.5	Interfaz de la línea de comandos.....	89
6.5.1	help	90
6.5.2	info	91
6.5.3	initialize.....	91
6.5.4	install	92
6.5.5	list	93
6.5.6	reset	94
6.5.7	setdeflang.....	95

6.5.8	uninstall.....	96
6.5.9	update	96
6.5.10	upgrade.....	97

Índice	98
---------------	-----------

1 Introducción

Altova StyleVision Server es una implementación del motor de ejecución integrado de [StyleVision](#). Funciona como módulo de [FlowForce Server](#)⁹ y también está disponible como [producto servidor independiente](#)¹⁰.

StyleVision Server ejecuta paquetes de transformación implementados en [FlowForce Server](#)⁹. FlowForce Server inicia estas transformaciones en base a desencadenadores temporizados programables, desencadenadores de archivos o desencadenadores remotos. Además las funciones de StyleVision Server se pueden invocar [desde la línea de comandos](#)¹⁰.

Requisitos del sistema, instalación y asignación de licencias

StyleVision Server es compatible con estos sistemas operativos:

▼ Windows

Windows 7 SP1 con actualización de la plataforma, Windows 8, Windows 10

▼ Windows Server

Windows Server 2008 R2 SP1 con actualización de la plataforma o superior

▼ Linux

- CentOS 7 o superior
- RedHat 7 o superior
- Debian 8 o superior
- Ubuntu 16.04 LTS o superior

Las bibliotecas que aparecen a continuación son un requisito previo para la instalación y ejecución de la aplicación. Si los paquetes que aparecen en esta tabla no están en su equipo Linux, ejecute el comando `yum` (o `apt-get` si procede) para instalarlos.

Requisito para	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

Nota: si quiere usar la función de gráficos de Altova, entonces debe haber al menos una fuente instalada en su sistema para que los gráficos se muestren correctamente. Para ver una lista de las fuentes instaladas use, por ejemplo, el comando `fc-list` de la [Fontconfig library](#).

▶ macOS

macOS 10.13 o superior

Además Altova ofrece versiones de 32 y 64 bits de StyleVision Server. Para más información sobre cómo instalar StyleVision Server y asignarle licencias, consulte los apartados sobre la instalación en [Windows](#)¹³, [Linux](#)²⁰ y [macOS](#)²⁷.

Nota: Si en uno de estos archivos PDF faltan partes editables al abrirlo en un sistema Mac OS, esto puede deberse a que Java 6 no esté instalado en el equipo. Si es así, puede instalar Java 6 desde https://support.apple.com/kb/dl1572?locale=en_US. Si ya tiene instalada una versión más reciente que Java 6, entonces la instalación de Java 6 no afectará al funcionamiento de la nueva versión, que será la versión predeterminada del sistema.

Última actualización: 2/26/2021

2 Funcionamiento

StyleVision Server transforma archivos XML en documentos de salida HTML, PDF, RTF y DOCX con ayuda de hojas de estilos XSLT. Estas hojas de estilos XSLT se obtienen a partir de archivos PXF creados con la aplicación de diseño de hojas de estilos [Altova StyleVision](#).

StyleVision Server puede utilizarse de dos maneras:

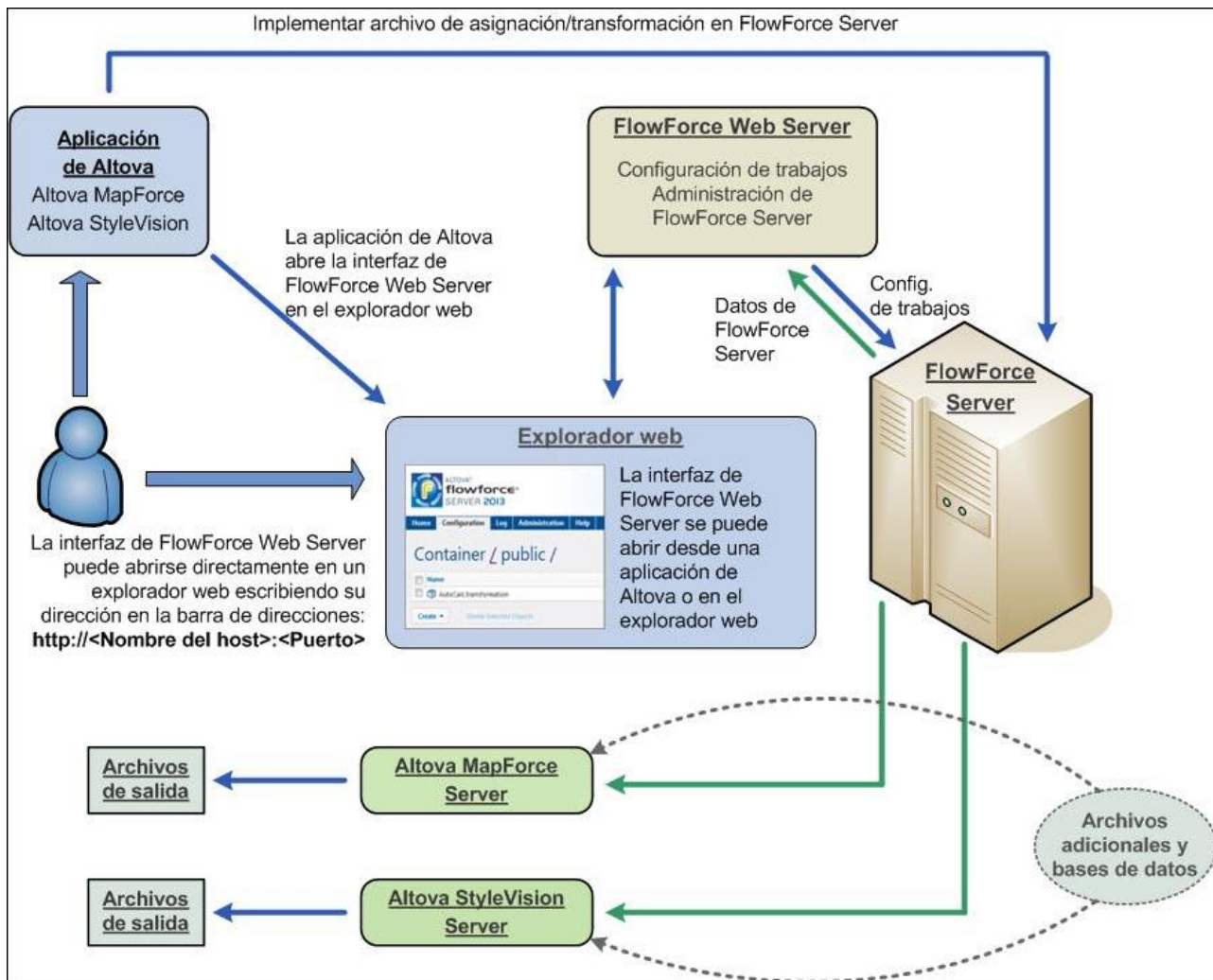
- Como módulo de [Altova FlowForce Server](#)⁹. Visite el [sitio web de Altova](#) para obtener más información sobre Altova FlowForce Server.
- [Como producto servidor independiente](#)¹⁰ al que puede acceder desde su interfaz de la línea de comandos (ILC).

El usuario pasa un archivo XML de entrada y un archivo PXF a StyleVision Server, que genera los documentos de salida necesarios.

2.1 Como módulo de FlowForce Server

En [Altova FlowForce Server](#) se crean trabajos de FlowForce. Un trabajo de FlowForce especifica: (i) las entradas y salidas de una transformación de StyleVision Server y (ii) los desencadenadores que inician la ejecución del trabajo (p. ej. la hora del día). En tiempo de ejecución Altova FlowForce Server pasa instrucciones de transformación a StyleVision Server, que ejecuta la transformación.

En el diagrama que aparece a continuación puede ver qué papel desempeña StyleVision Server en el flujo de trabajo de FlowForce Server. El diagrama también incluye MapForce Server porque también se pueden crear trabajos de FlowForce Server que envían asignaciones de [Altova MapForce](#) a [Altova MapForce Server](#) para que las ejecute.



StyleVision Server se puede invocar desde un trabajo de FlowForce Server o desde la línea de comandos. Consulte el apartado [Uso de la línea de comandos](#) ³⁴ para obtener más información.

2.2 Como producto servidor independiente

StyleVision Server se puede instalar como producto independiente en sistemas Windows, Linux y macOS. En la versión independiente las funciones de StyleVision Server se invocan desde la línea de comandos. Consulte el apartado [Uso de la línea de comandos](#)³⁴ para más información.

3 Configuración de StyleVision Server

En esta sección se explica cómo configurar StyleVision Server. La información se divide en tres partes:

- Información sobre [núcleos de procesador y licencias](#) ¹²
- Instalación y configuración de StyleVision Server en sistemas [Windows](#) ¹³
- Instalación y configuración de StyleVision Server en sistemas [Linux](#) ²⁰
- Instalación y configuración de StyleVision Server en sistemas [macOS](#) ²⁷

3.1 Núcleos de procesador y licencias

La asignación de licencias a productos servidor de Altova depende de cuántos núcleos físicos (en contraposición a núcleos lógicos) tiene el procesador del equipo donde se ejecuta el producto servidor de Altova. Por ejemplo, un procesador dual tiene dos núcleos, un procesador *quad* tiene cuatro núcleos, un procesador *hexa-core* tiene seis núcleos, y así sucesivamente. El número de núcleos de la licencia asignada a un producto debe ser mayor o igual al número de núcleos disponibles en dicho equipo servidor, ya sea un servidor físico o un equipo virtual. Por ejemplo, si un servidor tiene ocho núcleos (un procesador *octa-core*), deberá comprar una licencia para ocho núcleos. También puede combinar varias licencias para alcanzar el número de núcleos necesario. Es decir, puede usar dos licencias para cuatro núcleos para un servidor *octa-core* en lugar de una licencia para ocho núcleos, por ejemplo.

Si usa un equipo servidor con gran cantidad de núcleos, pero tiene un bajo volumen de procesamiento, también puede crear un equipo virtual que tenga adjudicados menos núcleos y comprar una licencia para ese menor número de núcleos. No obstante, dicha implementación será menos rápida que si utilizara todos los núcleos del equipo.

Nota: cada licencia de los productos servidor de Altova se puede usar de forma simultánea en un equipo como máximo (en el equipo donde está instalado el producto servidor de Altova), incluso si la capacidad de la licencia no está agotada. Por ejemplo, si utiliza una licencia para 10 núcleos para un equipo cliente que tiene 6 núcleos, los 4 núcleos restantes de la licencia no se pueden usar simultáneamente en otro equipo cliente.

Ejecución por subprocesos simples

Si su producto de Altova admite la ejecución por subprocesos simples verá que hay disponible la opción correspondiente. En estos casos, si en el repertorio de licencias hay una licencia de producto servidor de Altova para un solo núcleo, puede asignársela a un equipo que tenga varios núcleos. En este caso, el equipo ejecutará el producto en un solo núcleo. El procesamiento será lógicamente más lento porque solo se usa un núcleo. Es decir, el producto se ejecutará en modo de subprocesamiento simple.

Para asignar una licencia de un solo núcleo a un equipo con varios núcleos basta con marcar en LicenseServer la casilla *Limit to single thread execution* del producto.

Estimación de requisitos básicos

Además de lo mencionado anteriormente, tenga en cuenta que existen varios factores externos que suelen influir en los volúmenes y tiempos de procesamiento que su servidor puede manejar (por ejemplo, el hardware, la carga actual de la CPU, la memoria asignada a otras aplicaciones que se estén ejecutando en el servidor). Para poder conseguir un cálculo lo más exacto posible, recomendamos que primero ejecute las herramientas en su entorno para exponerlas a los factores y datos reales concretos de su negocio.

3.2 Instalación y configuración en Windows

Esta sección explica cómo [instalar](#)¹³ StyleVision Server y asignarle [licencias](#)¹⁶ en sistemas Windows.

[Instalación en Windows](#)¹³

- [Requisitos del sistema](#)¹³
- [Instalar StyleVision Server](#)¹³
- [Altova LicenseServer](#)¹³
- [Versiones de LicenseServer](#)¹³
- [Licencia de prueba](#)¹³
- [Ubicación de la carpeta de aplicación](#)¹³

[Asignación de licencias en Windows](#)¹⁶

- [Iniciar el controlador de servicios ServiceController](#)¹⁶
- [Iniciar LicenseServer](#)¹⁶
- [Registrar StyleVision Server](#)¹⁶
- [Asignar licencias](#)¹⁶

Nota: a partir de la versión 2021 no se puede instalar una versión de 32 bits de StyleVision Server sobre una de 64 bits o viceversa. Debe (i) desinstalar la versión antigua antes de instalar la nueva o (ii) actualizar su versión a una más reciente pero de la misma versión de bits que la que ya tenía.

3.2.1 Instalación en Windows

El proceso de instalación y configuración de StyleVision Server en Windows se describe a continuación.

▼ Requisitos del sistema

▼ Windows

Windows 7 SP1 con actualización de la plataforma, Windows 8, Windows 10

▼ Windows Server

Windows Server 2008 R2 SP1 con actualización de la plataforma o superior

▼ Instalar StyleVision Server

StyleVision Server se puede instalar en sistemas Windows:

- **Como producto servidor independiente llamado StyleVision Server:** para instalar StyleVision Server como producto independiente, descargue y ejecute el instalador de StyleVision Server y siga las instrucciones en pantalla.
- **Como parte del paquete de instalación de FlowForce Server:** para instalar StyleVision Server como parte del paquete de [FlowForce Server](#) descargue y ejecute el instalador de FlowForce Server. Siga las instrucciones en pantalla y marque la opción para instalar StyleVision Server.

Los programas de instalación de StyleVision Server y [FlowForce Server](http://www.altova.com/es/download.html) están disponibles en el centro de descargas de Altova (<http://www.altova.com/es/download.html>).

Una vez completada la instalación, el ejecutable de StyleVision Server estará en esta ubicación predeterminada:

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

El programa de instalación creará todos los registros necesarios para usar StyleVision Server desde la interfaz de COM, como interfaz Java y en el entorno .NET.

Instalación en Windows Server Core

Windows Server Core es una instalación ligera de Windows que no usa todas las características de la IGU. Para instalar StyleVision Server en un equipo con Windows Server Core:

1. Descargue el instalador de StyleVision Server desde el Centro de descargas de Altova. Este archivo se llama `stylevisionserver<version>.exe`. Asegúrese de que escoge el ejecutable que coincide con la plataforma de su servidor (32 bits o 64 bits).
2. En un equipo Windows estándar (no en el equipo con Windows Server Core) ejecute el comando `stylevisionserver<version>.exe /u`. Este comando abre los archivos `.msi` que necesita en el paso siguiente.
3. Copie el archivo `.msi` sin descomprimir en el equipo en el que está Windows Server Core.
4. Si está actualizando una versión antigua de StyleVision Server cierre StyleVision Server antes de ejecutar el paso siguiente.
5. Use el archivo `.msi` para la instalación; para ello ejecute el comando `msiexec /i stylevisionserver.msi`. Este comando inicia la instalación en Windows Server Core.

Mantenga los archivos `.msi` en un lugar seguro. Los necesitará más tarde para desinstalar, reparar o modificar sus instalaciones.

Para probar el valor de retorno de la instalación puede que quiera ejecutar un script parecido al que mostramos a continuación. El código de retorno estará en la variable de entorno `%errorlevel%`. El código de retorno `0` indica que la operación se ha realizado correctamente.

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

Si prefiere realizar una instalación silenciosa con un código de retorno y un registro del proceso de instalación use este comando:

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

Para modificar la instalación, ejecute:

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

Para modificar la instalación, ejecute:

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

Para desinstalar StyleVision Server, ejecute:

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

Para una desinstalación silenciosa de StyleVision Server y obtener un registro detallado del resultado en un archivo de registro:

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

Nota: en Windows Server Core las funciones de gráficos y de código de barras de StyleVision Server no estarán disponibles. Para instalar taxonomías use el Gestor de paquetes de taxonomías desde la línea de comandos. Consulte el manual de StyleVision Server para más información al respecto

▼ Altova LicenseServer

- Para que StyleVision Server funcione debe tener asignada una licencia desde un servidor Altova LicenseServer de la red.
- El programa de instalación de StyleVision Server o de [FlowForce Server](#) para sistemas Windows ofrece una opción para descargar e instalar Altova LicenseServer junto con StyleVision Server o con [FlowForce Server](#).
- Si en la red ya hay instalado un servidor Altova LicenseServer, no necesita instalar otro LicenseServer a no ser que se necesite una versión más reciente (ver el siguiente apartado [versiones de LicenseServer](#)).
- Durante el proceso de instalación de StyleVision Server o de [FlowForce Server](#), puede seleccionar si también se instala Altova LicenseServer. Tenga en cuenta esta información:
 - a. Si todavía no ha instalado Altova LicenseServer, no cambie las opciones seleccionadas por defecto. El asistente instalará la versión más reciente de Altova LicenseServer en el equipo donde se está ejecutando el asistente.
 - b. Si todavía no ha instalado Altova LicenseServer y quiere instalarlo en otro equipo, desactive la casilla *Instalar Altova LicenseServer en este equipo* y elija el botón de opción *Registrar más tarde*. En este caso deberá instalar Altova LicenseServer y registrar StyleVision Server por separado.
 - c. Si Altova LicenseServer ya está instalado en el equipo pero se trata de una versión anterior a la indicada por el asistente para la instalación, no cambie las opciones seleccionadas por defecto. En este caso, el asistente actualizará automáticamente la versión de LicenseServer, instalando la versión indicada en pantalla. Recuerde que la información de registro y de licencias disponible se conservará una vez completada la actualización.
 - d. Si Altova LicenseServer ya está instalado en el equipo o en la red y se trata de la misma versión que la indicada por el asistente para la instalación, entonces siga estos pasos:
 - i. Desactive la casilla *Instalar Altova LicenseServer en este equipo*.
 - ii. Seleccione el botón de opción *Registrar este producto con Servidor LicenseServer de:* y en el menú desplegable elija la instancia de Altova LicenseServer con la que quiere registrar StyleVision Server. También puede elegir *Registrar más tarde*. Siempre puede elegir la opción *Registrar más tarde* si no quiere asociar el producto con LicenseServer y continuar con la instalación de StyleVision Server.

Para más información sobre cómo registrar StyleVision Server y asignarle licencias con Altova LicenseServer, consulte la sección [Asignación de licencias en Windows](#) ¹⁶.

▼ Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia con la versión de LicenseServer correspondiente a la versión de StyleVision Server instalada o con una versión posterior de LicenseServer.
- La versión de LicenseServer correspondiente a la versión de StyleVision Server aparece en pantalla durante la instalación de StyleVision Server. Puede instalar esta versión de LicenseServer junto con StyleVision Server o puede instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores. El programa de instalación de LicenseServer se encarga de esto automáticamente si detecta versiones más recientes en el sistema.
- Las versiones de LicenseServer son compatibles y funcionan con versiones más antiguas de StyleVision Server.
- Si instala una versión nueva de StyleVision Server y la versión de LicenseServer que está instalada es anterior a la que le corresponde, instale la versión más reciente que está siempre disponible en el sitio web de Altova.
- Cuando se desinstala LicenseServer, todos los datos de registro y asignación de licencias almacenados en la versión antigua de LicenseServer se guardan en una base de datos en el equipo servidor. Estos datos se importan de forma automática a la siguiente versión que se instale en el equipo.
- El número de versión de LicenseServer siempre aparece al final de la página de configuración de LicenseServer.

Versión actual: 3.6

▼ Licencia de prueba

Durante el proceso de instalación tendrá la opción de solicitar una licencia de prueba de 30 días para StyleVision Server. Altova le enviará un correo electrónico con la licencia de prueba a la dirección de correo que usted indique en el formulario.

▼ Ubicación de la carpeta de la aplicación

La aplicación se instalará en esta carpeta:

Windows 7, 8 y 10	C:\Archivos de programa\Altova\
Versión de 32 bits en sistemas operativos de 64 bits	C:\Archivos de programa (x86)\Altova\

3.2.2 Asignación de licencias en Windows

Para poder trabajar con StyleVision Server es necesario asignarle una licencia con Altova LicenseServer. La asignación de licencias es un proceso de dos pasos:

1. El primero consiste en **registrar StyleVision Server** con LicenseServer desde StyleVision Server.
2. El segundo paso consiste en **asignar una licencia** a StyleVision Server desde LicenseServer. Descargue la versión más reciente de LicenseServer desde [el sitio web de Altova](#) e instálelo en su equipo local o en uno de los equipos de su red.

Más abajo explicamos qué pasos debe seguir. Para información más detallada consulte el [manual del usuario de LicenseServer](#) en [el sitio web de Altova](#).

▼ Iniciar el controlador de servicios ServiceController

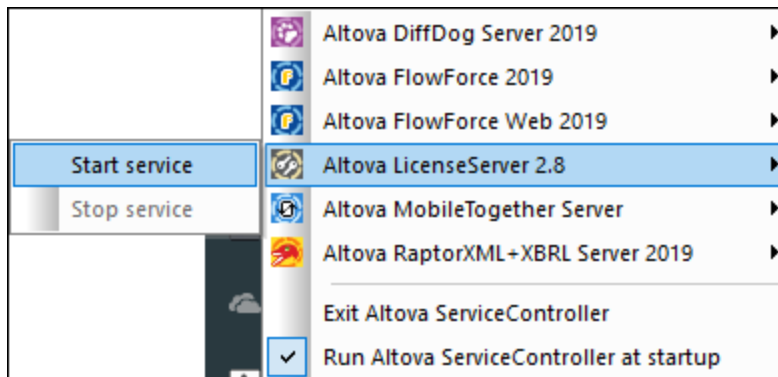
Altova ServiceController se inicia para arrancar Altova LicenseServer.

Altova ServiceController (en adelante *ServiceController*) es una práctica aplicación que sirve para iniciar, detener y configurar los servicios de Altova **en sistemas Windows**.

ServiceController se instala con Altova LicenseServer y con *los productos servidor de Altova que se instalan como servicios* (FlowForce Server, RaptorXML(+XBRL) Server y Mobile Together Server). Se puede iniciar haciendo clic en **Inicio | Altova LicenseServer | Altova ServiceController**. (Este comando también está en las carpetas del menú **Inicio** de *los productos servidor de Altova que se instalan como servicios* (FlowForce Server, RaptorXML(+XBRL) Server y Mobile Together Server).) Una vez iniciado, podrá acceder a ServiceController desde la bandeja del sistema (*imagen siguiente*).

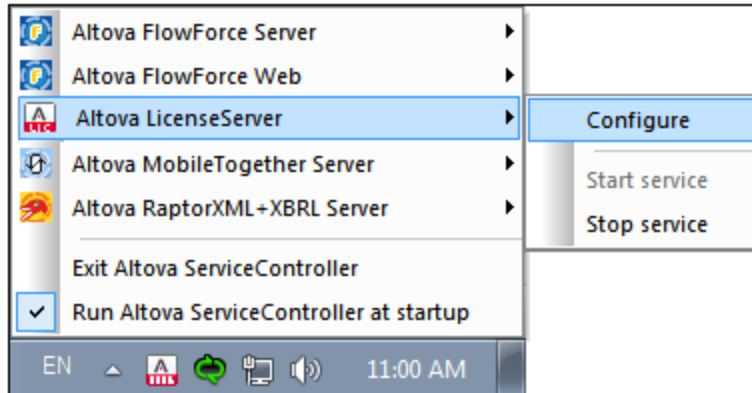


Si quiere que ServiceController se inicie automáticamente nada más iniciar sesión en el sistema, haga clic en el icono de ServiceController de la bandeja del sistema para abrir el menú de opciones de **ServiceController** (*imagen siguiente*) y active la opción **Run Altova ServiceController at Startup** (*Ejecutar Altova ServiceController al inicio*), que de todas maneras es la opción predeterminada. Para cerrar ServiceController haga clic en el icono de ServiceController de la bandeja del sistema y en el menú haga clic en la opción **Exit Altova ServiceController** (Salir de Altova ServiceController).



▼ Iniciar LicenseServer

Para iniciar LicenseServer haga clic en el icono de **ServiceController** en la bandeja del sistema, pase el puntero del ratón por encima de la opción **Altova LicenseServer** del menú emergente (*imagen siguiente*) y seleccione el comando **Start service** en el submenú. Si LicenseServer ya está en ejecución, este comando estará deshabilitado.



▼ Registrar StyleVision Server

Para registrar StyleVision Server desde la interfaz de la línea de comandos utilice el comando `licenseserver`:

```
StyleVisionServer licenseserver [opciones] NombreServidor-O--Dirección-IP
```

Por ejemplo, si `localhost` es el nombre del servidor donde está instalado LicenseServer:

```
StyleVisionServer licenseserver localhost
```

Si StyleVision Server se instaló durante la instalación de [FlowForce Server](#) y registró FlowForce Server con LicenseServer, entonces StyleVision Server también se registró automáticamente. Estos son los pasos que debe seguir: (i) Inicie Altova FlowForce Web como servicio con el controlador de servicios de Altova ServiceController (véase el punto anterior); (ii) Introduzca su contraseña para acceder a la página de configuración web; (iii) Seleccione el nombre o la dirección del servidor LicenseServer y haga clic en el botón **Registrarse con LicenseServer**. Para más información consulte el apartado Registrar FlowForce Server.

Una vez finalizado el proceso de registro, abra la pestaña de gestión de servidores de la página de configuración de LicenseServer para asignar una licencia a StyleVision Server.

▼ Asignar licencias

Tras registrarse con LicenseServer, StyleVision Server aparecerá en la lista de la pestaña de gestión de servidores **Client Management** de la página de configuración de LicenseServer. En esta pestaña puede asignar una licencia a StyleVision Server.

La asignación de licencias a productos servidor de Altova depende de cuántos núcleos físicos (en contraposición a núcleos lógicos) tiene el procesador del equipo donde se ejecuta el producto servidor de Altova. Por ejemplo, un procesador dual tiene dos núcleos, un procesador *quad* tiene cuatro núcleos, un procesador *hexa-core* tiene seis núcleos, y así sucesivamente. El número de núcleos de la licencia asignada a un producto debe ser mayor o igual al número de núcleos disponibles en dicho equipo servidor, ya sea un servidor físico o un equipo virtual.

Por ejemplo, si un servidor tiene ocho núcleos (un procesador *octa-core*), deberá comprar una licencia para ocho núcleos. También puede combinar varias licencias para alcanzar el número de núcleos necesario. Es decir, puede usar dos licencias para cuatro núcleos para un servidor *octa-core* en lugar de

una licencia para ocho núcleos, por ejemplo.

Si usa un equipo servidor con gran cantidad de núcleos, pero tiene un bajo volumen de procesamiento, también puede crear un equipo virtual que tenga adjudicados menos núcleos y comprar una licencia para ese menor número de núcleos. No obstante, dicha implementación será menos rápida que si utilizara todos los núcleos disponibles en el servidor.

Nota: cada licencia de los productos servidor de Altova se puede usar de forma simultánea en un equipo como máximo (en el equipo donde está instalado el producto servidor de Altova), incluso si la capacidad de la licencia no está agotada. Por ejemplo, si utiliza una licencia para 10 núcleos para un equipo cliente que tiene 6 núcleos, los 4 núcleos restantes de la licencia no se pueden usar simultáneamente en otro equipo cliente.

3.3 Instalación y configuración en Linux

Esta sección explica cómo [instalar](#)²⁰ StyleVision Server y [asignarle licencias](#)²³ en sistemas Linux (Debian, Ubuntu, CentOS, RedHat).

[Instalación en Linux](#)²⁰

- [Requisitos del sistema](#)²⁰
- [Desinstalar versiones antiguas de los productos servidor de Altova](#)²⁰
- [Descargar el paquete de instalación para Linux](#)²⁰
- [Instalar StyleVision Server](#)²⁰
- [Altova LicenseServer](#)²⁰
- [Versiones de LicenseServer](#)²⁰

[Asignación de licencias en Linux](#)²³

- [Iniciar LicenseServer](#)²³
- [Registrar StyleVision Server](#)²³
- [Asignar licencias](#)²⁴

[Notas sobre configuración del entorno](#)²⁵

3.3.1 Instalación en Linux

El proceso de instalación y configuración de StyleVision Server en Linux se describe a continuación.

▼ Requisitos del sistema

▼ *Linux*

- CentOS 7 o superior
- RedHat 7 o superior
- Debian 8 o superior
- Ubuntu 16.04 LTS o superior

Las bibliotecas que aparecen a continuación son un requisito previo para la instalación y ejecución de la aplicación. Si los paquetes que aparecen en esta tabla no están en su equipo Linux, ejecute el comando `yum` (o `apt-get` si procede) para instalarlos.

Requisito para	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

Nota: si quiere usar la función de gráficos de Altova, entonces debe haber al menos una fuente instalada en su sistema para que los gráficos se muestren correctamente. Para ver una lista de las fuentes instaladas use, por ejemplo, el comando `fc-list` de la [Fontconfig library](#).

▼ Integración con FlowForce Server

Si está instalando StyleVision Server junto con FlowForce Server, recomendamos instalar primero FlowForce Server. Si no lo hace así, después de instalar StyleVision Server y FlowForce Server, ejecute este comando:

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

Este comando copia el archivo `.tool` del directorio `/etc` de StyleVision Server al directorio `/tools` de FlowForce Server. El archivo `.tool` es un archivo que FlowForce Server necesita porque contiene la ruta de acceso del ejecutable de StyleVision Server. Si instala FlowForce Server antes que StyleVision Server entonces no es necesario ejecutar este comando.

▼ Desinstalar versiones antiguas de los productos servidor de Altova

En la interfaz de la línea de comandos de Linux puede comprobar si ya hay productos servidor de Altova instalados en el equipo. Para ello use este comando:

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

Si StyleVision Server no está instalado, continúe con la instalación tal y como se describe más abajo.

Si StyleVision Server ya está instalado y quiere instalar una versión más reciente, antes debe desinstalar la versión previa con este comando:

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

Si quiere desinstalar una versión previa de Altova LicenseServer, use este comando:

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

En los sistemas Debian y Ubuntu puede pasar que StyleVision Server siga apareciendo en la lista de productos instalados incluso después de haber desinstalado la aplicación. En este caso ejecute el comando `purge` para que desaparezca de la lista. También puede usar este comando en lugar del comando `remove`.

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

▼ Descargar el paquete de instalación para Linux

Los paquetes de instalación de StyleVision Server para sistemas Linux se pueden descargar del [sitio web de Altova](#).

Distribución	Extensión del paquete
Debian	.deb
Ubuntu	.deb
CentOS	.rpm

RedHat	.rpm
--------	------

Tras descargarlo, copie el paquete de instalación en cualquier directorio del sistema Linux. Para ejecutar StyleVision Server es necesario tener instalado Altova LicenseServer, que también se puede descargar del [sitio web de Altova](#).

▼ Instalar StyleVision Server

En una ventana de la Terminal, cambie al directorio donde copió el paquete de instalación para Linux. Por ejemplo, si lo copió en un directorio del usuario llamado `MiAltova` (ubicado en `/home/User` por ejemplo), cambie a ese directorio con esta línea de comandos:

```
cd /home/User/MiAltova
```

Instale StyleVision Server con este comando:

```
[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
```

La aplicación StyleVision Server se instala en este directorio:

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

▼ Altova LicenseServer

Para poder ejecutar los productos servidores de Altova, incluido StyleVision Server, es necesario asignarles una licencia con un servidor Altova LicenseServer de la red.

En los sistemas Linux es necesario instalar Altova LicenseServer por separado. Por tanto, descargue Altova LicenseServer del [sitio web de Altova](#) y copie el paquete de instalación en cualquier directorio. Siga las instrucciones anteriores para instalar LicenseServer (*ver apartado anterior*).

```
[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
```

La aplicación LicenseServer se instala en este directorio:

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

Consulte el apartado siguiente [Asignación de licencias en Linux](#) ²³ para obtener información sobre cómo registrar StyleVision Server con Altova LicenseServer y asignarle licencias. Consulte también la [documentación del LicenseServer](#) para información más detallada.

▼ Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia con la versión de LicenseServer

correspondiente a la versión de StyleVision Server instalada o con una versión posterior de LicenseServer.

- La versión de LicenseServer correspondiente a la versión de StyleVision Server aparece en pantalla durante la instalación de StyleVision Server. Puede instalar esta versión de LicenseServer junto con StyleVision Server o puede instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores. El programa de instalación de LicenseServer se encarga de esto automáticamente si detecta versiones más recientes en el sistema.
- Las versiones de LicenseServer son compatibles y funcionan con versiones más antiguas de StyleVision Server.
- Si instala una versión nueva de StyleVision Server y la versión de LicenseServer que está instalada es anterior a la que le corresponde, instale la versión más reciente que está siempre disponible en el sitio web de Altova.
- Cuando se desinstala LicenseServer, todos los datos de registro y asignación de licencias almacenados en la versión antigua de LicenseServer se guardan en una base de datos en el equipo servidor. Estos datos se importan de forma automática a la siguiente versión que se instale en el equipo.
- El número de versión de LicenseServer siempre aparece al final de la página de configuración de LicenseServer.

Versión actual: 3.6

3.3.2 Asignación de licencias en Linux

Para poder trabajar con StyleVision Server es necesario asignarle una licencia con Altova LicenseServer. La asignación de licencias es un proceso de dos pasos:

1. El primero consiste en **registrar StyleVision Server** con LicenseServer desde StyleVision Server.
2. El segundo paso consiste en **asignar una licencia** a StyleVision Server desde LicenseServer. Descargue la versión más reciente de LicenseServer desde [el sitio web de Altova](#) e instálelo en su equipo local o en uno de los equipos de su red.

Más abajo explicamos qué pasos debe seguir. Para información más detallada consulte el [manual del usuario de LicenseServer](#) en [el sitio web de Altova](#).

▼ Iniciar LicenseServer

Para poder registrar StyleVision Server con LicenseServer y asignarle una licencia, LicenseServer debe estar en ejecución como servicio. Inicie LicenseServer como servicio con este comando:

[< 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 por cualquier motivo necesita detener LicenseServer, use el mismo comando pero sustituya `stop` por `start`. Por ejemplo:

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

▼ Registrar StyleVision Server

Para registrar StyleVision Server desde la interfaz de la línea de comandos utilice el comando `licenseserver`:

```
sudo /opt/Altova/StyleVisionServer2021/bin/stylevisionserver licenseserver  
[opciones] NombreServidor-O-Dirección-IP
```

Por ejemplo, si el nombre del servidor donde está instalado LicenseServer es `localhost`:

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

En el comando anterior `localhost` es el nombre del servidor donde está instalado LicenseServer. Observe también la ubicación del ejecutable de StyleVision Server:

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

Una vez completado el registro, abra la pestaña **Client Management** de la página de configuración de LicenseServer para asignar una licencia a StyleVision Server.

▼ Asignar licencias

Tras registrarse con LicenseServer, StyleVision Server aparecerá en la lista de la pestaña de gestión de servidores **Client Management** de la página de configuración de LicenseServer. En esta pestaña puede asignar una licencia a StyleVision Server.

La asignación de licencias a productos servidor de Altova depende de cuántos núcleos físicos (en contraposición a núcleos lógicos) tiene el procesador del equipo donde se ejecuta el producto servidor de Altova. Por ejemplo, un procesador dual tiene dos núcleos, un procesador *quad* tiene cuatro núcleos, un procesador *hexa-core* tiene seis núcleos, y así sucesivamente. El número de núcleos de la licencia asignada a un producto debe ser mayor o igual al número de núcleos disponibles en dicho equipo servidor, ya sea un servidor físico o un equipo virtual.

Por ejemplo, si un servidor tiene ocho núcleos (un procesador *octa-core*), deberá comprar una licencia para ocho núcleos. También puede combinar varias licencias para alcanzar el número de núcleos necesario. Es decir, puede usar dos licencias para cuatro núcleos para un servidor *octa-core* en lugar de una licencia para ocho núcleos, por ejemplo.

Si usa un equipo servidor con gran cantidad de núcleos, pero tiene un bajo volumen de procesamiento, también puede crear un equipo virtual que tenga adjudicados menos núcleos y comprar una licencia para ese menor número de núcleos. No obstante, dicha implementación será menos rápida que si utilizara todos los núcleos disponibles en el servidor.

Nota: cada licencia de los productos servidor de Altova se puede usar de forma simultánea en un equipo como máximo (en el equipo donde está instalado el producto servidor de Altova), incluso si la capacidad de la licencia no está agotada. Por ejemplo, si utiliza una licencia para 10 núcleos para un equipo cliente que tiene 6 núcleos, los 4 núcleos restantes de la licencia no se pueden usar simultáneamente en otro equipo cliente.

3.3.3 Notas sobre configuración del entorno

Carpetas

A continuación enumeramos carpetas importantes de su sistema StyleVision Server.

▣ Directorio raíz de instalación

`/opt/Altova/StyleVisionServer2021/`

▣ Archivos de licencia

`/var/opt/Altova/StyleVisionServer`

▣ Parámetros del entorno

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

El archivo de parámetros del entorno (por lo general llamado `jdbc.sh`) se ejecuta al iniciar el sistema. Las definiciones que contiene deben ser específicas para un entorno en concreto. La ruta de acceso anterior es un ejemplo solamente.

Nota: el archivo de parámetros del entorno establece las variables para **todos los usuarios** del sistema, por lo que debe tener cuidado al modificar las opciones. Por ejemplo, si modifica la ruta de una clase en este archivo, el cambio tendrá lugar en todo el sistema. Si quiere realizar cambios que afecten únicamente a StyleVision Server, recomendamos que considere usar un fichero de unidades (véase *más abajo Conexiones JDBC*).

Conexiones de BD

En Linux se admiten las siguientes conexiones de BD:

- JDBC: puede usar JDBC para todas las BD compatibles salvo para Microsoft Access
- Conexiones nativas: actualmente disponibles para las bases de datos SQLite y PostgreSQL

Si usa JDBC debe tener en cuenta estos aspectos:

- Debe tener instalado Java Runtime Environment o el kit de desarrollo de software.
- Debe tener instalados controladores JDBC para la base de datos de destino.
- Debe establecer correctamente estas variables de entorno:
 - `CLASSPATH`: para encontrar los archivos jar que se conectan a la base de datos JDBC; los archivos jar se pueden introducir en (i) un archivo ejecutable (como `jdbc.sh`) que se ejecuta al iniciar el sistema o (ii) en un fichero de unidades que se ejecuta cuando StyleVision Server se inicia como servicio. Usar un fichero de unidades para especificar los archivos jar tiene la ventaja de que los

archivos necesarios para las conexiones JDBC de StyleVision Server se ubican sin que tenga que modificar la configuración del sistema. Más abajo se muestra un fichero de unidades.

- o PATH: para buscar el entorno JRE, aunque a veces no es necesaria, dependiendo de la instalación.
- o JAVA_HOME: a veces no es necesaria, dependiendo de la instalación.

Lista de archivos importantes

Puede copiar este script de shell (o archivo de unidades) en la carpeta `/opt/Altova/StyleVisionServer/etc` para no sobrescribir los archivos de configuración actuales. Realice los cambios que necesite en el script. Consulte también el apartado anterior *Conexiones JDBC*. Las partes que aparecen resaltadas en azul son propias del entorno y deberán ajustarse.

Script de shell (fichero de unidades)

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

3.4 Instalación y configuración en macOS

Esta sección explica cómo [instalar](#)²⁷ StyleVision Server y [asignarle licencias](#)²⁹ en sistemas macOS.

[Instalación en macOS](#)²⁷

- [Requisitos del sistema](#)²⁷
- [Desinstalar versiones previas de productos servidor de Altova](#)²⁷
- [Descargar el paquete de instalación para macOS](#)²⁷
- [Instalar StyleVision Server](#)²⁷
- [Altova LicenseServer](#)²⁷
- [Versiones de LicenseServer](#)²⁷

[Asignación de licencias en macOS](#)²⁹

- [Iniciar LicenseServer](#)²⁹
- [Registrar StyleVision Server](#)²⁹
- [Asignar licencias](#)²⁹

[Notas sobre configuración del entorno](#)³¹

3.4.1 Instalación en macOS

El proceso de instalación y configuración de StyleVision Server en macOS se describe a continuación.

▼ Requisitos del sistema

▶ [macOS](#)

macOS 10.13 o superior

▼ Integración con FlowForce Server

Si instala StyleVision Server junto con FlowForce Server, recomendamos instalar primero FlowForce Server. Si no lo hace así, después de instalar StyleVision Server y FlowForce Server, ejecute este comando:

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

Este comando copia el archivo `.tool` del directorio `/etc` de StyleVision Server al directorio `/tools` de FlowForce Server. El archivo `.tool` es un archivo que FlowForce Server necesita porque contiene la ruta de acceso del ejecutable de StyleVision Server. Si instala FlowForce Server antes que StyleVision Server entonces no es necesario ejecutar este comando.

▼ Desinstalar versiones antiguas de los productos servidor de Altova

En la terminal de Aplicaciones haga clic con el botón derecho en el icono de StyleVision Server y seleccione **Mover a la papelera**. La aplicación se envía a la papelera pero debe quitar la aplicación de la carpeta `usr`. Para ello puede utilizar este comando:

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

Si necesita desinstalar una versión antigua de Altova LicenseServer, antes debe detener el servicio con este comando:

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

Para comprobar si el servicio se detuvo correctamente, abra la terminal del Monitor de actividad y confirme que LicenseServer no está en la lista. Después desinstale LicenseServer siguiendo las instrucciones dadas más arriba para StyleVision Server.

▼ Descargar el archivo de imagen de disco

Descargue el archivo de imagen de disco (.dmg) del sitio web de Altova (<https://www.altova.com/es/download.html>).

▼ Instalar StyleVision Server

Haga clic en el archivo de imagen de disco (.dmg) para abrirlo. El programa de instalación de StyleVision Server aparece como unidad virtual en el equipo. En esta unidad virtual nueva haga doble clic en el paquete de instalación (.pkg). Siga las instrucciones que aparecen en pantalla y acepte el contrato de licencia. Para expulsar la unidad cuando termine la instalación, haga clic con el botón derecho en la unidad y seleccione **Expulsar**.

El paquete de StyleVision Server se instalará en esta carpeta:

```
/usr/local/Altova/StyleVisionServer2021 (archivos binarios de la aplicación)  
/var/Altova/StyleVisionServer (archivos de datos como bases de datos y registros)
```

▼ Altova LicenseServer

Para poder ejecutar los productos servidor de Altova, incluido StyleVision Server, es necesario asignarles una licencia desde un servidor Altova LicenseServer que esté instalado en la red.

El paquete de instalación de Altova LicenseServer está disponible en la unidad virtual creado en el paso anterior. Para instalar Altova LicenseServer haga doble clic en el paquete de instalación que está en la unidad virtual y siga las instrucciones que aparecen en pantalla. También debe aceptar el contrato de licencia para poder continuar con la instalación.

Altova LicenseServer también puede descargarse desde el sitio web de Altova (<https://www.altova.com/es/download.html>) e instalarse por separado.

El paquete de LicenseServer se instalará en esta carpeta:

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

Para obtener más información sobre cómo registrar StyleVision Server con Altova LicenseServer y asignarle una licencia, consulte el apartado [Asignación de licencias en macOS](#)²⁹.

▼ Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia con la versión de LicenseServer correspondiente a la versión de StyleVision Server instalada o con una versión posterior de LicenseServer.
- La versión de LicenseServer correspondiente a la versión de StyleVision Server aparece en pantalla durante la instalación de StyleVision Server. Puede instalar esta versión de LicenseServer junto con StyleVision Server o puede instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores. El programa de instalación de LicenseServer se encarga de esto automáticamente si detecta versiones más recientes en el sistema.
- Las versiones de LicenseServer son compatibles y funcionan con versiones más antiguas de StyleVision Server.
- Si instala una versión nueva de StyleVision Server y la versión de LicenseServer que está instalada es anterior a la que le corresponde, instale la versión más reciente que está siempre disponible en el sitio web de Altova.
- Cuando se desinstala LicenseServer, todos los datos de registro y asignación de licencias almacenados en la versión antigua de LicenseServer se guardan en una base de datos en el equipo servidor. Estos datos se importan de forma automática a la siguiente versión que se instale en el equipo.
- El número de versión de LicenseServer siempre aparece al final de la página de configuración de LicenseServer.

Versión actual: 3.6

3.4.2 Asignación de licencias en macOS

Para poder trabajar con StyleVision Server es necesario asignarle una licencia con Altova LicenseServer. La asignación de licencias es un proceso de dos pasos:

1. El primero consiste en **registrar StyleVision Server** con LicenseServer desde StyleVision Server.
2. El segundo paso consiste en **asignar una licencia** a StyleVision Server desde LicenseServer. Descargue la versión más reciente de LicenseServer desde [el sitio web de Altova](#) e instálelo en su equipo local o en uno de los equipos de su red.

Más abajo explicamos qué pasos debe seguir. Para información más detallada consulte el [manual del usuario de LicenseServer](#) en [el sitio web de Altova](#).

▼ Iniciar LicenseServer

Para registrar y asignar una licencia correctamente a StyleVision Server, LicenseServer debe estar en ejecución como demonio. Inicie LicenseServer como demonio con este comando:

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

Si por cualquier motivo necesita detener LicenseServer, use el mismo comando pero sustituya `load` por `unload`. Por ejemplo:

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

▼ Registrar StyleVision Server

Para registrar StyleVision Server desde la interfaz de la línea de comandos utilice el comando

`licenseserver`:

```
sudo /usr/local/Altova/StyleVisionServer2021/bin/StyleVisionServer licenseserver  
[opciones] NombreServidor-O-Dirección-IP
```

Por ejemplo, si el nombre del servidor donde está instalado LicenseServer es `localhost`:

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

En el comando anterior `localhost` es el nombre del servidor donde está instalado LicenseServer. Observe también la ubicación del ejecutable de StyleVision Server:

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

Una vez completado el registro, abra la pestaña **Client Management** de la página de configuración de LicenseServer para asignar una licencia a StyleVision Server..

▼ Asignar licencias

Tras registrarse con LicenseServer, StyleVision Server aparecerá en la lista de la pestaña de gestión de servidores **Client Management** de la página de configuración de LicenseServer. En esta pestaña puede asignar una licencia a StyleVision Server.

La asignación de licencias a productos servidor de Altova depende de cuántos núcleos físicos (en contraposición a núcleos lógicos) tiene el procesador del equipo donde se ejecuta el producto servidor de Altova. Por ejemplo, un procesador dual tiene dos núcleos, un procesador *quad* tiene cuatro núcleos, un procesador *hexa-core* tiene seis núcleos, y así sucesivamente. El número de núcleos de la licencia asignada a un producto debe ser mayor o igual al número de núcleos disponibles en dicho equipo servidor, ya sea un servidor físico o un equipo virtual.

Por ejemplo, si un servidor tiene ocho núcleos (un procesador *octa-core*), deberá comprar una licencia para ocho núcleos. También puede combinar varias licencias para alcanzar el número de núcleos necesario. Es decir, puede usar dos licencias para cuatro núcleos para un servidor *octa-core* en lugar de una licencia para ocho núcleos, por ejemplo.

Si usa un equipo servidor con gran cantidad de núcleos, pero tiene un bajo volumen de procesamiento, también puede crear un equipo virtual que tenga adjudicados menos núcleos y comprar una licencia para ese menor número de núcleos. No obstante, dicha implementación será menos rápida que si utilizara todos los núcleos disponibles en el servidor.

Nota: cada licencia de los productos servidor de Altova se puede usar de forma simultánea en un equipo como máximo (en el equipo donde está instalado el producto servidor de Altova), incluso si la capacidad de la licencia no está agotada. Por ejemplo, si utiliza una licencia para 10

núcleos para un equipo cliente que tiene 6 núcleos, los 4 núcleos restantes de la licencia no se pueden usar simultáneamente en otro equipo cliente.

3.4.3 Notas sobre configuración del entorno

Carpetas

A continuación enumeramos carpetas importantes de su sistema StyleVision Server.

☐ Directorio raíz de instalación

`/usr/local/Altova/StyleVisionServer2021/`

☐ Archivos de licencias

`/var/Altova/StyleVisionServer`

☐ Parámetros del entorno

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

El archivo de parámetros del entorno debe definirse en función del entorno de cada usuario. La ruta de acceso anterior es un ejemplo solamente.

Nota: estas variables del entorno solamente se establecen para el proceso de StyleVision Server y por tanto no afectan a los demás usuarios.

Conexiones de BD

En macOS se admiten las siguientes conexiones de BD:

- JDBC: puede usar JDBC para todas las BD compatibles salvo para Microsoft Access
- Conexiones nativas: actualmente disponibles para las bases de datos SQLite y PostgreSQL

Si usa JDBC debe tener en cuenta estos aspectos:

- Debe tener instalado Java Runtime Environment o el kit de desarrollo de software.
- Debe tener instalados controladores JDBC para la base de datos de destino.
- Debe establecer correctamente estas variables de entorno:
 - `CLASSPATH`: para encontrar los archivos jar: la ruta de la clase se determina en el archivo `plist`.
 - `PATH`: para buscar el entorno JRE, aunque a veces no es necesaria, dependiendo de la instalación.
 - `JAVA_HOME`: a veces no es necesaria, dependiendo de la instalación.

Java 6 para formularios PDF editables

Si en uno de estos archivos PDF faltan partes editables al abrirlo en un sistema Mac OS, esto puede deberse a que Java 6 no esté instalado en el equipo. Si es así, puede instalar Java 6 desde https://support.apple.com/kb/dl1572?locale=en_US. Si ya tiene instalada una versión más reciente que Java 6,

entonces la instalación de Java 6 no afectará al funcionamiento de la nueva versión, que será la versión predeterminada del sistema.

3.5 Notas sobre la configuración

Para poder ejecutar en Windows los ejemplos que vienen incluidos con StyleVision Server y que puede encontrar en la subcarpeta `etc\examples` de la carpeta de la aplicación, StyleVision Server DLL debe estar correctamente registrado en el sistema. Puede generarse un error de registro si ya se han instalado distintas versiones de StyleVision Server (de 32 y 64 bits) en un mismo equipo.

Para registrar correctamente el DLL de StyleVision Server (de 32 o de 64 bits) en equipos Windows, siga estos pasos:

1. Abra la línea de comandos en modo administrador
2. Vaya a la carpeta en la que está el DLL. Para ello puede usar este comando: `cd C:\Program Files\Altova\StyleVisionServer2021\bin`
3. Ejecute este comando para registrar el DLL (versión de 32 o de 64 bits): `regsvr32 styleVisionServer.dll`
4. Debería aparecer un cuadro de diálogo que confirme que el registro se ha efectuado correctamente
5. Abra Visual Studio
6. Cargue el proyecto con `styleVisionServerAPI_Sample.sln`
7. Confirme que las rutas de su archivo `Program.cs` son válidas
8. Ejecute el proyecto con **Ctrl+F5**

Nota: la ruta de acceso a la carpeta de la aplicación en sistemas Windows suele ser: `C:\Program Files\Altova\StyleVisionServer2021`.

4 Línea de comandos de StyleVision Server

Ubicación predeterminada del ejecutable de StyleVision Server

A continuación se muestra la ubicación predeterminada del ejecutable de StyleVision Server en las plataformas compatibles:

<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><CarpetaArchivosPrograma>\Altova\StyleVisionServer2021\bin\StyleVisionServer.exe</code>

Uso de la línea de comandos y lista de comandos

La sintaxis de la línea de comandos es:

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

- `--help` (`--h` en versión corta) muestra el texto de ayuda del comando dado. Si no se indica ningún comando, entonces se enumeran todos los comandos del ejecutable, cada uno con una breve descripción.
- `--version` muestra el número de versión de StyleVision Server.
- `<command>` es el comando que se ejecuta. Los comandos se describen en las subsecciones de este apartado (véase la lista más abajo).
- `[opciones]` son las opciones de un comando; se enumeran y describen con sus comandos correspondientes.
- `[arguments]` son los argumentos de un comando; se enumeran y describen con sus comandos correspondientes.

▼ Uso de mayúsculas/minúsculas y de barras diagonales en la línea de comandos

`StyleVisionServer` en Windows
`stylevisionserver` en Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que la mezcla de mayúsculas y minúsculas (`StyleVisionServer`) sólo funciona en Windows y Mac.

* En Linux y Mac utilice barras diagonales. En Windows utilice barras diagonales inversas.

Comandos CLI

A continuación se enumeran los comandos disponibles, que se explican en las subsecciones de este apartado.

- [assignlicense](#)³⁶: carga una licencia a LicenseServer y asigna esa licencia a StyleVision Server.
- [exportresourcestrings](#)³⁸: exporta todas las cadenas de los recursos de la aplicación a un archivo XML.
- [generate](#)⁴⁰: genera uno o varios documentos a partir de un archivo XML de entrada y una hoja de estilos XSLT en el archivo PXF de entrada.

- [help](#)⁴³: muestra información sobre el comando que se envía en el argumento (o sobre todos los comandos si no se envía ningún argumento).
- [licenseserver](#)⁴⁴: registra StyleVision Server con un LicenseServer de la red local.
- [pdfdata](#)⁴⁶: genera datos de formularios a partir de un archivo PDF en un archivo FDF o XML.
- [setdeflang](#)⁴⁸: establece el idioma predeterminado de StyleVision Server.
- [setfopath](#)⁴⁹: selecciona un procesador FO alternativo para generar archivos PDF.
- [verifylicense](#)⁵¹: comprueba si StyleVision Server tiene una licencia y si usa la clave de licencia dada.
- [version](#)⁵³: muestra el número de versión de StyleVision Server.

4.1 assignlicense (Windows only)

Sintaxis y descripción

El comando `assignlicense` está disponible en Windows solamente (no en sistemas Linux o Mac). Al ejecutarse carga el archivo de licencia indicado por el argumento `ARCHIVO` en el servidor LicenseServer con el que está registrado StyleVision Server (véase el comando `licenseserver`) y asigna esa licencia a StyleVision Server en este equipo. El argumento `ARCHIVO` toma la ruta de acceso del archivo de licencia. El comando también permite comprobar la validez de las licencias.

```
stylevisionserver assignlicense [opciones] FILE
```

- El argumento `FILE` toma la ruta al archivo de la licencia.
- La opción `--test-only` sirve para cargar la licencia a LicenseServer y validarla sin asignarla primero a StyleVision Server.

Para más información sobre el proceso de asignación de licencias consulte la documentación de Altova LicenseServer (<https://www.altova.com/manual/es/AltovaLicenseServer/>).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionserver` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, `"Mi archivo"`. Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, `"C:\Mi Directorio\"`) es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\"` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\\"`. En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: `"C:\Mi Directorio\\"`.

Ejemplos

Ejemplos del comando `assignlicense`:

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

- El primer ejemplo carga la licencia indicada a LicenseServer y se la asigna a StyleVision Server.
- El tercer comando carga la licencia indicada a LicenseServer y la valida, sin asignársela a StyleVision Server.

Opciones

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

▼ test-only [t]

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

Son valores permitidos: `true|false`. Si su valor es `true`, el archivo de licencia solamente se carga y valida en LicenseServer, pero no se asigna

4.2 exportresourcestrings

Sintaxis y descripción

El comando `exportresourcestrings` genera un archivo XML que contiene todas las cadenas de recursos de la aplicación StyleVision Server en el idioma indicado. Los idiomas en los que se puede generar el archivo son inglés (`en`), español (`es`), francés (`fr`), alemán (`de`) y japonés (`ja`).

```
stylevisionserver exportresourcestrings [opciones] LanguageCode XMLOutputFile
```

- El argumento `LanguageCode` indica el idioma de las cadenas de recursos del archivo XML de salida; se trata del *lenguaje de exportación*. Se admiten estos idiomas de exportación (se indica el código correspondiente entre paréntesis): inglés (`en`), alemán, (`de`), español (`es`), francés (`fr`) y japonés (`ja`).
- El argumento `XMLOutputFile` indica la ruta y el nombre del archivo XML de salida.

A continuación explicamos cómo localizar.

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionserver` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, "`Mi archivo`". Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, "`C:\Mi Directorio\`") es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\`". En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: "`C:\Mi Directorio\\`".

Ejemplos

Ejemplos del comando `exportresourcestrings`:

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

- El comando crea un archivo llamado `Strings.xml` en la carpeta `c:\`; ese archivo contiene las cadenas de los recursos de **StyleVision Server** en inglés.

Localizar StyleVision Server en otros idiomas

Si quiere puede localizar StyleVision Server en cualquier idioma. Altova ya ofrece la aplicación en cinco idiomas: inglés, español, francés, alemán y japonés (todos los archivos están en la carpeta `C:\Program`

Files (x86)\Altova\StyleVisionServer2021\bin) pero puede localizarla en cualquier otro idioma.

Siga estos pasos para localizar la aplicación:

1. Genere un archivo XML con las cadenas de recursos usando el comando `exportresourcestrings` (ver *más arriba*). Las cadenas de recursos de este archivo XML puede estar en uno de estos idiomas: inglés (en), español (es), francés (fr), alemán (de) o japonés (ja), dependiendo del argumento `códigoIdioma` que utilice con el comando.
2. Traduzca las cadenas de recursos al idioma de destino. Las cadenas de recursos son el contenido de los elementos `<string>` del archivo XML. No traduzca las variables que aparecen entre llaves, p. ej. `{option}` o `{product}`.
3. Póngase en contacto con [el equipo de soporte técnico de Altova](#), que le ayudarán a generar un archivo DLL localizado de StyleVision Server a partir de su archivo XML traducido.
4. Cuando reciba el archivo DLL localizado del equipo de [soporte técnico de Altova](#), guárdelo en la carpeta `C:\Archivos de programas (x86)\Altova\StyleVisionServer2021\bin`. El DLL tendrá un nombre similar a este `StyleVisionServer2021_ci.dll`. La parte `_ci` del nombre contiene el código del idioma. Por ejemplo, en `StyleVisionServer2021_de.dll`, la parte `de` es el código del idioma alemán (Deutsch).
1. Ejecute el comando `setdeflang` para establecer el archivo DLL localizado como aplicación StyleVision Server predeterminada. Use el código de idioma del nombre del archivo DLL como argumento del comando `setdeflang`

Nota: Altova ya ofrece StyleVision Server en estos cinco idiomas: inglés, español, francés, alemán y japonés. Para usar uno de estos idiomas como idioma predeterminado, use el comando `setdeflang` de StyleVision Server.

4.3 generate

Sintaxis y descripción

El comando `generate` (`gen` en versión abreviada) genera uno o más archivos de salida (HTML, PDF, RTF y/o DOCX) al transformar el archivo XML de entrada con los documentos XSLT que contiene el archivo PXF de entrada.

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

- La opción `--inputxml` es obligatoria y da la ruta de acceso del archivo XML.
- El argumento `InputPXF` indica la ruta de acceso al archivo PXF que contiene los documentos XSLT que se usan para generar los documentos de salida. Los archivos PXF se crean con la [aplicación de Altova StyleVision](#).
- Cada formato de salida se genera indicando la opción correspondiente para ese resultado (véase la *lista Opciones, más abajo*). El valor de cada una de las opciones es una ruta que indica dónde se debe generar el resultado.

Nota: StyleVision Server usa [Apache FOP](#), el procesador FO del Proyecto Apache, para generar archivos PDF a partir de FO. Apache FOP se instala automáticamente con StyleVision Server en estas ubicaciones: en sistemas Windows, `ProgramData\Altova\SharedBetweenVersions`; en sistemas Linux y macOS, en una carpeta bajo la carpeta `StyleVisionServer2021`. Para poder usar Apache FO necesita tener instalado **Java Runtime Environment 1.6** o posterior en el equipo de StyleVision Server. Debe instalar la versión de Java que corresponda a su versión de StyleVision Server (32 o 64 bits).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionserver` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, `"Mi archivo"`. Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, `"C:\Mi Directorio\"`) es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\"` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\\"`. En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: `"C:\Mi Directorio\\"`.

Ejemplos

Ejemplos del comando `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
```

- Los comandos anteriores contienen la opción obligatoria `--inputxml`, el argumento `--inputxml` (`Test.pxf`) y al menos una opción de creación de resultado (`--html` en todos los ejemplos anteriores).
- El archivo XML que se debe usar se encuentra dentro del archivo PXF (véanse *el segundo y el tercer ejemplo*) o puede ser un archivo XML externo (*ubicado fuera del archivo PXF; véanse los ejemplos primero y cuarto*).
- El conmutador `--inputxml` se ignora si la fuente principal del esquema es BD o XML-DB, pero sigue teniendo que estar presente por razones de sintaxis; debe usar algo parecido a: `--inputxml=database`.
- Si la opción de creación de resultado `--html` toma una ruta relativa, como en los ejemplos anteriores, entonces la ubicación del archivo de salida será relativa a la carpeta en la que se encuentra el archivo PXF.

Opciones

▼ inputxml [xml]

`--xml, --inputxml = PathToXMLFile`

Esta opción es obligatoria e indica la ruta de acceso del archivo XML que se quiere procesar. El archivo XML se puede encontrar dentro o fuera del archivo PXF. Para indicar archivos XML que estén dentro de un archivo PXF debe usar el localizador `|zip` (véase *la parte resaltada en los ejemplos anteriores*). La opción se ignora si la fuente principal del esquema del archivo PXF de entrada es una DB o DB-XML.

▼ dbwhere [dbw]

`--dbw, --dbwhere = WHEREClause`

Una cláusula SQL `WHERE` que determina qué filas de una fuente DB-XML de deben procesar.

▼ param [p]

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

Asigna un valor a un parámetro definido en el archivo PXF. El conmutador `--param` se debe usar antes de cada parámetro. Use comillas si `ParamName` o `ParamValue` contienen algún espacio. Ejemplo: `--p=$company:"Nanonull Inc"`

▼ prohibit-output-outside-target-folder

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

Usa los valores `true|false`. Con `true` no permite crear resultados en ninguna carpeta que no sea aquella en la que se crea el archivo principal de salida (HTML, PDF, RTF, DOCX). De esta forma se pueden proteger otras carpetas si hace falta. El valor predeterminado es `false`.

▼ outhtml [html]

`--html, --outhtml = FilePath`

La ruta de acceso del archivo HTML.

▼ outpdf [pdf]

```
--pdf, --outpdf = FilePath
```

La ruta de acceso al archivo PDF que se quiere generar.

▼ outrtf [rtf]

```
--rtf, --outrtf = FilePath
```

La ruta de acceso al archivo RTF que se quiere generar.

▼ outdocx [docx]

```
--docx, --outdocx = FilePath
```

La ruta de acceso al archivo DOCX que se quiere generar.

▼ outfo [fo]

```
--fo, --outfo = FilePath
```

La ruta de acceso al archivo FO que se quiere generar.

▼ generate-html-output-as-mime

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

Usa los valores `true|false`. Si no se indica esta opción, el valor predeterminado es `false`, si se indica pero no se le asigna ningún valor, entonces se lee como `true`. Si es `true`, entonces los resultados HTML se generan como una secuencia mime.

▼ verbose [v]

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

Usa los valores `true|false`. Activa o desactiva todos los mensajes que aparecen. El valor predeterminado es `false` si no se indica esta opción y `true` si se indica pero no se se asigna ningún valor.

▼ lang [l]

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

El idioma que se usa para mostrar mensajes.

Use la opción `--h, --help` para mostrar información sobre el comando.

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

4.4 help

Sintaxis y descripción

El comando `help` toma un único argumento (`Command`), que es el nombre del comando para el que necesita la ayuda, y muestra la sintaxis del comando, sus opciones y otra información relevante. Si no se especifica el comando `Command`, entonces se enumeran todos los comandos del ejecutable, cada uno con una breve descripción.

```
stylevisionserver help Command
```

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`StyleVisionServer` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

Ejemplo

Ejemplo del comando `help` para mostrar información sobre el comando `licenseserver`:

```
stylevisionserver help licenseserver
```

La opción `--help`

También puede acceder a la información de ayuda sobre un comando usando la opción `--help` del comando para el que necesita la ayuda. Los comandos siguientes obtienen el mismo resultado:

```
stylevisionserver licenseserver --help
```

El comando anterior usa la opción `--help` del comando `licenseserver`.

```
stylevisionserver help licenseserver
```

El comando `help` toma como argumento `licenseserver`.

En ambos casos, aparece información de ayuda sobre el comando `licenseserver`.

4.5 licenseserver

Sintaxis y descripción

Al ejecutarse, el comando `licenseserver` registra StyleVision Server con el servidor LicenseServer indicado por el argumento *Servidor-O-Dirección-IP*. Para que el comando `licenseserver` se ejecute correctamente, los dos servidores (StyleVision Server y LicenseServer) deben estar conectados en la red y LicenseServer debe estar en ejecución. Además debe tener privilegios de administrador para poder registrar StyleVision Server con LicenseServer.

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

- El argumento *Server-Or-IP-Address* toma el nombre o la dirección IP del equipo en el que se está ejecutando LicenseServer.
- Para solicitar una licencia de evaluación para StyleVision Server, indique los valores correspondientes para las opciones `--company`, `--email` y `--name`. Para solicitar una licencia de evaluación para otro producto de Altova (además de StyleVision Server) agregue la opción `--request-eval-license` y use como valor el nombre del producto para el que quiere la licencia. Consulte más abajo la lista de opciones para más información.

Cuando StyleVision Server se registre con LicenseServer, recibirá un mensaje de confirmación. El mensaje incluirá la URL del servidor LicenseServer. Ahora puede usar la URL para ir a LicenseServer y asignarle una licencia a StyleVision Server. Consulte la documentación de Altova LicenseServer para obtener más información (<https://www.altova.com/manual/es/AltovaLicenseServer/>).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionServer` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`stylevisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, `"Mi archivo"`. Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, `"C:\Mi Directorio\"`) es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\"` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\\"`. En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: `"C:\Mi Directorio\\"`.

Ejemplo

Ejemplo del comando `licenseserver`:

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

Estos comandos indican respectivamente que el equipo que ejecuta el servidor Altova LicenseServer es un equipo llamado `DOC.altova.com` y el equipo del usuario (`localhost` y `127.0.0.1`). En cada caso el comando registra StyleVision Server con el servidor LicenseServer del equipo correspondiente. El último comando indica al ejecutable del servidor que ejecute el comando.

Opciones

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

▼ json [j]

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

Usa los valores `true|false`. Si se establece en `true` imprime el resultado del intento de registro como objeto JSON analizable por máquina

4.6 pdfdata

Sintaxis y descripción

El comando `pdfdata` genera un archivo PDF o XML a partir del archivo PDF que se envía como argumento *InputPDF*.

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

- El argumento *InputPDF* indica la ruta de acceso al archivo PDF a partir del cual se genera el archivo FDF o XML de salida. Si el archivo PDF no contiene datos de formulario, tampoco los contiene el archivo generado.
- Use la opción `--outfdf` para indicar la ubicación del archivo PDF generado y la opción `--outxml` para indicar la del archivo XML.

Para más información sobre archivos FDF y formularios PDF para rellenar consulte la documentación de [Altova StyleVision \(Enterprise Edition\)](#).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

StyleVisionServer en Windows

stylevisionserver en Windows y Unix (Linux, Mac)

* Las minúsculas (*stylevisionserver*) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (*StyleVisionServer*) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, "*Mi archivo*". Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, "*C:\Mi Directorio*") es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\ "` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\ "`. En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: "*C:\Mi Directorio*".

Ejemplos

Ejemplos del comando `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
```

Estos ejemplos crean, respectivamente, un archivo FDF y un archivo XML a partir del mismo archivo PDF de entrada.

Opciones

▼ outfdf

`--outfdf = FilePath`

La ruta de acceso al archivo FDF generado.

▼ outxml

`--outxml = FilePath`

La ruta de acceso al archivo XML generado.

Use la opción `--h, --help` para mostrar información sobre el comando.

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

4.7 setdeflang

Sintaxis y descripción

El comando `setdeflang` (`sdl` en versión corta) establece el idioma predeterminado de StyleVision Server. Los idiomas disponibles son inglés (`en`), alemán (`de`), español (`es`), francés (`fr`) y japonés (`ja`). El comando toma el argumento obligatorio `LanguageCode`.

```
stylevisionserver setdeflang [opciones] LanguageCode
```

- El argumento `LanguageCode` es obligatorio y define el idioma predeterminado de StyleVision Server. Los valores correspondientes son: `en`, `de`, `es`, `fr`, `ja`.
- Use la opción `--h, --help` para mostrar información sobre el comando.

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`StyleVisionServer` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

Ejemplo

Ejemplos del comando `setdeflang` (`sdl`):

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

- El primer comando define el alemán como idioma predeterminado de StyleVision Server.
- El segundo comando define el español como idioma predeterminado de StyleVision Server.

Opciones

Use la opción `--h, --help` para mostrar información sobre el comando.

4.8 setfopath

Sintaxis y descripción

El comando `setfopath` (`sfp` en versión abreviada) indica la ruta de acceso a un procesador Apache FOP distinto al que viene incluido en el paquete de StyleVision Server.

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

- El procesador Apache FOP que viene incluido con StyleVision Server se usa por defecto para procesar los documentos FO y generar archivos PDF de salida. Si quiere usar una instancia de procesador Apache FOP distinta a la que viene integrada en StyleVision Server use el comando `setfopath` con el argumento `Path` e indique la ruta de acceso al procesador FO que quiere usar.
- Una vez haya indicado un procesador FO alternativo con el comando `setfopath`, ese será el procesador que se use al generar archivos PDF más adelante con el comando [generate](#)⁴⁰. Para volver a cambiar el procesador solo tiene que volver a usar el comando `setfopath`. Para que StyleVision Server vuelva a usar el procesador FOP debe ubicar la carpeta FOP en su sistema y usar esa ruta de acceso como argumento para `setfopath`.
- En sistemas Windows la carpeta FOP que se instala con StyleVision Server está en `ProgramData\Altova\SharedBetweenVersions`; en sistemas Linux y macOS está en una carpeta bajo la carpeta `StyleVisionServer2021`.

Para más información sobre archivos FDF y los formularios PDF para rellenar consulte la documentación de [Altova StyleVision \(Enterprise Edition\)](#).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionserver` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

▼ Barra diagonal inversa y espacios en sistemas Windows

En sistemas Windows: si hay espacios o caracteres especiales (por ejemplo en los nombres de archivos o de carpetas, empresas, personas o productos) debe usar comillas: por ejemplo, `"Mi archivo"`. Sin embargo, debe tener en cuenta que una barra diagonal inversa seguida de comillas dobles (por ejemplo, `"C:\Mi Directorio\"`) es posible que no se lea correctamente. Esto se debe a que la barra diagonal inversa también se usa para empezar secuencias de escape y la secuencia de escape `\"` también usa las comillas dobles. Para evitar secuencia de caracteres puede añadir otra barra diagonal inversa: `\\\"`. En resumen: si necesita escribir una ruta que contenga espacios y una barra diagonal inversa, esta es la mejor forma de hacerlo: `"C:\Mi Directorio\\"`.

Ejemplos

Una vez haya ejecutado el comando `setfopath` puede usar el comando [generate](#)⁴⁰ para generar archivos PDF usando el procesador FO que acaba de especificar.

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

Estos comandos hacen lo siguiente:

1. El comando `setfopath` indica que el procesador ubicado en `C:\FOP\FOP.bat` debe usarse para generar archivos PDF más adelante con el comando correspondiente.
2. El comando `generate` genera un archivo PDF a partir de los datos XML de entrada indicados, para lo que usa los archivos de transformación que contiene el archivo PXF. El procesador FO indicado en el comando anterior se usa para generar el PDF.

Opciones

Use la opción `--h, --help` para mostrar información sobre el comando.

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

4.9 verifylicense (Windows only)

Sintaxis y descripción

El comando `verifylicense` comprueba si un producto tiene licencia. La opción `--license-key` también permite comprobar si ya se ha asignado al producto una clave de licencia. Este comando solo funciona en Windows. No se puede usar en sistemas Linux ni Mac.

```
stylevisionserver verifylicense [opciones]
```

- Para comprobar si una licencia en concreto está asignada a StyleVision Server debe indicar la clave de licencia como valor de la opción `--license-key`.

Para más información sobre el proceso de asignación de licencias consulte la documentación de LicenseServer (<https://www.altova.com/manual/es/AltovaLicenseServer/>).

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`stylevisionserver` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`StyleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

Ejemplo

Ejemplo del comando `verifylicense`:

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

- El primer comando comprueba si StyleVision Server tiene licencia.
- El segundo comando comprueba si StyleVision Server usa la clave de licencia indicada en la opción `--license-key`.

Opciones

Las opciones se enumeran en versión corta (si existe) y larga. Puede usar una o dos barras tanto para la versión corta como para la larga. Las opciones pueden tomar un valor o no hacerlo. Si lo hacen, se escribe así: `--option=value`. Los valores se pueden indicar sin comillas, salvo en dos casos: (i) cuando la cadena del valor contiene espacios o (ii) cuando se indica de forma explícita en la descripción de la opción que las comillas son necesarias. Si una opción toma un valor booleano y no se especifica ningún valor, entonces el valor predeterminado de la opción es `TRUE`. Use la opción `--h, --help` para mostrar información sobre el comando.

▼ license-key [I]

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

Comprueba si StyleVision Server usa la clave de licencia indicada como valor de esta opción.

4.10 version

Sintaxis y descripción

El comando `version` muestra el número de versión de StyleVision Server.

```
stylevisionserver version
```

▼ Mayúsculas/minúsculas y barras en la línea de comandos

`styleVisionServer` en Windows

`stylevisionserver` en Windows y Unix (Linux, Mac)

* Las minúsculas (`stylevisionserver`) funcionan en todas las plataformas (Windows, Linux y Mac), mientras que las mayúsculas (`styleVisionServer`) funcionan solamente en Windows y Mac.

* Use la barra diagonal en Linux y Mac y la barra diagonal inversa en Windows.

Ejemplo

Ejemplo del comando `version`:

```
stylevisionserver version
```

5 API de StyleVision Server

StyleVision Server ofrece una interfaz de programación de aplicaciones (API) a la que puede acceder mediante programación desde su código .NET, Java o COM.

En esta sección de referencia encontrará:

- [Información sobre la interfaz .NET](#) ⁵⁵
- [Información sobre la interfaz COM](#) ⁵⁶
- [Información sobre la interfaz Java](#) ⁵⁷
- [Ejemplos de código](#) ⁵⁸
- [Una referencia de la API](#) ⁶⁷

5.1 Información sobre la interfaz .NET

La interfaz .NET está construida como contenedor alrededor de la interfaz COM. Se ofrece como ensamblado de interoperabilidad primario firmado por Altova y utiliza el espacio de nombres `Altova.StyleVisionServer`.

Durante el proceso de instalación, StyleVision Server se registra automáticamente como objeto de servidor COM, por lo que no es necesario registrarlo a mano. Si recibe un error de acceso, abra los Servicios de componentes y otorgue permiso a la misma cuenta que ejecuta el grupo de aplicaciones que incluye StyleVision Server.

Para poder usar StyleVision Server en su proyecto .NET basta con añadir una referencia al archivo `Altova.StyleVisionServer.dll` (ver instrucciones más abajo). El archivo `Altova.StyleVisionServer.dll` está en la carpeta `bin` de la carpeta de instalación de StyleVision Server. Este archivo DLL se añade automáticamente al caché global de ensamblados (GAC) durante la instalación de StyleVision Server (el GAC suele estar en la carpeta `C:\WINDOWS\assembly`).

Si StyleVision Server está registrado como objeto de servidor COM y el archivo `Altova.StyleVisionServer.dll` está a disposición de la interfaz .NET, las funciones de la API de StyleVision Server estarán disponibles para el proyecto .NET.

Para añadir una referencia al archivo DLL de StyleVision Server en un proyecto Visual Studio .NET:

1. Abra el proyecto .NET en Visual Studio y haga clic en **Proyecto | Agregar referencia**. Aparece el cuadro de diálogo "Agregar referencia".
2. Abra la pestaña *Examinar*, navegue hasta la carpeta *<carpeta de aplicación de StyleVisionServer>/bin*, seleccione el archivo `Altova.StyleVisionServer.dll` y haga clic en **Aceptar**.

Para ver la estructura del ensamblado `Altova.StyleVisionServer` utilice el Examinador de objetos de Visual Studio (haciendo clic en **Ver | Examinador de objetos**).

5.2 Información sobre la interfaz COM

StyleVision Server se registra automáticamente como objeto de servidor COM durante la instalación. Para comprobar si se registró correctamente, abra el Editor del Registro (p. ej. escriba el comando `regedit.exe` en la línea de comandos). Si el proceso de registro funcionó, el registro incluirá dos clases `StyleVision.Server` (suelen estar en `HKEY_LOCAL_MACHINE\SOFTWARE\Classes`).

Una vez registrado como objeto de servidor COM, podrá invocarlo desde otras aplicaciones y lenguajes de scripting que sean compatibles con llamadas COM. Si quiere cambiar la ubicación de la carpeta de instalación de StyleVision Server, lo mejor es desinstalar StyleVision Server y volver a instalarlo en la ubicación deseada. De esta manera el programa de instalación se encargará de los procesos de anulación de registro y registro automáticamente.

5.3 Interfaz sobre la interfaz Java

Para acceder a la API de StyleVision Server desde código Java debe añadir estas referencias al archivo `.classpath` del proyecto Java.

<code>StyleVisionServer.jar</code>	Biblioteca que se comunica con StyleVision Server
<code>StyleVisionServer_JavaDoc.zip</code>	Documentación de la API de StyleVision Server

Ambos archivos están en la carpeta `bin` de la carpeta de instalación de StyleVision Server. Puede hacer referencia a estos archivos desde su ubicación original o copiarlos en otra ubicación si lo prefiere.

Para añadir las referencias de la biblioteca de StyleVision Server en Eclipse:

En Eclipse puede añadir las referencias editando las propiedades del proyecto Java, tal y como se explica en las siguientes instrucciones (solo relevantes para Eclipse 4.4).

1. Abra el proyecto en Eclipse, haga clic en **Project | Properties** y después seleccione Java Build Path.
2. En la pestaña *Libraries* de bibliotecas haga clic en el botón **Add External JARs** para añadir archivos JAR externos y navegue hasta el archivo `StyleVisionServer.jar` (situado en la carpeta de instalación de StyleVision Server).
3. En el panel *JARs and class folders on the build path*, expanda el nodo `StyleVisionServer.jar` y después haga doble clic en el registro `Javadoc location: (None).ç`
4. Compruebe que están marcadas las opciones **Javadoc in archive** y **External file** y en el campo *Archive path* seleccione el archivo `StyleVisionServer_JavaDoc.zip` (situado en la carpeta de instalación de StyleVision Server).
5. Haga clic en **OK** para terminar. La referencia a la biblioteca de StyleVision Server y el archivo de almacenamiento Javadoc se añade ahora al archivo `.classpath` del proyecto.

5.4 Ejemplos de código

En este apartado encontrará ejemplos para estos lenguajes de programación:

- [C++](#)⁵⁸
- [C#](#)⁶⁰
- [VBScript](#)⁶¹
- [Visual Basic](#)⁶³
- [Java](#)⁶⁴

5.4.1 C++

Este ejemplo explica cómo usar código C++ para generar un archivo de salida RTF con un archivo PXF y un archivo XML de entrada.

Antes de empezar a probar el ejemplo, compruebe que cumple con estos requisitos:

- StyleVision Server está instalado y tiene asignada una licencia válida.
- StyleVision Server está disponible como objeto de servidor COM (este proceso suele tener lugar automáticamente durante la instalación de StyleVision Server. Consulte el apartado [Información sobre la interfaz COM](#)⁵⁶ para obtener más información).

```
// 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;
}

```

5.4.2 C#

Este ejemplo explica cómo usar código C# para generar un archivo de salida RTF con un archivo PXF y un archivo XML de entrada.

Antes de empezar a probar el ejemplo, compruebe que cumple con estos requisitos:

- StyleVision Server está instalado y tiene asignada una licencia válida.
- StyleVision Server está disponible como objeto de servidor COM (este proceso suele tener lugar automáticamente durante la instalación de StyleVision Server. Consulte el apartado [Información sobre la interfaz COM](#)⁵⁶ para obtener más información).

```

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";
            }
        }
    }
}

```



```

    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

Este ejemplo muestra cómo usar código Visual Basic para generar un archivo RTF de salida con un archivo PXF y un archivo XML de entrada.

Antes de empezar a probar el ejemplo, compruebe que cumple con estos requisitos:

- StyleVision Server está instalado y tiene asignada una licencia válida.
- Su proyecto de Visual Studio [incluye una referencia al DLL](#) ⁵⁵ de StyleVision Server.

```
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

End Module

```

5.4.5 Java

Este ejemplo muestra cómo usar código Java para generar un archivo RTF de salida con un archivo PXF y un archivo XML de entrada.

Antes de empezar a probar el ejemplo, compruebe que cumple con estos requisitos:

- StyleVision Server está instalado y tiene asignada una licencia válida.
- El archivo `.classpath` de su proyecto incluye las bibliotecas de StyleVision Server (consulte el apartado [Información sobre la interfaz Java](#) ⁵⁷ para obtener más información).

```
public class Program
{
    public static void main(String[] args)
    {
        try
        {
            //Cree un objeto de StyleVision Server
            com.altova.stylevisionserver.StyleVisionServer objSVS = new
com.altova.stylevisionserver.StyleVisionServer();

            //Defina el directorio de trabajo para los archivos de salida y los archivos
intermedios
            objSVS.setWorkingDirectory( "C:\\temp" );

            //La ruta predeterminada del ejecutable de StyleVision Server es la ruta de
acceso de instalación (el mismo directorio de StyleVisionServer.jar)
            //Si movió el archivo JAR de sitio, deberá definir explícitamente la ruta de
acceso del archivo .exe
            //objSVS.setServerPath( "C:\\Archivos de programa (x86)\\Altova\\
\\StyleVisionServer2021\\bin\\StyleVisionServer.exe" );

            //Prepare el nombre del archivo XML
            objSVS.setInputXML( "ExpReport.xml" );

            //Añada las rutas de acceso de salida (absolutas o relativas con respecto al
directorio de trabajo) para los archivos de salida
            objSVS.setOutputRTF( "ExpReport.rtf" );

            //Prepare los parámetros, si el diseño utiliza alguno
            //objSVS.AddParameter( "paramprueba1", "valor 1" );

            //Ejecute la transformación. El resultado se almacenará en C:\\temp\\ExpReport.rtf
            //NOTA: adapte la ruta de acceso del archivo de entrada para poder ejecutar el
ejemplo
            if ( objSVS.generate( "C:\\Archivos de programa (x86)\\Altova\\
\\StyleVisionServer2021\\etc\\Examples\\ExpReport.pxf" ) )
                System.out.println( "La ejecución finalizó correctamente" );
            else
                System.out.println( objSVS.getLastExecutionMessage() );
        }
        catch ( Exception e)
        {
            e.printStackTrace();
        }
    }
}
```


5.5 Referencia de la API

Este apartado ofrece una referencia del usuario de la API de StyleVision Server para estas plataformas:

- [COM y .NET](#)⁶⁷
- [Java](#)⁷⁴

5.5.1 COM y .NET

La API de StyleVision Server expone la **interfaz `IServer`**, que crea una instancia de objeto StyleVision Server nueva y ofrece acceso a StyleVision Server.

La interfaz `IServer` tiene los métodos y propiedades que aparecen a continuación.

Métodos

▼ AddParameter

Asigna un valor a un parámetro definido en el archivo 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

Borra la lista de parámetros.

▣ C#

```
void ClearParameterList()
```

▣ C++

```
HRESULT ClearParameterList();
```

▣ VB

```
Sub ClearParameterList()
```

▼ Generate

Genera archivos de salida (HTML, PDF, RTF o DOCX) con el archivo PXF indicado por `TransfPath`. Transforma el archivo XML de entrada (el archivo XML de trabajo del archivo PXF) con el documento XSLT que incluye el archivo PXF. Devuelve `TRUE` si la generación finaliza correctamente y `FALSE` si se produce un error.

▣ 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
```

Propiedades

▼ APIMajorVersion

Obtiene la versión principal de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property APIMajorVersion As Integer
```

▼ APIMinorVersion

Obtiene la versión secundaria de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property APIMinorVersion As Integer
```

▼ APIServicePackVersion

Obtiene la versión de service pack de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property APIServicePackVersion As Integer
```

▼ InputXML

Define la ruta de acceso y el nombre del archivo XML que se debe procesar (el archivo XML de trabajo del archivo PXF).

▣ C#

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

▣ C++

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

▣ VB

```
Property InputXML As String
```

▼ Is64Bit

Devuelve `TRUE` si el motor de StyleVision Server es un ejecutable de 64 bits.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property Is64Bit As Boolean
```

▼ LastExecutionMessage

Obtiene el mensaje recibido durante la última ejecución del comando `Generate`.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property LastExecutionMessage As String
```

▼ MajorVersion

Obtiene la versión principal de StyleVision Server.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property MajorVersion As Integer
```

▼ MinorVersion

Obtiene la versión secundaria de StyleVision Server.

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property MinorVersion As Integer
```

▼ OutputDOCX

Establece la ruta de acceso y el nombre del archivo DOCX de salida.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputDOCX As String
```

▼ OutputFO

Establece la ruta de acceso y el nombre del archivo FO de salida.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputFO As String
```

▼ OutputHTML

Establece la ruta de acceso y el nombre del archivo HTML de salida.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputHTML As String
```

▼ OutputPDF

Establece la ruta de acceso y el nombre del archivo PDF de salida.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputPDF As String
```

▼ OutputRTF

Establece la ruta de acceso y el nombre del archivo RTF de salida.

▣ C#

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

▣ C++

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

▣ VB

```
Property OutputRTF As String
```

▼ ProductName

Obtiene el nombre del producto: "StyleVision Server"

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property ProductName As String
```

▼ ProductNameAndVersion

Obtiene el nombre completo del producto, incluido el número de versión: "StyleVision Server 2014r2 sp1 (x64)".

▣ C#

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

▣ C++

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

▣ VB

```
ReadOnly Property ProductNameAndVersion As String
```

▼ ServerPath

Obtiene o establece la ruta de acceso del ejecutable de 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

Obtiene la versión de service pack de StyleVision Server (p. ej.: **1** si el nombre completo del producto es 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

Establece una cláusula `WHERE` de SQL que determina qué filas se deben procesar de la fuente de esquema XML de BD.

▣ C#

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

- ▣ C++

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

- ▣ VB

```
Property WhereClause As String
```

▼ WorkingDirectory

Obtiene o establece el directorio actual para los trabajos en ejecución. Las rutas de acceso relativas se evalúan con respecto al directorio de trabajo.

- ▣ 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

El paquete `com.altova.stylevisionserver` está compuesto por estas dos clases:

- `public class StyleVisionServer` (ver descripción más abajo)
- `public class StyleVisionServerException` extends [Exception](#)

clase `styleVisionServer`

La clase `StyleVisionServer` crea una instancia de objeto `StyleVision Server` nueva y ofrece acceso a `StyleVision Server`. Los métodos de la clase `StyleVisionServer` se describen a continuación.

Métodos de la clase `styleVisionServer`

A continuación se enumeran por orden alfabético todos los métodos de la clase `StyleVisionServer`.

▼ `addParameter`

```
public void addParameter(String nombre, String valor)
```

Añade el nombre y el valor de un parámetro nuevo. Cada parámetro y su valor se especifican en una llamada distinta al método. Los parámetros se deben declarar en el documento XSLT.

Parámetros:

`nombre`: almacena el nombre del parámetro en forma de cadena de texto

`valor`: almacena el valor del parámetro en forma de cadena de texto

▼ clearParameterList

```
public void clearParameterList()
```

Borra la lista de parámetros.

▼ generate

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

Procesa el archivo PXF indicado por `transfPath`. Emite `StyleVisionServerException`.

Parámetros:

`transfPath`: dirección URL absoluta que da la ubicación del archivo PXF.

Devuelve:

`true()` si la ejecución finaliza correctamente

`false()` si se produce un error de ejecución

Si se produce un error, utilice `getLastExecutionMessage()`

▼ getAPIMajorVersion

```
public int getAPIMajorVersion()
```

Obtiene la versión principal de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▼ getAPIMinorVersion

```
public int getAPIMinorVersion()
```

Obtiene la versión secundaria de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▼ getAPIServicePackVersion

```
public int getAPIServicePackVersion()
```

Obtiene la versión de service pack de la API de StyleVision Server. Puede ser distinta a la versión del producto si la API está conectada a otro servidor.

▼ getLastExecutionMessage

```
public String getLastExecutionMessage()
```

Obtiene el mensaje recibido durante la última ejecución del comando `generate`.

▼ getMajorVersion

```
public int getMajorVersion()
```

Obtiene la versión principal de la aplicación.

▼ getMinorVersion

```
public int getMinorVersion()
```

Obtiene la versión secundaria de la aplicación.

▼ getProductName

```
public String getProductName()
```

Obtiene el nombre del producto.

▼ getProductNameAndVersion

```
public String getProductAndVersion()
```

Obtiene el nombre completo del producto y su número de versión.

▼ getServicePackVersion

```
public int getServicePackVersion()
```

Obtiene la versión de service pack de la API de StyleVision Server.

▼ is64bit

```
public boolean is64bit()
```

Comprueba si el ejecutable es de 64 bits.

Devuelve:

true() para StyleVision Server (x64), de lo contrario devuelve false().

▼ setInputXML

```
public void setInputXML(String rutaAcceso)
```

Establece qué archivo XML se debe procesar. Debe ser la ruta de acceso del archivo XML de trabajo indicado en el archivo PXF.

Parámetros:

rutaAcceso: almacena la ruta de acceso del archivo XML de trabajo del archivo PXF.

▼ setOutputDOCX

```
public void setOutputDOCX(String rutaAcceso)
```

Establece la ruta de acceso y el nombre del archivo DOCX que se debe generar.

Parámetros:

rutaAcceso: ruta de acceso y nombre del archivo DOCX que se debe generar.

▼ setOutputFO

```
public void setOutputFO(String rutaAcceso)
```

Establece la ruta de acceso y el nombre del archivo FO que se debe generar.

Parámetros:

rutaAcceso: la ruta de acceso y el nombre del archivo FO que se debe generar.

▼ setOutputHTML

```
public void setOutputHTML(String rutaAcceso)
```

Establece la ruta de acceso y el nombre del archivo HTML que se debe generar.

Parámetros:

rutaAcceso: ruta de acceso y nombre del archivo HTML que se debe generar.

▼ setOutputPDF

```
public void setOutputPDF(String rutaAcceso)
```

Establece la ruta de acceso y el nombre del archivo PDF que se debe generar.

Parámetros:

rutaAcceso: ruta de acceso y nombre del archivo PDF que se debe generar.

▼ setOutputRTF

```
public void setOutputRTF(String rutaAcceso)
```

Establece la ruta de acceso y el nombre del archivo `RTF` que se debe generar.

Parámetros:

`rutaAcceso`: ruta de acceso y nombre del archivo `RTF` que se debe generar.

▼ **setServerPath**

```
public void setServerPath(String archivoServidor)
```

Establece la ruta de acceso del ejecutable de `StyleVisionServer`.

Parámetros:

`archivoServidor`: ruta de acceso del ejecutable de `StyleVisionServer`.

▼ **setWhereClause**

```
public void setWhereClause(String cláusulaWhere)
```

Establece una cláusula `WHERE` de SQL que determina qué filas se deben procesar de una fuente de esquema XML de BD.

Parámetros:

`cláusulaWhere`: la cláusula `WHERE` de SQL que determina qué filas se deben procesar de una fuente de esquema XML de BD.

▼ **setWorkingDirectory**

```
public void setWorkingDirectory(String directorioTrabajo)
```

Establece un directorio predeterminado. Las rutas de acceso relativas se resuelven en relación a este directorio.

Parámetros:

`directorioTrabajo`: ruta de acceso del directorio predeterminado (de trabajo).

6 Gestor de taxonomías XBRL

El Gestor de taxonomías XBRL es una herramienta que ofrece una forma centralizada de instalar y administrar taxonomías XBRL para usarlas en todas las aplicaciones de Altova compatibles con XBRL, incluido **StyleVision Server**. En Windows, el gestor tiene una interfaz gráfica del usuario a la que también puede acceder desde la línea de comandos. En Linux y Mac* esta herramienta solo está disponible en la línea de comandos.

* Esto solo aplica si está ejecutando el Gestor de taxonomías XBRL en Linux o macOS junto con las aplicaciones servidor de Altova que son multiplataforma, como son MapForce Server, StyleVision Server o RaptorXML+XBRL Server.

El Gestor de taxonomías XBRL permite:

- Ver las taxonomías XBRL que hay instaladas en su equipo y comprobar si hay versiones nuevas para descargar.
- Descargar las versiones más recientes de las taxonomías XBRL independientemente del ciclo de versiones de Altova. Altova guarda todas las taxonomías en un sistema de almacenamiento en línea al que tiene acceso el Gestor de taxonomías XBRL y desde donde puede descargarlas tan pronto como estén disponibles.
- Instalar o desinstalar cualquiera de las múltiples versiones de una taxonomía en concreto (o todas ellas, si las necesita).
- Una sola taxonomía XBRL representa un "paquete", pero puede tener dependencias en otras taxonomías. Al instalar o desinstalar una taxonomía, se detectan e instalan o desinstalan también automáticamente todas sus dependencias. La interfaz gráfica del usuario (o la línea de comandos, en su caso) le informa cuando se añaden o eliminan taxonomías.
- Las taxonomías XBRL administradas con el Gestor de taxonomías XBRL pueden usar el [catálogo XML](#), que permite resolver referencias a URI en documentos de instancia o esquema desde archivos locales, en vez de a través de Internet. Esta opción es muy importante en el caso de las taxonomías XBRL grandes, para las que no es práctico o no se recomienda resolver el esquema desde URIs remotos, principalmente por razones de rendimiento.

El Gestor de taxonomías XBRL permite administrar cualquiera de las taxonomías XBRL necesarias para usar en las [aplicaciones de Altova compatibles con XBRL](#)⁷⁹. Entre estas taxonomías se encuentran las del marco de informes de la Autoridad Bancaria Europea, las taxonomías de informes financieros US-GAAP y otras taxonomías específicas de otros países o ámbitos. Para ver la lista completa ejecute Gestor de taxonomías XBRL o el comando `list` de la línea de comandos.

Taxonomías XBRL personales

Si necesita trabajar con taxonomías XBRL que no estén incluidas en el Gestor de taxonomías XBRL, puede habilitarlas así:

- Desde la aplicación de escritorio de Altova compatible con XBRL correspondiente ejecute el comando de menú **Herramientas | Opciones** y después vaya a la página de opciones **XBRL | Paquetes de taxonomía**. Desde allí navegue hasta el paquete .zip de la taxonomía XBRL personal que quiera usar.
- Desde la aplicación servidor de Altova compatible con XBRL correspondiente proporcione las opciones `--taxonomy-package 0 --taxonomy-package-config-file` al ejecutar los comandos o métodos de API compatibles. Por ejemplo, estas opciones son compatibles con los comandos de validación XBRL como `valxbml` o `valxbmltaxonomy` en RaptorXML+XBRL Server, o el comando `run` (o su equivalente

en método API) en MapForce Server.

Aplicaciones de Altova compatibles con XBRL

Estas son las aplicaciones de Altova compatibles con XBRL, es decir, que pueden usar las funciones del Gestor de taxonomías XBRL:

- Complementos XBRL de Altova para Excel (EBA, Solvency II)
- MapForce Enterprise Edition
- MapForce Server
- MapForce Server Advanced Edition
- RaptorXML+XBRL Server
- StyleVision Server
- StyleVision Enterprise Edition
- XMLSpy Enterprise Edition

Los cambios realizados en las taxonomías con el Gestor de taxonomías XBRL afectan a todas las aplicaciones de la lista anterior si están instaladas en ese equipo.

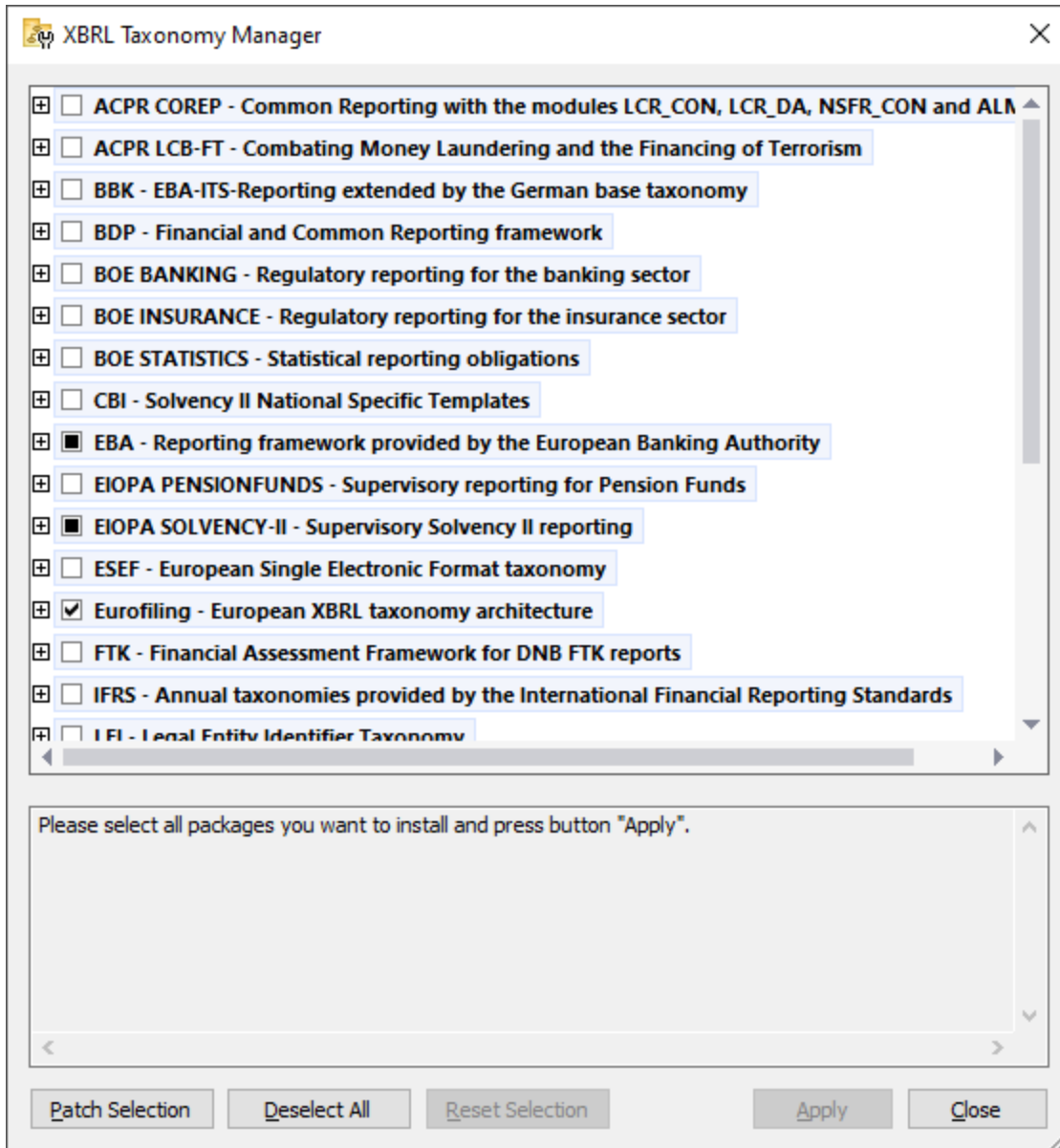
Instalación

Gestor de taxonomías XBRL se instala automáticamente al instalar cualquiera de las aplicaciones de Altova compatibles con XBRL o el **Altova Mission Kit Enterprise Edition**. También se elimina automáticamente si desinstala todas las aplicaciones de Altova compatibles con XBRL o el **Altova Mission Kit Enterprise Edition** del equipo.

Funcionamiento

Altova mantiene un almacenamiento en línea donde guarda todas las taxonomías XBRL de los productos de Altova. Este almacenamiento se actualiza de forma periódica, por ejemplo, poco después de que las organizaciones correspondientes publiquen las versiones nuevas de las taxonomías respectivas.

Al ejecutar Gestor de taxonomías XBRL desde la interfaz gráfica del usuario aparece información sobre las taxonomías más recientes disponibles en un cuadro de diálogo en el que puede visualizarlas, instalarlas, actualizarlas o desinstalarlas. También puede llevar a cabo las mismas acciones desde la línea de comandos.



También puede instalar las taxonomías ejecutando los archivos `.altova_taxonomies` descargados desde el sitio web de Altova (<https://www.altova.com/taxonomy-manager>). Asimismo, cuando ejecuta una acción que requiere cargar taxonomías XBRL, Gestor de taxonomías XBRL le pide que instale las taxonomías XBRL que faltan.

Independientemente de cómo se instalen las taxonomías, toda la información sobre las taxonomías instaladas se almacena en una ubicación centralizada de su equipo, el directorio caché. El directorio caché local también incluye información sobre los paquetes de Altova (que no incluye los archivos de las taxonomías, ya que estos se instalan a petición). El directorio caché local está en:

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

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

El directorio caché local se actualiza automáticamente de vez en cuando para que el estado más actual del equipo corresponda con el del almacenamiento en línea. Más concretamente, el caché se actualiza:

- Al ejecutar el Gestor de taxonomías XBRL.
- Al ejecutar StyleVision Server por primera vez en un mismo día natural.
- Si StyleVision Server ya se está ejecutando, el directorio caché se actualiza cada 24 horas.
- También puede actualizar el caché local desde el almacenamiento en línea manualmente ejecutando el comando de actualización desde la línea de comandos.

Si instala o desinstala taxonomías, el directorio caché local se actualiza automáticamente con información sobre las taxonomías disponibles e instaladas, además de con los propios archivos de taxonomía.

El directorio caché local se mantiene automáticamente en base a las taxonomías que instale o desinstale; no debe modificarlo ni eliminarlo manualmente. Si necesita restaurar el Gestor de taxonomías XBRL a su estado original, ejecute el comando `reset` desde la línea de comandos y después ejecute el comando `initialize`.

6.1 Ejecutar Gestor de taxonomías XBRL

Puede ejecutar Gestor de taxonomías XBRL de dos maneras:

Interfaz de la línea de comandos (solo Windows)

Si está ejecutando StyleVision Server en Windows siga estos pasos para ejecutar el Gestor de taxonomías XBRL:

- Ejecute (haga doble clic en) un archivo con la extensión `.altova_packages` que haya descargado del sitio web de Altova.
- Desde el panel de control de Windows haga clic con el botón derecho en la entrada **Gestor de taxonomías de Altova** y seleccione **Cambiar** o **Desinstalar** en el menú contextual.

También puede usar la casilla de verificación **Invocar al Gestor de taxonomías de Altova**, que encontrará en la última pantalla del asistente de instalación una vez haya completado la instalación de StyleVision Server.

Interfaz de la línea de comandos (Windows)

Para ejecutar el Gestor de taxonomías XBRL desde una interfaz de la línea de comandos:

1. Abra una ventana de la línea de comandos y cambie el directorio a **C:\ProgramData\Altova\SharedBetweenVersions**.
2. Para mostrar la ayuda en la línea de comandos ejecute:

```
TaxonomyManager.exe --help
```

Interfaz de la línea de comandos (Linux, macOS)

Para ejecutar el Gestor de taxonomías XBRL desde la línea de comandos:

1. Abra una ventana de la línea de comandos y cambie el directorio a **%INSTALLDIR%/bin**, donde **%INSTALLDIR%** es el directorio de instalación del programa.
2. Para mostrar la ayuda en la línea de comandos ejecute:

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

6.2 Instalar una taxonomía

Siga los pasos que se explican más abajo para instalar una taxonomía, según su plataforma.

Tenga en cuenta que:

- Si instala o desinstala una taxonomía desde el Gestor de taxonomías XBRL, esto afecta a todas las cuentas de usuario de ese equipo.
- Si instala o desinstala una taxonomía con el Gestor de taxonomías XBRL, esto afecta a todas las aplicaciones de Altova compatibles con XBRL que estén instaladas en el mismo equipo.
- Si la taxonomía que instala o desinstala tiene dependencias con otras taxonomías, estas también se instalan o desinstalan respectivamente.

Interfaz de la línea de comandos (solo Windows)

Para instalar una taxonomía:

1. Ejecute el Gestor de taxonomías XBRL.
2. Marque las casillas que hay junto a las taxonomías que quiere instalar y haga clic en **Aplicar**.

Otra opción, si descargó un archivo con la extensión **.altova_taxonomies** del sitio web de Altova (<https://www.altova.com/taxonomy-manager>), haga doble clic en el archivo **.altova_taxonomies** para ejecutarlo. Esto inicializará el Gestor de taxonomías XBRL.

Interfaz de la línea de comandos (Windows)

Para instalar una taxonomía ejecute:

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

Donde `FILTER` significa una de estas opciones:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: **eba-2.10, us-gaap-2020.0**. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list`.
2. Un archivo **.altova_taxonomies** descargado desde el sitio web de Altova.

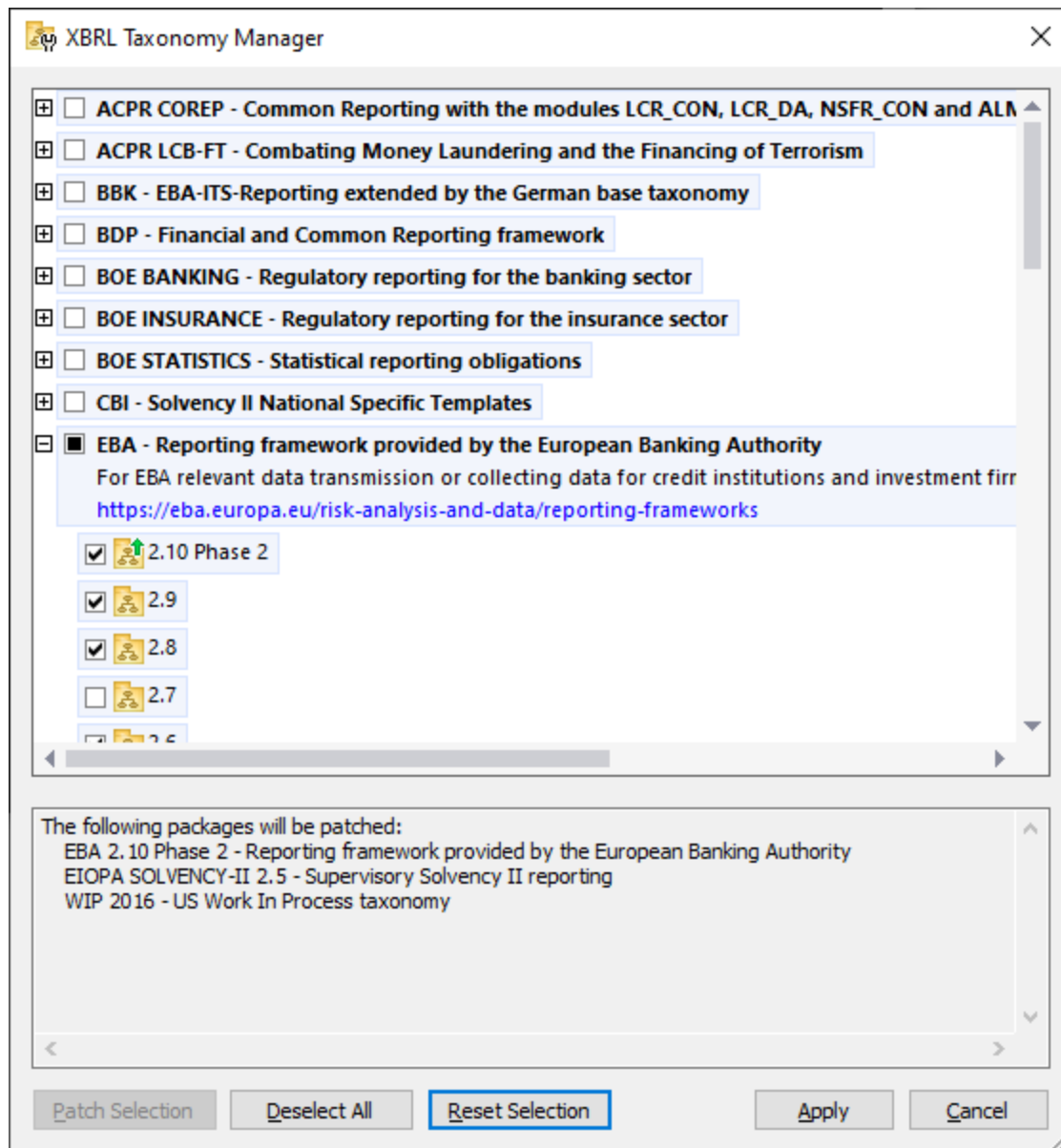
Interfaz de la línea de comandos (Linux, macOS)

Para instalar una taxonomía ejecute:

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

Donde `FILTER` significa una de estas opciones:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: **eba-2.10, us-gaap-2020.0**. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list`.
2. Un archivo **.altova_taxonomies** descargado desde el sitio web de Altova.



Nota: el botón **Seleccionar parche** se habilita solamente cuando hay parches disponibles para alguna de las taxonomías XBRL instaladas.

2. Haga clic en **Aplicar**.

Para aplicar un parche desde la línea de comandos:

1. Ejecute el comando `list -u`. Aparece una lista con las taxonomías para las que hay parches disponibles.
2. Ejecute el comando `upgrade` para instalar los parches.

Para más información consulte la referencia de la Interfaz de la línea de comandos.

6.3 Ver las taxonomías instaladas

Siga los pasos que se explican más abajo para ver las taxonomías instaladas, según su plataforma.

Interfaz de la línea de comandos (solo Windows)

Para ver todas las taxonomías instaladas desde una interfaz gráfica del usuario ejecute el Gestor de taxonomías XBRL. Si hay una casilla marcada junto a la taxonomía (o junto a esa versión) significa que esta está instalada.

Interfaz de la línea de comandos (Windows)

Para ver todas las taxonomías disponibles desde la línea de comandos ejecute:

```
TaxonomyManager.exe list
```

Para ver solamente las taxonomías instaladas ejecute:

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

Para ver solamente las taxonomías de las que hay una versión más reciente disponible, ejecute:

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

Interfaz de la línea de comandos (Linux, macOS)

Para ver todas las taxonomías disponibles ejecute:

```
sudo ./taxonomymanager list
```

Para ver solamente las taxonomías instaladas ejecute:

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

Para ver solamente las taxonomías de las que hay una versión más reciente disponible, ejecute:

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

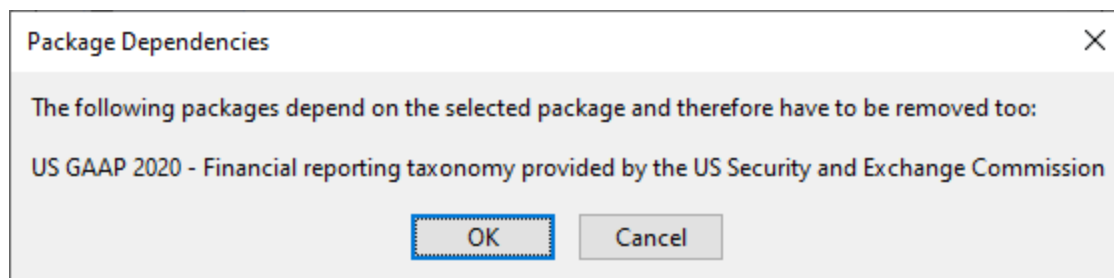
6.4 Desinstalar taxonomías

Siga los pasos que se explican más abajo para desinstalar una taxonomía, según su plataforma.

Interfaz de la línea de comandos (solo Windows)

Para desinstalar una taxonomía:

1. Ejecute el Gestor de taxonomías XBRL.
2. Desmarque las casillas que hay junto a las taxonomías que quiere desinstalar. Si la taxonomía seleccionada depende de otros paquetes de taxonomía se abre un cuadro de diálogo que le informa de que se eliminarán también todas esas dependencias:



3. Haga clic en **Aplicar** para eliminar la taxonomía y sus dependencias.

Interfaz de la línea de comandos (Windows)

Para desinstalar una taxonomía ejecute:

```
TaxonomyManager.exe uninstall PKG
```

Donde `FILTER` significa una de estas opciones:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: **eba-2.10**, **us-gaap-2020.0**. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list`.
2. Un archivo **.altova_taxonomies** descargado desde el sitio web de Altova.

Interfaz de la línea de comandos (Linux, macOS)

Para desinstalar una taxonomía ejecute:

```
sudo ./taxonomymanager uninstall PKG
```

Donde `FILTER` significa una de estas opciones:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: **eba-2.10**, **us-gaap-2020.0**. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list`.
2. Un archivo **.altova_taxonomies** descargado desde el sitio web de Altova.

6.5 Interfaz de la línea de comandos

Para llamar a Gestor de taxonomías XBRL desde la línea de comandos necesita saber la ruta del ejecutable. Por defecto, el ejecutable del Gestor de taxonomías XBRL se encuentra en:

<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

* Las rutas de Linux y macOS se aplican solo si está ejecutando el Gestor de taxonomías XBRL en los sistemas operativos correspondientes junto con las aplicaciones servidor multiplataforma de Altova, como MapForce Server, StyleVision Server o RaptorXML+XBRL Server.

Normalmente en esta documentación no se incluye la ruta completa del ejecutable al describir un comando y se usa **<exec>** en vez del nombre del ejecutable, por ejemplo:

```
<exec> help
```

Donde **<exec>** es la ruta o el nombre del ejecutable.

Llamar al Gestor de taxonomías XBRL desde el directorio de instalación

Para llamar al ejecutable sin tener que teclear toda la ruta puede cambiar el directorio a uno de estos:

<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"

Ahora puede llamar al ejecutable desde una ruta relativa para ejecutar un comando, por ejemplo:

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

Nota: en los sistemas Linux y macOS el prefijo `./` indica que el ejecutable está en el directorio actual. El prefijo `sudo` indica que el comando se debe ejecutar con derechos de administrador.

Llamar al Gestor de taxonomías XBRL desde cualquier directorio

Para llamar al ejecutable desde cualquier directorio haga referencia a este con la ruta absoluta. Si quiere teclear solamente el nombre del ejecutable para llamar al programa puede editar la variable de entorno de su

sistema operativo PATH para que incluya la ruta completa al directorio del ejecutable. Para ver cómo cambiar la variable de entorno PATH consulte la documentación de su sistema operativo.

Notas:

- Después de cambiar la variable de entorno PATH puede que necesite cerrar la ventana terminal y abrir una nueva para que los cambios surtan efecto.
- En Linux y macOS si `sudo` no se tiene en cuenta la variable PATH del usuario.

Sintaxis de la línea de comandos

La sintaxis general para usar la línea de comandos es:

```
<exec> --h | --help | --version | <command> [opciones] [argumentos]
```

En el código anterior la barra vertical | separa elementos que se excluyen mutuamente. Los corchetes [] indican elementos opcionales. Básicamente, puede teclear la ruta del ejecutable seguida por las opciones `--h`, `--help` o `--version`, o por un comando. Cada comando puede tener opciones y argumentos. Los comandos se describen en los apartados siguientes.

6.5.1 help

Este comando ofrece ayuda contextual sobre los comandos del ejecutable de Gestor de taxonomías XBRL.

Sintaxis

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

Donde [command] es un argumento opcional que indica cualquier nombre válido de comando.

Observaciones

También puede invocar la ayuda tecleando un comando seguido por `--h` or `--help`, por ejemplo:

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

O puede invocar la ayuda general tecleando `--h` o `--help` directamente después del ejecutable, por ejemplo:

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

Ejemplo

Este comando muestra la ayuda del comando `list`:

```
Linux, macOS ./taxonomymanager help list
```

<i>Windows</i>	<code>TaxonomyManager.exe help list</code>
----------------	--

6.5.2 info

Este comando muestra información detallada sobre cada una de las taxonomías dadas como argumento. Esta información incluye el título, la versión, la descripción, el editor y las referencias de las dependencias.

Sintaxis

<code><exec> info [options] FILTER...</code>
--

Para indicar varias taxonomías repita `FILTER` tantas veces como sea necesario.

Argumentos

PKG	<p>Donde <code>FILTER</code> significa una de estas opciones:</p> <ol style="list-style-type: none"> 1. Un identificador de taxonomía en el formato <code><name>-<version></code>, por ejemplo: eba-2.10, us-gaap-2020.0. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando <code>list</code>. 2. Un archivo .altova_taxonomies descargado desde el sitio web de Altova.
-----	---

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
--------------------------	--

Ejemplo

Este comando muestra información detallada sobre las taxonomías **eba-20.12.0** y **us-gaap-20.04.0**:

<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

Este comando inicializa el entorno del Gestor de taxonomías XBRL y crea un directorio caché donde se guardan todas las taxonomías localmente. El Gestor de taxonomías XBRL se inicializa automáticamente la primera vez que instale una aplicación de Altova compatible con él, por lo que normalmente no es necesario ejecutar este comando. Por lo general solo es necesario ejecutarlo después de haber ejecutado el comando `reset`.

Sintaxis

```
<exec> initialize [opciones]
```

El alias de este comando es `init`.

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

6.5.4 install

Este comando instala una o más taxonomías. Tenga en cuenta que:

- Si instala o desinstala una taxonomía desde el Gestor de taxonomías XBRL, esto afecta a todas las cuentas de usuario de ese equipo.
- Si instala o desinstala una taxonomía con el Gestor de taxonomías XBRL, esto afecta a todas las aplicaciones de Altova compatibles con XBRL que estén instaladas en el mismo equipo.
- Si la taxonomía que instala o desinstala tiene dependencias con otras taxonomías, estas también se instalan o desinstalan respectivamente.

Sintaxis

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

Para indicar varias taxonomías repita `FILTER` tantas veces como necesite.

Argumentos

PKG	<p>Donde <code>FILTER</code> significa una de estas opciones:</p> <ol style="list-style-type: none"> 1. Un identificador de taxonomía en el formato <code><name>--<version></code>, por ejemplo: eba-2.10, us-gaap-2020.0. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando <code>list</code>. 2. Un archivo .altova_taxonomies descargado desde el sitio web de Altova.
-----	--

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
--------------------------	--

<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

Ejemplo

Este comando instala las taxonomías **eba** (Autoridad Bancaria Europea) y **us-gaap** (principios de contabilidad generalmente aceptados en EE UU):

<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

Use este comando para ver las taxonomías en la línea de comandos; tiene varias opciones:

- lista de todas las taxonomías disponibles
- lista de taxonomías específicas
- lista de las taxonomías instaladas
- lista de las taxonomías que se pueden actualizar.

Sintaxis

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

Este comando se puede abreviar con `ls`.

Argumentos

<code>FILTER</code>	Enumera solo las taxonomías que contengan esta cadena en su nombre. Puede usarse este argumento varias veces.
---------------------	---

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--installed, --i</code>	Muestra solamente las taxonomías instaladas. El valor predeterminado es false .
<code>--upgradeable, --u</code>	Muestra solamente las taxonomías para las que hay disponible una versión más reciente. El valor predeterminado es false .

Ejemplos

Para ver todas las taxonomías disponibles ejecute:

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

Para ver solamente las taxonomías instaladas ejecute:

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

Para ver todas las taxonomías cuyos nombres contienen "eba" o "us-gaap" ejecute:

<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

Este comando elimina todas las taxonomías instaladas, así como el directorio de caché.

Advertencia: este comando elimina todas las taxonomías instaladas y su información.

Una vez haya ejecutado este comando, asegúrese de que ejecuta el comando `initialize` para volver a crear el directorio de caché. También puede ejecutar el comando `reset` con la opción `-i`.

Tenga en cuenta que `reset -i` restaura la instalación original del producto, por lo que se recomienda ejecutar también el comando `update` después de una restauración. También puede ejecutar el comando `reset` con las opciones `-i` y `-u`.

Sintaxis

```
<exec> reset [opciones]
```

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--init, --i</code>	Inicializa el entorno del Gestor de taxonomías XBRL después de una restauración. Son valores válidos true y false . El valor predeterminado es false .
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .

<code>--update, --u</code>	Inicializa y actualiza el entorno del Gestor de taxonomías XBRL después de una restauración. Son valores válidos true y false . El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

Ejemplos

Para restaurar el Gestor de taxonomías XBRL ejecute:

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

6.5.7 setdeflang

Este comando define el idioma del Gestor de taxonomías XBRL.

Sintaxis

```
<exec> setdeflang language
```

Donde `language` es un argumento obligatorio que indica el código del idioma. El alias de este comando es `sdl`.

Argumentos

<code>language</code>	El idioma que se quiere usar. Son valores válidos inglés (<code>en</code>), francés (<code>fr</code>), alemán (<code>de</code>), japonés (<code>ja</code>) y español (<code>es</code>).
-----------------------	---

Opciones

<code>--help, --h</code>	Muestra ayuda sobre este comando en la línea de comandos.
--------------------------	---

Ejemplos

Para definir el idioma como español ejecute:

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

6.5.8 uninstall

Este comando desinstala una o más taxonomías. Por defecto, cualquier taxonomía a la que haga referencia la taxonomía actual también se desinstala. Para desinstalar solamente la taxonomía actual y mantener aquellas a las que se hace referencia, use la opción `--k`.

Sintaxis

```
<exec> uninstall FILTER...
```

Para indicar varias taxonomías, repita `FILTER` tantas veces como sea necesario.

Argumentos

PKG	<p>Donde <code>FILTER</code> significa una de estas opciones:</p> <ol style="list-style-type: none"> 1. Un identificador de taxonomía en el formato <code><name>-<version></code>, por ejemplo: eba-2.10, us-gaap-2020.0. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando <code>list</code>. 2. Un archivo .altova_taxonomies descargado desde el sitio web de Altova.
-----	---

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--keep-references, --k</code>	Si usa esta opción, las taxonomías referenciadas no se desinstalan. El valor predeterminado es false .
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

Ejemplo

Este comando desinstala las taxonomías **eba-20.12.0** y **us-gaap-20.04.0**:

<i>Linux, macOS</i>	<code>./taxonomymanager uninstall eba-20.12.0 us-gaap-20.04.0</code>
<i>Windows</i>	<code>TaxonomyManager.exe uninstall eba-20.12.0 us-gaap-20.04.0</code>

6.5.9 update

Este comando consulta la lista de taxonomías disponibles en el almacenamiento en línea y actualiza el directorio de caché local. Esta información se actualiza de forma implícita, por lo que no es necesario ejecutar

este comando a no ser que haya ejecutado `reset` e `initialize`.

Sintaxis

```
<exec> update [opciones] [PATH]
```

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

Ejemplo

Este comando actualiza la lista de taxonomías.

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

6.5.10 upgrade

Este comando actualiza todas las taxonomías aptas a la versión *parche* más reciente disponible. En otras palabras, solamente actualiza a nivel de los parches de un release específico. Se recomienda ejecutar este comando solo si hay taxonomías por actualizar. Puede identificar cuáles lo son con el comando `list -u`.

Sintaxis

```
<exec> upgrade [opciones]
```

Opciones

<code>--help, --h</code>	Muestra la ayuda sobre este comando en la línea de comandos.
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es false .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es false .

Índice

A

Asignación de licencias en Linux, 23
Asignación de licencias en macOS, 29
Asignación de licencias en Windows, 16

E

Ejecución por subprocesos simples, 12

F

FlowForce Server,
y StyleVision Server, 9

G

Gestor de taxonomías XBRL, 78

I

Instalación,

Linux, 20
macOS, 27
Windows, 13

Instalación en Linux, 20
Instalación en macOS, 27
Instalación en Windows, 13

L

Linux,

asignación de licencias en, 23
instalación en, 20

M

macOS,

asignación de licencias en, 29
instalación en, 27

N

Núcleos, 12
Núcleos de procesador, 12
Núcleos de procesador y licencias, 12

S

StyleVision Server,

características, 8
como módulo de FlowForce Server, 9
funciones, 8

W

Windows,

asignación de licencias en, 16
instalación en, 13