

Altova MapForce Server 2024 Advanced Edition



Manual del usuario y referencia

Altova MapForce Server 2024 Advanced Edition

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: 2024

© 2018-2024 Altova GmbH

Contenido

1	Introducción	6
2	Instalación y asignación de licencias	7
2.1	Instalación y configuración en Windows.....	8
2.1.1	Instalar en Windows.....	8
2.1.2	Instalar en Windows Server Core.....	9
2.1.3	Instalar LicenseServer (Windows).....	11
2.1.4	Asignar licencias a MapForce Server (Windows).....	12
2.2	Instalación y configuración en Linux.....	16
2.2.1	Instalar en Linux.....	16
2.2.2	Instalar LicenseServer (Linux).....	18
2.2.3	Asignar licencias a MapForce Server (Linux).....	19
2.3	Instalación y configuración en macOS.....	22
2.3.1	Instalar en macOS.....	22
2.3.2	Instalar LicenseServer (macOS).....	23
2.3.3	Asignar licencias a MapForce Server (macOS).....	24
2.4	Actualizar MapForce Server.....	27
2.5	Migrar MapForce Server a un equipo nuevo.....	28
3	Ejecutar asignaciones	29
3.1	Preparar asignaciones para ejecución en servidor.....	32
3.2	Recursos globales.....	38
3.3	Optimización por combinación.....	40
3.4	Credenciales.....	43
3.5	Autenticación dinámica.....	46
4	Línea de comandos de MapForce Server	47

4.1	assignlicense.....	50
4.2	exportresourcestrings.....	51
4.3	help	53
4.4	licenseserver.....	54
4.5	run	55
4.6	setdeflang.....	59
4.7	verifylicense.....	60
4.8	version.....	61

5 API de MapForce Server 62

5.1	Interfaz .NET.....	63
5.1.1	Ejemplo en C#.....	64
5.1.2	Ejemplo en Visual Basic .NET.....	67
5.2	Interfaz COM.....	71
5.2.1	Ejemplo en C++.....	71
5.2.2	Ejemplo en VBScript.....	74
5.2.3	Ejemplo en Visual Basic for Applications (VBA).....	77
5.3	Interfaz Java.....	80
5.3.1	Ejemplo en Java.....	84
5.4	Ejemplo: ejecutar asignaciones con parámetros.....	87
5.5	Referencia de la API (.NET y COM).....	93
5.5.1	Interfaces.....	93
5.6	Referencia de la API (Java).....	111
5.6.1	Clases.....	111

6 Gestión de certificados digitales 125

6.1	Confiar en certificados servidor en Linux.....	128
6.2	Confiar en certificados servidor en OS X / macOS.....	129
6.3	Confiar en certificados servidor en Windows.....	130
6.4	Acceso a los almacenes de certificados en Windows.....	131
6.5	Exportar certificados de Windows.....	132
6.6	Certificados de cliente en Linux.....	138
6.7	Certificados de cliente en OS X / macOS.....	140

6.8	Certificados de cliente en Windows.....	142
7	Gestor de taxonomías	144
7.1	Ejecutar el gestor de taxonomías.....	148
7.2	Categorías de estado.....	151
7.3	Aplicar parches o instalar una taxonomía.....	153
7.4	Desinstalar o restaurar taxonomías.....	155
7.5	Interfaz de la línea de comandos (ILC).....	156
7.5.1	help	156
7.5.2	info	157
7.5.3	initialize.....	157
7.5.4	install	158
7.5.5	list	158
7.5.6	reset	159
7.5.7	uninstall.....	160
7.5.8	update.....	161
7.5.9	upgrade.....	161
8	Archivos de catálogo	163
	Índice	164

1 Introducción

MapForce Server es una solución de software comercial que ejecuta transformaciones de asignaciones de datos en sistemas operativos Windows, Linux y macOS. Las asignaciones de datos (o archivos *.mfd) se diseñan con Altova MapForce (<http://www.altova.com/es/mapforce.html>), donde puede definir las entradas y salidas y otros pasos de procesamiento intermedios que se deban aplicar a los datos. La función de MapForce Server es ejecutar archivos de ejecución de MapForce Server (.mfx) compilados con Altova MapForce y producir archivos o datos de salida o incluso actualizar bases de datos o llamar a servicios web, en base al diseño de la asignación subyacente.



MapForce Server puede ejecutarse por separado o bajo el control de Altova FlowForce Server (<http://www.altova.com/es/flowforce.html>). Si se instala en el mismo equipo que MapForce Server, FlowForce Server automatiza la ejecución de asignaciones mediante trabajos programados o basados en eventos, que también se pueden exponer como servicios web. Además, FlowForce Server viene con una biblioteca integrada de funciones que permiten llevar a cabo otras acciones automatizadas antes o después de ejecutar la asignación (p. ej. enviar un correo electrónico, copiar archivos y directorios, cargar archivos a una FTP, ejecutar comandos de shell, etc.).

Características

- Rendimiento nivel servidor durante la ejecución de asignaciones de datos
- Compatibilidad multiplataforma: MapForce Server funciona con Windows, Linux y macOS
- Interfaz de la línea de comandos
- API que se puede llamar desde código C++, C#, Java, VB.NET, VBScript o VBA
- Integración nativa con Altova FlowForce Server
- Compatibilidad con la característica de recursos globales de Altova, que permite hacer referencia a archivos, carpetas y bases de datos y poder reutilizar este tipo de recursos, configurarlos y ponerlos a disposición de otras aplicaciones de Altova (véase [Recursos globales de Altova](#)³⁸)
- Optimización por combinación (véase [Optimización por combinación](#)⁴⁰)
- Ejecución de asignaciones que aplican funciones y valores predeterminados a varios elementos al mismo tiempo. Estas asignaciones permiten, por ejemplo, reemplazar de forma sencilla todos los valores null que se encuentren con cadenas de texto vacías o con texto personalizado
- Ejecución de asignaciones que leen datos de y escriben datos en formato binario Protocol Buffers
- Ejecución de asignaciones que hacen inserciones masivas en bases de datos

Limitaciones

- Las firmas XML digitales no son compatibles con MapForce Server.
- Las conexiones de base de datos ADO, ADO.NET y ODBC solo son compatibles con Windows. En Linux y macOS se dispone de conectividad de base de datos nativa para SQLite y PostgreSQL. Para el resto de bases de datos en Linux o macOS deberá utilizar JDBC.

Última actualización: 09.04.2024

2 Instalación y asignación de licencias

En esta sección se explica cómo instalar su producto, asignarle una licencia y configurarlo. La sección se divide en varios apartados:

- [Instalación en Windows](#) ⁸
- [Instalación en Linux](#) ¹⁶
- [Instalación en macOS](#) ²²
- [Actualizar MapForce Server](#) ²⁷
- [Migrar MapForce Server a un equipo nuevo](#) ²⁸

2.1 Instalación y configuración en Windows

Esta sección explica cómo [instalar](#)⁸ MapForce Server y asignarle [licencias](#)¹² en sistemas Windows.

Requisitos del sistema (Windows)

Tenga en cuenta estos requisitos del sistema:

- Windows 10, Windows 11
- Windows Server 2016 o superior

Requisitos previos

Tenga en cuenta los siguientes requisitos:

- La instalación debe realizarse como usuario con privilegios de administrador.
- A partir de la versión 2021, la versión de 32 bits de MapForce Server no se puede instalar sobre una versión de 64 bits o viceversa. Debe (i) eliminar la versión anterior antes de instalar la nueva o (ii) actualizar la versión a una más reciente de la misma versión de bits que la que ya tenía instalada.

2.1.1 Instalar en Windows

MapForce Server se puede instalar en sistemas Windows. A continuación explicamos cómo instalar y configurar su producto. Para información detallada sobre partes específicas del proceso de instalación consulte los apartados correspondientes.

Instalar MapForce Server

Para instalar MapForce Server en sistemas Windows siga estos pasos:

- Como producto servidor independiente. Para instalar MapForce Server descargue y ejecute su instalador. Siga las instrucciones que aparecen en pantalla.
- Para instalar MapForce Server como parte del paquete de [FlowForce](#) Server descargue e instale el instalador de FlowForce Server. Siga las instrucciones que aparecen en pantalla y asegúrese de que marca la casilla para que también se instale MapForce Server.

Puede encontrar tanto el instalador de MapForce Server como el de [FlowForce Server](#) en el Centro de descargas de Altova (<http://www.altova.com/download.html>). Puede seleccionar el idioma de instalación en la parte inferior izquierda del asistente de instalación. Tenga en cuenta que el idioma que elija en la instalación también será el idioma predeterminado de MapForce Server. Puede cambiar el idioma más tarde desde la línea de comandos.

Instalar LicenseServer

Para poder utilizar MapForce Server debe asignarle una registros y licencia con un [Altova LicenseServer](#) de su red. En sistemas Windows puede instalar LicenseServer al mismo tiempo que MapForce Server o FlowForce Server. Para más detalles consulte [Instalar LicenseServer](#)¹¹.

Después de la instalación encontrará el ejecutable de MapForce Server en esta carpeta:


```
<CarpetaArchivosPrograma>\Altova\MapForceServer2024\bin\MapForceServer.exe
```

El instalador se encarga de realizar los registros necesarios para usar MapForce Server a través de una interfaz COM, como interfaz Java y en el entorno .NET.

Instalación en Windows Server Core

Windows Server Core no tiene ninguna IGU y debe instalarse a través de la línea de comandos. Para más información sobre cómo instalar el equipo, consulte el apartado [Instalar en Windows Server Core](#) ⁹.

Desinstalar MapForce Server

Para desinstalar MapForce Server siga estos pasos:

1. Haga clic con el botón derecho en el botón de **Inicio** de Windows y seleccione **Opciones**.
2. Abra el Panel de control (empiece a escribir "Panel de control" y haga clic en la entrada sugerida).
3. En *Programas* haga clic en **Desinstalar un programa**.
4. En el Panel de control seleccione MapForce Server y haga clic en **Desinstalar**.

Licencia de evaluación

Durante el proceso de instalación podrá solicitar una licencia de evaluación para MapForce Server que dura 30 días. Una vez enviada la solicitud recibirá una licencia de evaluación en la dirección de correo electrónico que indicó al registrarse.

2.1.2 Instalar en Windows Server Core

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

1. Descargue el instalador de MapForce Server desde el Centro de descargas de Altova. Este archivo se llama **MapForceServerAdv.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 **MapForceServerAdv.exe /u**. Este comando abre los archivos **.msi** en el mismo directorio que el ejecutable del instalador.
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 MapForce Server cierre MapForce Server antes de ejecutar el paso siguiente.
5. Use el archivo **.msi** para la instalación; para ello ejecute el comando **msiexec /i MapForceServerAdvanced.msi**. Este comando inicia la instalación en Windows Server Core.

Importante: conserve el archivo MSI

Debe tener en cuenta que:

- Mantenga los archivos **.msi** extraídos en un lugar seguro. Lo necesitará más tarde para desinstalar, reparar o modificar su instalación.
- Si quiere cambiar el nombre del archivo MSI debe hacerlo antes de instalar MapForce Server.

- El nombre del archivo MSI se almacena en el registro, que es donde puede actualizar el nombre si este ha cambiado.

Registrar MapForce Server con LicenseServer

Si instala MapForce Server por primera vez o si está actualizando el programa a la **versión siguiente**, necesita registrar MapForce Server con Altova LicenseServer en su red. Si está instalando una actualización menor de MapForce Server, el instalador reconocerá que ya registró el programa con LicenseServer, por lo que no será necesario volver a registrarlo. Sin embargo, si quiere cambiar el LicenseServer que usa MapForce Server en algún momento, debe registrar MapForce Server con License Server.

Para registrar MapForce Server con Altova LicenseServer después de la instalación ejecute el comando de instalación con la propiedad `REGISTER_WITH_LICENSE_SERVER`, como se ve a continuación, e indique el nombre o la dirección del LicenseServer correspondiente como valor de la propiedad, por ejemplo:

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

Para registrar MapForce Server con un Altova LicenseServer después de instalarlo ejecute este comando:

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

Comandos útiles

A continuación encontrará varios comandos que pueden serle útiles en el contexto de la instalación.

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 MapForceServerAdvanced.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 MapForceServerAdvanced.msi /q /L*v! <pathToInstallLogFile>
```

Para modificar la instalación, ejecute:

```
msiexec /m MapForceServerAdvanced.msi
```

Para reparar la instalación, ejecute:

```
msiexec /r MapForceServerAdvanced.msi
```

Para desinstalar MapForce Server:

```
msiexec /x MapForceServerAdvanced.msi
```

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

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

Para instalar MapForce Server en otro idioma (los idiomas disponibles y sus códigos son: Alemán=de; Español=es; Francés=fr):

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

Nota: En Windows Server Core no están disponibles los gráficos de MapForce Server.

Nota: Para instalar taxonomías use el Gestor de taxonomías desde la línea de comandos. Para más información consulte el Manual del usuario de MapForce Server.

2.1.3 Instalar LicenseServer (Windows)

Para poder utilizar MapForce Server debe asignarle una licencia con un [Altova LicenseServer](#) de su red. En sistemas Windows puede instalar LicenseServer al mismo tiempo que MapForce Server o FlowForce Server. Si ya hay un LicenseServer instalado en su red no necesita instalar uno nuevo, a no ser que necesite actualizar la versión de LicenseServer. (Véase el punto siguiente, [versiones de LicenseServer](#).)

Durante el proceso de instalación de MapForce Server or FlowForce Server, active o desactive la opción para instalar LicenseServer según corresponda. Tenga en cuenta que:

- Si no ha instalado Altova LicenseServer, no cambie las opciones seleccionadas por defecto. El asistente instalará la versión más reciente en el equipo donde se está ejecutando.
- Si 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 LicenseServer por separado y registrar MapForce Server por más adelante.
- Si 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 (para la actualización a una nueva versión). En este caso, el asistente de instalación actualizará la versión de su LicenseServer automáticamente. La información del registro y de la licencia se traspasa a la versión más reciente de LicenseServer.
- Si 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:
 - Desactive la casilla *Instalar Altova LicenseServer en este equipo*.
 - En *Registrar este producto con* elija el LicenseServer con el que quiere registrar MapForce Server. También puede optar por la opción **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 MapForce Server.

Para más información sobre cómo registrar MapForce Server y asignarle una licencia con [Altova LicenseServer](#) consulte el apartado [Asignar licencias a MapForce Server](#)¹².

Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia (i) con la versión de LicenseServer correspondiente a la versión de MapForce Server instalada o (ii) con una versión posterior de LicenseServer.
- La versión de LicenseServer correspondiente a la versión actual de MapForce Server es **3.14**.
- En Windows, puede instalar esta versión de LicenseServer junto con MapForce Server o puede instalar LicenseServer por separado. En Linux y macOS, tiene que instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores.
- 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.
- Las versiones de LicenseServer son compatibles con versiones antiguas. Más concretamente,

funcionan con versiones más antiguas de MapForce Server.

- La última versión de LicenseServer está disponible en el sitio web de Altova. Esta versión funcionará con cualquier versión actual o anterior de MapForce Server.
- El número de versión de LicenseServer siempre aparece al final de la [página de configuración de LicenseServer](#).

2.1.4 Asignar licencias a MapForce Server (Windows)

Para poder trabajar con MapForce 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 MapForce Server** con LicenseServer desde MapForce Server.
2. El segundo paso consiste en **asignar una licencia** a MapForce 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](#).

2.1.4.1 Iniciar LicenseServer, MapForce Server

Tanto Altova LicenseServer (LicenseServer para abreviar) como MapForce Server se inician con Altova ServiceController.

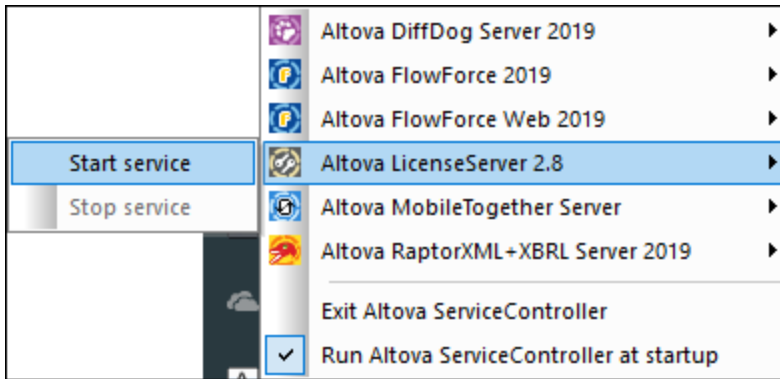
Altova ServiceController

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 (DiffDog Server, FlowForce Server, Mobile Together Server y RaptorXML(+XBRL) Server). Una vez iniciado, podrá acceder a ServiceController desde la bandeja del sistema (*imagen siguiente*).

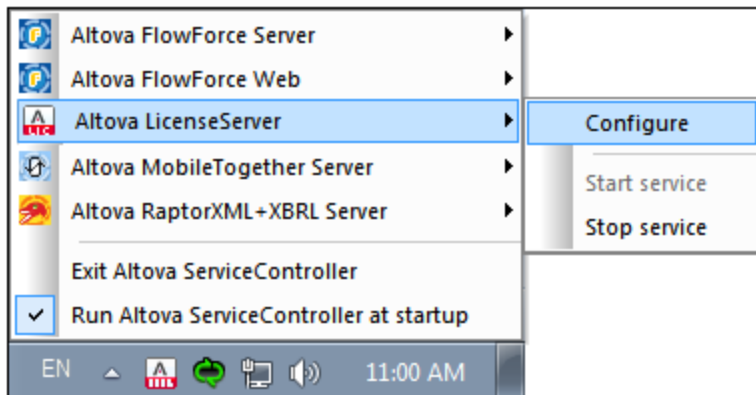


Si quiere que ServiceController se inicie automáticamente nada más iniciar sesión en el sistema, haga clic en el icono de **ServiceController** de la bandeja del sistema para abrir el menú de opciones de **ServiceController** (*imagen siguiente*) y active la opción **Run Altova ServiceController at Startup** (*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 Altova LicenseServer en el menú que aparece (*imagen siguiente*) y seleccione **Iniciar servicio** en el submenú de LicenseServer. Si LicenseServer ya está en ejecución, la opción *Start Service* se deshabilita. También puede detener el servicio con ServiceController.



Start MapForce Server

Para iniciar MapForce Server haga clic en el icono de **ServiceController** en la bandeja del sistema, pase el puntero del ratón por encima de Altova MapForce Server en el menú que aparece y seleccione **Iniciar servicio** en el submenú de MapForce Server. Si MapForce Server ya está en ejecución, la opción *Iniciar servicio* se deshabilita. También puede detener el servicio con ServiceController.

2.1.4.2 Registrar MapForce Server

Para poder asignar una licencia a MapForce Server con Altova LicenseServer debe registrar primero la aplicación con LicenseServer.

Para registrar MapForce Server desde la línea de comandos use el comando `licenseserver` e introduzca la dirección del equipo en el que está instalado LicenseServer (*ver más abajo*).

```
MapForceServer licenseserver [opciones] Servidor-O-Dirección-IP
```

Por ejemplo, si el nombre del servidor donde está instalado LicenseServer es `localhost`, entonces debe usar este comando:

```
MapForceServer licenseserver localhost
```

Si MapForce Server se instaló como parte de la instalación de [FlowForce Server](#), al registrar FlowForce Server con con LicenseServer también se registrará automáticamente MapForce Server. En resumen: (i) Inicie Altova FlowForce Web con ServiceController (*véase punto anterior*), (ii) introduzca su contraseña para acceder a la página de configuración, (iii) seleccione el nombre o la dirección del LicenseServer y haga clic en **Registrar con LicenseServer**. Para más información consulte [Registrar FlowForce Server](#).

Una vez haya registrado su producto correctamente vaya a [la pestaña Client Management de la página de configuración de LicenseServer](#) y asigne una licencia a MapForce Server.

Para más información sobre cómo registrar los productos de Altova con LicenseServer consulte el [manual del usuario de LicenseServer](#).

2.1.4.3 Asignar licencias a MapForce Server

Una vez haya registrado MapForce Server correctamente, la aplicación aparecerá en la pestaña Client Management de la página de configuración de LicenseServer. Ahora puede [asignar una licencia](#) a MapForce 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 sería menos rápida que si utilizaran 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.

Licencias de FlowForceServer y MapForceServer

Flowforce Server Advanced Edition y Mapforce Server Advanced Edition solo pueden ejecutarse en equipos con dos o más núcleos.

Para calcular el número de núcleos para los que necesita licencias, debe tener en cuenta el volumen de datos que necesita procesar y el tiempo de procesamiento que cabe esperar en su entorno. En la mayoría de los casos, un mayor número de núcleos significa un mayor volumen de datos procesados en menos tiempo. A continuación facilitamos algunos consejos para aplicaciones:

- El servidor de FlowForce se ejecuta como una aplicación con varios subprocesos. Si el número de solicitudes simultáneas en el servidor es grande, un número insuficiente de núcleos resultará en latencia (tiempos de espera). Por ejemplo, si está exponiendo trabajos como servicios web, es posible que haya cientos de solicitudes de clientes al mismo tiempo. En este caso, un mayor número de núcleos beneficiaría al servidor de FlowForce.
- El servidor de MapForce usará un único núcleo por asignación. Por tanto, si necesita ejecutar varias asignaciones al mismo tiempo, es muy recomendable contar con un mayor número de núcleos. Por ejemplo, cuando el servidor de MapForce se ejecuta bajo la gestión del servidor de FlowForce, es posible que se solape la ejecución de varios trabajos de asignación al mismo tiempo, lo que también depende de la configuración. Sin embargo, hay que tener en cuenta que si los volúmenes procesador por las asignaciones son demasiado grandes, seguiría existiendo riesgo de latencia.

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.

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 del número de núcleos

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.

2.2 Instalación y configuración en Linux

Esta sección explica cómo [instalar](#) ¹⁶ MapForce Server y [asignarle licencias](#) ¹⁹ en sistemas Linux (Debian, Ubuntu, CentOS, RedHat).

Requisitos del sistema (Linux)

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

Requisitos

- Puede instalarlo como usuario raíz o como usuario con privilegios `sudo`.
- Debe desinstalar la versión anterior de MapForce Server antes de instalar una nueva.
- Estas son las fuentes necesarias como requisito para instalar y ejecutar 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.

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

2.2.1 Instalar en Linux

MapForce Server se puede instalar en sistemas Linux. Puede instalarlo como usuario `raíz` o como usuario con privilegios `sudo`.

Integración de FlowForce Server y otros productos servidor de Altova

Si instala MapForce Server junto con FlowForce Server, recomendamos que instale primero FlowForce Server. Si instala MapForce Server antes que FlowForce Server, una vez haya instalado las dos aplicaciones ejecute este comando:

```
cp /opt/Altova/MapForceServer2024/etc/*.tool /opt/Altova/FlowForceServer2024/tools
```

Con este comando se copia el archivo `.tools` del directorio `/etc` de MapForce Server en el directorio `/tools` de FlowForce Server. El archivo `.tool` es necesario para usar FlowForce Server. Este archivo contiene la ruta al ejecutable de MapForce Server. Si instala FlowForce antes de instalar MapForce Server no necesita ejecutar este comando.

Desinstalar MapForce Server

Tiene que desinstalar cualquier versión anterior de MapForce Server que tenga instalada antes de poder instalar la versión nueva de la aplicación.

Para comprobar qué productos servidor de Altova están ya instalados, utilice uno de los siguientes comandos:

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

Si tiene que desinstalar una versión anterior de MapForce Server puede usar uno de los siguientes comandos:

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

En sistemas Debian y Ubuntu puede ocurrir que MapForce Server siga apareciendo en la lista de productos instalados incluso después de haberlo desinstalado. En este caso puede ejecutar el comando `purge` para eliminar MapForce Server de la lista. También puede usar el comando `purge` en lugar del comando *instead* que mencionamos más arriba.

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

Descargar el paquete de instalación de MapForce Server para Linux

Puede descargar los paquetes de instalación de MapForce Server para los sistemas Linux siguientes en el [sitio web de Altova](#).

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

Una vez haya descargado el paquete de instalación para Linux, copie el paquete en cualquier directorio del sistema Linux. Como necesitará un [Altova LicenseServer](#) para ejecutar MapForce Server, puede que quiera descargar LicenseServer del [sitio web de Altova](#) al mismo tiempo que MapForce Server en lugar de descargarlo más tarde.

Instalar MapForce Server

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

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

Instale MapForce Server con el comando correspondiente:

```
[Debian]: sudo dpkg --install mapforceserveradv-2024-debian.deb
[Ubuntu]: sudo dpkg --install mapforceserveradv-2024-ubuntu.deb
[CentOS]: sudo rpm -ivh mapforceserveradv-2024-1.x86_64.rpm
[RedHat]: sudo rpm -ivh mapforceserveradv-2024-1.x86_64.rpm
```

Quizás sea necesario ajustar el nombre del paquete anterior para que tenga el número de versión o de service pack actual.

El paquete de MapForce Server se instalará en esta carpeta:

```
/opt/Altova/MapForceServer2024
```

2.2.2 Instalar LicenseServer (Linux)

Para poder utilizar MapForce Server debe asignarle una licencia con un [Altova LicenseServer](#) de su red. Descargue el instalador de LicenseServer del [sitio web de Altova](#) y copie el paquete en cualquier directorio. Instálelo tal y como instaló MapForce Server (consulte [el apartado anterior](#)¹⁶).

```
[Debian]: sudo dpkg --install licenseserver-3.14-debian.deb
[Ubuntu]: sudo dpkg --install licenseserver-3.14-ubuntu.deb
[CentOS]: sudo rpm -ivh licenseserver-3.14-1.x86_64.rpm
[RedHat]: sudo rpm -ivh licenseserver-3.14-1.x86_64.rpm
```

El paquete de LicenseServer se instalará en la siguiente ruta de acceso:

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

Para más información sobre cómo registrar MapForce Server y asignarle una licencia con [Altova LicenseServer](#) consulte el apartado [Asignar licencias a MapForce Server](#)¹⁹. Consulte también [la documentación de LicenseServer](#) para obtener información más detallada.

Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia (i) con la versión de LicenseServer correspondiente a la versión de MapForce Server instalada o (ii) con una versión posterior de LicenseServer.
- La versión de LicenseServer correspondiente a la versión actual de MapForce Server es **3.14**.
- En Windows, puede instalar esta versión de LicenseServer junto con MapForce Server o puede instalar LicenseServer por separado. En Linux y macOS, tiene que instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores.
- 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.
- Las versiones de LicenseServer son compatibles con versiones antiguas. Más concretamente, funcionan con versiones más antiguas de MapForce Server.
- La última versión de LicenseServer está disponible en el sitio web de Altova. Esta versión funcionará con cualquier versión actual o anterior de MapForce Server.
- El número de versión de LicenseServer siempre aparece al final de la [página de configuración de LicenseServer](#).

2.2.3 Asignar licencias a MapForce Server (Linux)

Para poder trabajar con MapForce 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 MapForce Server** con LicenseServer desde MapForce Server.
2. El segundo paso consiste en **asignar una licencia** a MapForce 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](#).

2.2.3.1 Iniciar LicenseServer, MapForce Server

Inicie Altova LicenseServer y MapForce Server o como usuario `root` o como usuario normal con privilegios `sudo`.

Iniciar LicenseServer

Para registrar y asignar una licencia a MapForce Server con LicenseServer debe ejecutar LicenseServer como demonio. Para ello use este comando:

```
sudo systemctl start licenseserver
```

Si necesita detener LicenseServer en algún momento, reemplace `start` con `stop` en el comando anterior. Por ejemplo:

```
sudo systemctl stop licenseserver
```

2.2.3.2 Registrar MapForce Server

Para poder asignar una licencia a MapForce Server con Altova LicenseServer debe registrar primero la aplicación con LicenseServer.

Para registrar MapForce Server desde la línea de comandos use el comando `licenseserver`:

```
sudo /opt/Altova/MapForceServer2024/bin/mapforceserver licenseserver [opciones]  
Servidor-O-Dirección-IP
```

Por ejemplo, si el nombre del servidor donde está instalado LicenseServer es `localhost`, entonces debe usar este comando:

```
sudo /opt/Altova/MapForceServer2024/bin/mapforceserver licenseserver localhost
```

En el comando anterior, `localhost` es el nombre del servidor en el que está instalado LicenseServer. Observe que el ejecutable de MapForce Server se encuentra en:

```
/opt/Altova/MapForceServer2024/bin/
```

Una vez haya registrado su producto correctamente vaya a [la pestaña Client Management de la página de configuración de LicenseServer](#) y asigne una licencia a MapForce Server.

Para más información sobre cómo registrar los productos de Altova con LicenseServer consulte el [manual del usuario de LicenseServer](#).

2.2.3.3 Asignar licencias a MapForce Server

Una vez haya registrado MapForce Server correctamente, la aplicación aparecerá en la pestaña Client Management de la página de configuración de LicenseServer. Ahora puede [asignar una licencia](#) a MapForce 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 sería menos rápida que si utilizaran 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.

Licencias de FlowForceServer y MapForceServer

Flowforce Server Advanced Edition y Mapforce Server Advanced Edition solo pueden ejecutarse en equipos con dos o más núcleos.

Para calcular el número de núcleos para los que necesita licencias, debe tener en cuenta el volumen de datos que necesita procesar y el tiempo de procesamiento que cabe esperar en su entorno. En la mayoría de los casos, un mayor número de núcleos significa un mayor volumen de datos procesados en menos tiempo. A continuación facilitamos algunos consejos para aplicaciones:

- El servidor de FlowForce se ejecuta como una aplicación con varios subprocesos. Si el número de

solicitudes simultáneas en el servidor es grande, un número insuficiente de núcleos resultará en latencia (tiempos de espera). Por ejemplo, si está exponiendo trabajos como servicios web, es posible que haya cientos de solicitudes de clientes al mismo tiempo. En este caso, un mayor número de núcleos beneficiaría al servidor de FlowForce.

- El servidor de MapForce usará un único núcleo por asignación. Por tanto, si necesita ejecutar varias asignaciones al mismo tiempo, es muy recomendable contar con un mayor número de núcleos. Por ejemplo, cuando el servidor de MapForce se ejecuta bajo la gestión del servidor de FlowForce, es posible que se solape la ejecución de varios trabajos de asignación al mismo tiempo, lo que también depende de la configuración. Sin embargo, hay que tener en cuenta que si los volúmenes procesador por las asignaciones son demasiado grandes, seguiría existiendo riesgo de latencia.

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.

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 del número de núcleos

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.

2.3 Instalación y configuración en macOS

Esta sección explica cómo [instalar](#)²² MapForce Server y [asignarle licencias](#)²⁴ en sistemas macOS.

Requisitos del sistema (macOS)

Tenga en cuenta estos requisitos del sistema:

- macOS 12 o superior

Requisitos

Tenga en cuenta los siguientes requisitos:

- Asegúrese de que Altova LicenseServer está instalado y ejecutándose.
- Puede instalarlo como usuario `raíz` o como usuario con privilegios `sudo`.
- Debe desinstalar la versión anterior de MapForce Server antes de instalar una nueva.
- El equipo macOS debe estar configurado de forma que su nombre se resuelva en una dirección de IP. Esto significa que debe poder hacerle ping al nombre de host desde la terminal con el comando `ping <hostname>`.

2.3.1 Instalar en macOS

Esta sección explica cómo instalar y configurar MapForce Server en sistemas macOS.

Integración con FlowForce

Si instala MapForce Server junto con FlowForce Server, recomendamos que instale primero FlowForce Server. Si instala MapForce Server antes que FlowForce Server, entonces una vez haya instalado las dos aplicaciones ejecute este comando:

```
cp /usr/local/Altova/MapForceServer2024/etc/*.tool /usr/local/Altova/FlowForceServer2024/tools
```

Con este comando se copia el archivo `.tools` del directorio `/etc` de MapForce Server en el directorio `/tools` de FlowForce Server. El archivo `.tool` es necesario para usar FlowForce Server. Este archivo contiene la ruta al ejecutable de MapForce Server. Si instala FlowForce antes de instalar MapForce Server no necesita ejecutar este comando.

Desinstalar MapForce Server

En la carpeta Aplicaciones, haga clic con el botón derecho en el icono de MapForce Server y seleccione **Mover a la papelera**. La aplicación se moverá a la papelera. pero debe quitar la aplicación de la carpeta `usr`. Para ello use el comando:

```
sudo rm -rf /usr/local/Altova/MapForceServer2024/
```

Si tiene que desinstalar una versión antigua de Altova License Server primer tiene que detener su ejecución como servicio. Para ello use el comando:

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

Para comprobar si el servicio se detuvo o no, abra la terminal del Monitor de actividad en Finder y confirme que LicenseServer no está en la lista. Ahora procesa a desinstalar la aplicación con los mismos pasos que hemos explicado antes para MapForce Server.

Instalar MapForce Server

Para instalar MapForce Server siga estos pasos:

1. Descargue el archivo de imagen de disco (.dmg) de MapForce Server desde el sitio web de Altova (<http://www.altova.com/es/download.html>).
2. Haga clic en el archivo para abrir la imagen de disco (.dmg). El instalador de MapForce Server aparece en el equipo como otra unidad de disco más.
3. En el disco virtual haga doble clic en el paquete del instalador (.pkg).
4. Siga los pasos del asistente de instalación. En uno de los pasos debe aceptar el contrato de licencia para poder seguir con la instalación. Consulte también [Asignar licencias a MapForce Server](#)²⁴.
5. Para expulsar el disco después de la instalación haga clic en él con el botón derecho y seleccione **Expulsar**.

El paquete de MapForce Server se instalará en la carpeta:

```
/usr/local/Altova/MapForceServer2024 (archivos binarios de la aplicación)  
/var/Altova/MapForceServer (archivos de datos: BD y registros)
```

El servidor demonio de MapForce Server se inicia automáticamente después de la instalación y de reiniciar el equipo. Siempre puede iniciar MapForce Server como demonio con este comando:

```
sudo launchctl load /Library/LaunchDaemons/com.altova.MapForceServer2024.plist
```

2.3.2 Instalar LicenseServer (macOS)

Puede descargar Altova LicenseServer desde el sitio web de Altova (<http://www.altova.com/es/download.html>). Ahora proceda al proceso de instalación descrito [aquí](#)²².

El paquete de LicenseServer se instalará en la siguiente carpeta:

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

Para más información sobre cómo registrar MapForce Server con [Altova LicenseServer](#) y asignarle una licencia consulte [Asignar licencias en macOS](#)²⁴.

Versiones de LicenseServer

- Los productos servidor de Altova deben tener una licencia (i) con la versión de LicenseServer correspondiente a la versión de MapForce Server instalada o (ii) con una versión posterior de LicenseServer.
- La versión de LicenseServer correspondiente a la versión actual de MapForce Server es **3.14**.
- En Windows, puede instalar esta versión de LicenseServer junto con MapForce Server o puede instalar

- LicenseServer por separado. En Linux y macOS, tiene que instalar LicenseServer por separado.
- Antes de instalar una versión nueva de LicenseServer, es necesario desinstalar versiones anteriores.
 - 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.
 - Las versiones de LicenseServer son compatibles con versiones antiguas. Más concretamente, funcionan con versiones más antiguas de MapForce Server.
 - La última versión de LicenseServer está disponible en el sitio web de Altova. Esta versión funcionará con cualquier versión actual o anterior de MapForce Server.
 - El número de versión de LicenseServer siempre aparece al final de la [página de configuración de LicenseServer](#).

2.3.3 Asignar licencias a MapForce Server (macOS)

Para poder trabajar con MapForce 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 MapForce Server** con LicenseServer desde MapForce Server.
2. El segundo paso consiste en **asignar una licencia** a MapForce 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](#).

2.3.3.1 Iniciar LicenseServer, MapForce Server

Inicie Altova LicenseServer y MapForce Server o como usuario `root` o como usuario normal con privilegios `sudo`.

Iniciar LicenseServer

Para registrar y asignar una licencia a MapForce Server con LicenseServer debe ejecutar LicenseServer como demonio. Para iniciar LicenseServer como demonio, ejecute este comando:

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

Si necesita detener LicenseServer en algún momento, reemplace `load` con `unload` en el comando anterior.

2.3.3.2 Registrar MapForce Server

Para poder asignar una licencia a MapForce Server con Altova LicenseServer debe registrar primero la aplicación con LicenseServer.

Para registrar MapForce Server desde la línea de comandos use el comando `licenseserver`:


```
sudo /usr/local/Altova/MapForceServer2024/bin/MapForceServer licenseserver [opciones]
Servidor-O-Dirección-IP
```

Por ejemplo, si el nombre del servidor donde está instalado LicenseServer es `localhost`, entonces debe usar este comando:

```
sudo /usr/local/Altova/MapForceServer2024/bin/MapForceServer licenseserver localhost
```

En el comando anterior, `localhost` es el nombre del servidor en el que está instalado LicenseServer. Observe que el ejecutable de MapForce Server se encuentra en:

```
/usr/local/Altova/MapForceServer2024/bin/
```

Una vez haya registrado su producto correctamente vaya a [la pestaña Client Management de la página de configuración de LicenseServer](#) y asigne una licencia a MapForce Server.

Para más información sobre cómo registrar los productos de Altova con LicenseServer consulte el [manual del usuario de LicenseServer](#).

2.3.3.3 Asignar licencias a MapForce Server

Una vez haya registrado MapForce Server correctamente, la aplicación aparecerá en la pestaña Client Management de la página de configuración de LicenseServer. Ahora puede [asignar una licencia](#) a MapForce 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 sería menos rápida que si utilizaran 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.

Licencias de FlowForceServer y MapForceServer

Flowforce Server Advanced Edition y Mapforce Server Advanced Edition solo pueden ejecutarse en equipos con dos o más núcleos.

Para calcular el número de núcleos para los que necesita licencias, debe tener en cuenta el volumen de datos que necesita procesar y el tiempo de procesamiento que cabe esperar en su entorno. En la mayoría de los casos, un mayor número de núcleos significa un mayor volumen de datos procesados en menos tiempo. A continuación facilitamos algunos consejos para aplicaciones:

- El servidor de FlowForce se ejecuta como una aplicación con varios subprocesos. Si el número de solicitudes simultáneas en el servidor es grande, un número insuficiente de núcleos resultará en latencia (tiempos de espera). Por ejemplo, si está exponiendo trabajos como servicios web, es posible que haya cientos de solicitudes de clientes al mismo tiempo. En este caso, un mayor número de núcleos beneficiaría al servidor de FlowForce.
- El servidor de MapForce usará un único núcleo por asignación. Por tanto, si necesita ejecutar varias asignaciones al mismo tiempo, es muy recomendable contar con un mayor número de núcleos. Por ejemplo, cuando el servidor de MapForce se ejecuta bajo la gestión del servidor de FlowForce, es posible que se solape la ejecución de varios trabajos de asignación al mismo tiempo, lo que también depende de la configuración. Sin embargo, hay que tener en cuenta que si los volúmenes procesador por las asignaciones son demasiado grandes, seguiría existiendo riesgo de latencia.

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.

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 del número de núcleos

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.

2.4 Actualizar MapForce Server

La forma más fácil de transferir una licencia de una versión anterior de MapForce Server a la más reciente es seguir los pasos del proceso de instalación. Estos son los pasos principales que debe llevar a cabo durante la instalación:

1. Registre la versión nueva de MapForce Server con el LicenseServer en el que está la licencia que usaba la versión anterior de MapForce Server.
2. Acepte el acuerdo de la licencia de MapForce Server. (Si no lo hace no se instalará la versión más reciente.)

Nota: Si no registra MapForce Server con LicenseServer durante el proceso de instalación, podrá hacerlo más adelante y después asignarle una licencia.

2.5 Migrar MapForce Server a un equipo nuevo

Si quiere migrar MapForce Server de un equipo a otro (también entre plataformas compatibles) siga las instrucciones de más abajo.

Migrar MapForce Server a otro equipo consiste en asignar la licencia del equipo antiguo al equipo nuevo. Para ello siga estas instrucciones:

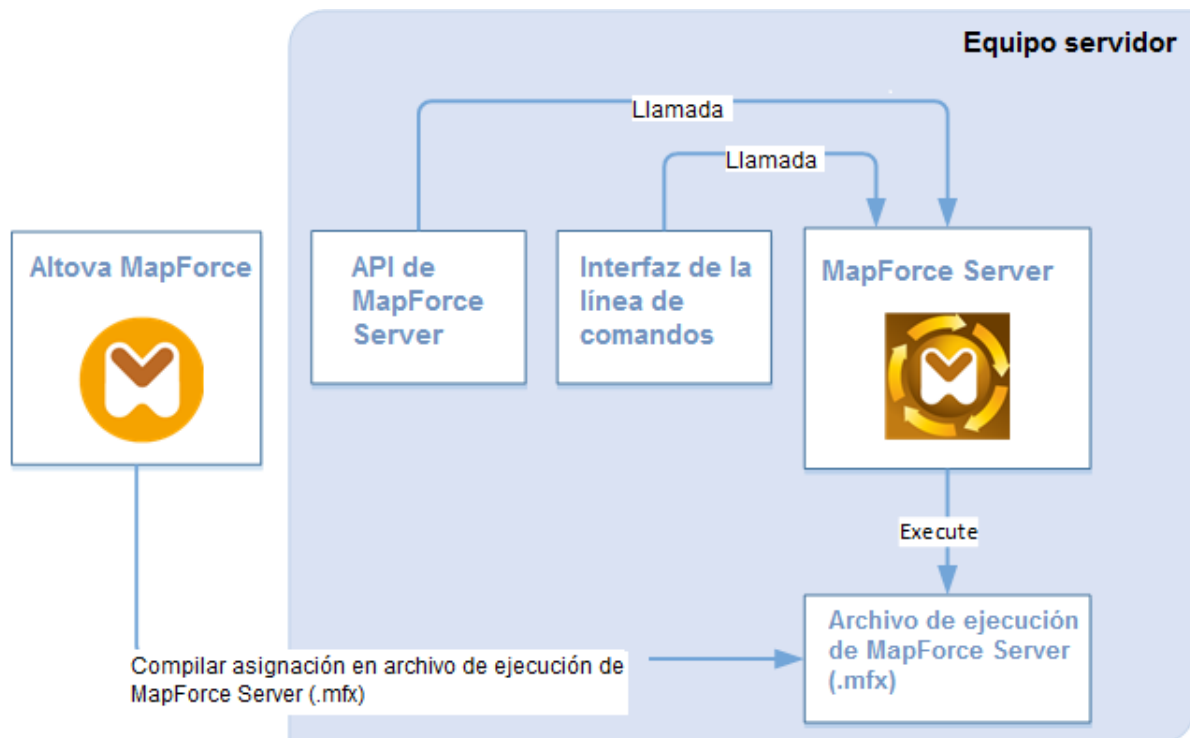
1. Instale MapForce Server en el equipo nuevo. Si ya instaló el producto como parte de la instalación de FlowForce Server, ignore este paso.
2. En el equipo nuevo registre MapForce Server con Altova LicenseServer.
3. En el equipo antiguo asegúrese de que ningún cliente está usando el servidor (p.ej. no se están ejecutando asignaciones de datos).
4. Abra la página de administración de Altova LicenseServer. Desactive la licencia de MapForce Server en el equipo antiguo y vuelva a asignarla en el equipo nuevo.

3 Ejecutar asignaciones

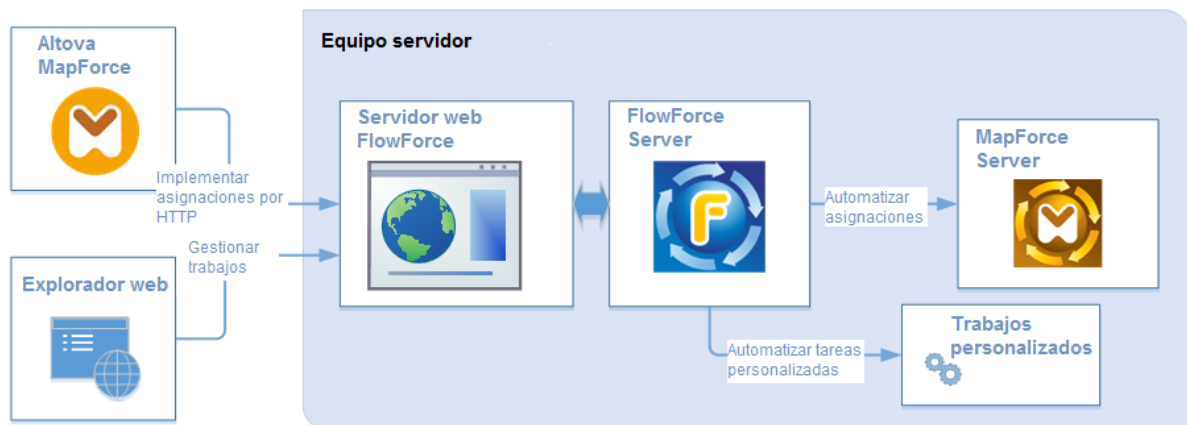
La función de MapForce Server es ejecutar asignaciones de datos creadas con Altova MapForce (<http://www.altova.com/es/mapforce.html>).

Primero debe diseñar asignaciones de datos (archivos *.mfd) en Altova MapForce, donde puede definir las entradas y salidas y otros pasos de procesamiento intermedios que se deban aplicar a los datos (p. ej. ordenación, filtrado, funciones personalizadas, etc.). Cuando la asignación esté lista, podrá ejecutarla con MapForce Server de varias maneras:

- En el equipo Windows donde se ejecute MapForce, compile la asignación en un archivo de ejecución de MapForce Server (.mfx). Los archivos .mfx de hecho son asignaciones de datos que se empaquetan para poder ejecutarse en un entorno servidor. Estos archivos se pueden copiar en cualquier sistema operativo compatible con MapForce Server (véase [Requisitos del sistema](#)⁷). En el equipo servidor puede ejecutar el archivo .mfx desde la interfaz de la línea de comandos de MapForce Server o con la API de MapForce Server.



- En el equipo Windows donde se ejecuta Altova MapForce, implemente la asignación en un equipo servidor donde estén instalados MapForce Server y FlowForce Server. El equipo servidor puede tener un sistema operativo distinto (véase [Requisitos del sistema](#)⁷). Las asignaciones implementadas de esta forma se convierten en funciones de FlowForce Server y puede crear trabajos programados o basados en eventos a partir de ellas. Cuando las asignaciones de datos se ejecutan como trabajos de FlowForce Server, también se pueden exponer como servicios web, se pueden encadenar como pasos subordinados de otros trabajos o pueden formar parte de flujos de trabajo (como enviar correos electrónicos, verificar códigos de salida, ejecutar comandos de shell, etc.).



Para obtener más información sobre este modelo de trabajo consulte la documentación de Altova FlowForce Server (<http://www.altova.com/es/documentation>).

Cómo ejecutar asignaciones compiladas como archivos de ejecución de MapForce Server

1. En MapForce Enterprise o Professional abra la asignación que desea compilar.
2. En el menú **Archivo** haga clic en **Compilar** en archivo de ejecución de MapForce Server y seleccione el directorio de destino.
3. Copie el archivo `.mfx` al directorio de destino o servidor junto con cualquier archivo de entrada o dependencia. Para más información consulte el apartado [Preparar asignaciones para ejecución en servidor](#)³².
4. Llame al comando `run` desde la [interfaz de la línea de comandos](#)⁴⁷ o el método equivalente de la [API de MapForce Server](#)⁶².

Cómo ejecutar asignaciones implementadas en FlowForce Server

1. En MapForce Enterprise o Professional abra la asignación que desea compilar.
2. Compruebe que el lenguaje de transformación (el motor de ejecución) de la asignación es el motor integrado. Si no lo es, selecciónelo con el comando **Resultados | Motor de ejecución integrado** o con el botón **BUILT IN** de la barra de herramientas.
3. En el menú **Archivo** haga clic en **Implementar en FlowForce Server**.
4. Introduzca los datos de conexión del servidor (host y puerto), las credenciales de FlowForce y el contenedor de destino de FlowForce. Para crear inmediatamente el trabajo de FlowForce en el explorador seleccione la opción *Abrir el explorador web para crear un trabajo nuevo*. También puede crear el trabajo de FlowForce más adelante (*ver paso siguiente*).
5. Abra el explorador e inicie sesión en la interfaz de administración de FlowForce Server. Navegue hasta el contenedor donde se implementó la asignación. Este paso no es necesario si en el paso anterior seleccionó la opción *Abrir el explorador web para crear un trabajo nuevo*.
6. Defina el trabajo de FlowForce Server, incluidos sus desencadenadores, parámetros y pasos de ejecución adicionales (consulte la [documentación de FlowForce Server](#) para ver ejemplos). Cuando se ejecute el trabajo, la transformación de la asignación subyacente se ejecutará y se generará el resultado de la asignación.

Nota: si MapForce Server se ejecuta en un equipo distinto al equipo donde se diseñó la asignación, deberá ajustar las rutas de acceso de los archivos de entrada o los datos de conexión a la base de datos para que sean válidos en el entorno de ejecución de destino. Por ejemplo, si una asignación llama a una base de datos y necesita un controlador de base de datos, este controlador debe estar instalado en el entorno de destino. Si no lo está, la asignación no se podrá ejecutar correctamente. Para ver o ajustar los datos de conexión a la base de datos, haga clic con el botón derecho en el componente de base de datos en MapForce y seleccione **Propiedades**. Tras realizar los cambios necesarios en el diseño de asignación, no se olvide de volver a compilarlo en un archivo de ejecución de MapForce Server (.mfx) o, dependiendo del caso, de volver a implementarlo en FlowForce Server. Para más información consulte el apartado siguiente ([Preparar asignaciones para ejecución en servidor](#)³²).

3.1 Preparar asignaciones para ejecución en servidor

Una asignación de datos diseñada y probada con MapForce puede hacer referencia a recursos (p. ej. bases de datos) situados fuera del equipo actual y en otro sistema operativo. Además, en MapForce, las rutas de acceso de las asignaciones de datos siguen por defecto las normas de Windows. Y, por último, el equipo donde se ejecuta MapForce Server puede ser compatible o no con las conexiones de base de datos con las que era compatible el equipo donde se diseñó la asignación de datos. Por todos estos motivos, será necesario preparar las asignaciones para ejecutarlas en el entorno del servidor, sobre todo si el equipo de destino no es el mismo que el equipo de origen.

Nota: El término "equipo de origen" se usa para designar al equipo donde está instalado MapForce, mientras que "equipo de destino" se refiere al equipo donde está instalado MapForce Server o FlowForce Server. En los casos más sencillos, se trata del mismo equipo, pero en casos más complejos sería más posible ejecutar MapForce en un equipo Windows y ejecutar MapForce Server o FlowForce Server en un equipo Linux o macOS.

Por todo ello, lo más recomendable es comprobar que la asignación es válida (desde MapForce) antes de implementarla en FlowForce Server o compilarla en un archivo de ejecución de MapForce Server.

Si MapForce Server se ejecuta de forma independiente (y no junto con FlowForce Server), estas son las licencias que necesitará:

- En el equipo de origen, necesitará la edición MapForce Enterprise o Professional para diseñar la asignación y compilarla en un archivo de ejecución de MapForce Server (.mfx).
- En el equipo de destino, necesitará la edición MapForce Server o MapForce Server Advanced Edition para ejecutar la asignación de datos.

Si MapForce Server se gestiona desde FlowForce Server, entonces necesitará:

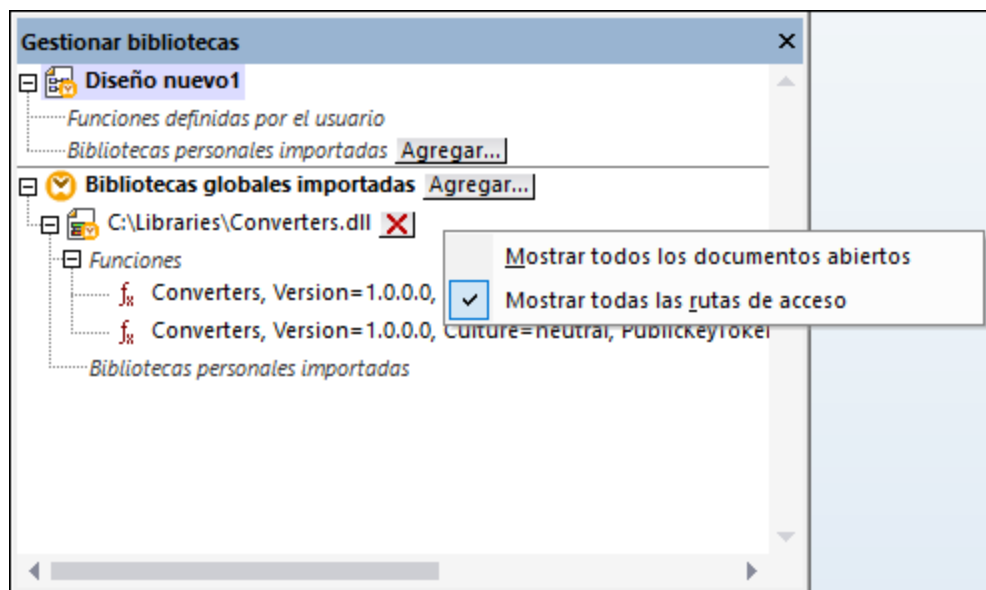
- En el equipo de origen, necesitará la edición MapForce Enterprise o Professional para diseñar la asignación e implementarla en el equipo de destino.
- Tanto MapForce Server como FlowForce Server deben tener asignada una licencia en el equipo de destino. MapForce Server se encarga de ejecutar la asignación de datos, mientras que FlowForce Server sirve la asignación en forma de trabajo, lo cual permite acceder a numerosas características (como la ejecución bajo demanda o programada, la ejecución como servicio web, control de errores, procesamiento condicional, notificaciones por correo electrónico, etc.).
- FlowForce Server debe estar en funcionamiento en la dirección de red y puerto configurados. Es decir, el servicio "FlowForce Web Server" debe estar iniciado y configurado para aceptar conexiones de clientes HTTP (o HTTPS si corresponde) y no debe estar bloqueado por el servidor de seguridad. El servicio "FlowForce Server" también debe estar iniciado y en ejecución en la dirección y puerto designados.
- Debe tener una cuenta de usuario de FlowForce Server con permiso para acceder a uno de los contenedores (al contenedor **/public** pueden acceder todos los usuarios que inicien sesión).

Consideraciones generales

- Si tiene pensado ejecutar la asignación en un equipo de destino con una versión independiente de MapForce Server, deberá copiar todos los archivos de entrada a los que haga referencia la asignación al equipo de destino. Si por el contrario MapForce Server se gestiona desde FlowForce Server, entonces no es necesario copiar estos archivos a mano porque los archivos de instancia y de

esquema se incluyen en el paquete que se implementa en el equipo de destino.

- Si la asignación incluye componentes de base de datos que necesitan controladores de base de datos específicos, dichos controladores deberán estar instalados en el equipo de destino también. Por ejemplo, si la asignación lee datos de una base de datos de Microsoft Access, en el equipo de destino también deberá tener instalado Microsoft Access o Microsoft Access Runtime (<https://www.microsoft.com/es-es/download/details.aspx?id=50040>).
- Cuando se implementa una asignación en plataformas que no son Windows, las conexiones de base de datos ADO, ADO.NET y ODBC se cambian automáticamente por JDBC. Las conexiones nativas SQLite y PostgreSQL se mantienen como tal y no requieren otra configuración. (véase también "Conexiones de BD", más abajo).
- Si la asignación contiene llamadas a funciones personalizadas (por ejemplo, a archivos .dll o .class), dichas dependencias no se implementan junto con la asignación porque no se conocen antes del tiempo de ejecución. En dichos casos, cópielas a mano en el equipo de destino. La ruta de acceso a los archivos .dll o .class en el servidor debe ser la misma en el cuadro de diálogo "Opciones" de MapForce; por ejemplo:



- Algunas asignaciones leen varios archivos de entrada con ayuda de una ruta de acceso comodín. En este caso el nombre de los archivos de entrada no se conocen antes del tiempo de ejecución y, por tanto, no se implementan. Para poder ejecutar la asignación correctamente los archivos de entrada deben existir en el equipo de destino.
- Si la ruta de salida de la asignación incluye directorios, estos deben existir en el equipo de destino. De lo contrario se generará un error al ejecutar la asignación. Para evitar eso, en MapForce los directorios que no existen se generan automáticamente si se habilita la opción Generar resultados en archivos temporales .
- Si la asignación llama a un servicio web que requiere autenticación HTTPS con un certificado de cliente, el certificado debe transferirse al equipo de destino (véase [Gestión de certificados digitales](#) ¹²⁵).
- Si la asignación se conecta a bases de datos basadas en archivos (como Microsoft Access y SQLite), deberá transferir a mano el archivo de base de datos al equipo de destino o guardarlo en un directorio compartido al que pueda acceder tanto el equipo de origen como el de destino (ver más abajo).

Rutas portátiles

Si tiene pensado ejecutar la asignación en un servidor, asegúrese de que la asignación sigue la convención

sobre rutas de acceso correspondiente y que utiliza una conexión de base de datos compatible.

1. Abra un archivo de diseño de asignación (.mfd) con MapForce en Windows.
2. En el menú **Archivo**, seleccione **Configurar asignación** y desmarque la casilla *Convertir las rutas de acceso en absolutas en el código generado* si estaba marcada.
3. En cada componente de la asignación abra el cuadro de diálogo "Propiedades" (haga doble clic en la barra de título del componente, por ejemplo) y cambie todas las rutas de los archivos de absolutas a relativas. Marque también la casilla *Guardar todas las rutas de acceso de archivos como relativas al archivo MFD*. Si le resulta más cómodo, puede copiar todos los archivos y esquemas de entrada en la misma carpeta en la que se encuentra la asignación para poder hacer referencia a ellos usando solamente su nombre.

Para más información sobre las rutas relativas y absolutas en el diseño de asignaciones consulte la documentación de MapForce.

Es importante saber que tanto MapForce Server como FlowForce Server funcionan con un *directorio de trabajo* con el que se resuelven todas las rutas de acceso relativas . El directorio de trabajo se especifica en tiempo de ejecución:

- En FlowForce Server se especifica editando el parámetro `Directorio de trabajo` de un trabajo.
- En la API de MapForce Server API se especifica con la propiedad `WorkingDirectory` de la API de COM y .NET API o con el método `setWorkingDirectory` de la API de Java.
- En la línea de comandos de MapForce Server el directorio de trabajo es el directorio actual de la shell de comandos.

Conexiones de BD

Tenga en cuenta que los equipos Linux y macOS no son compatibles con conexiones ADO, ADO.NET y ODBC. Por tanto, si el equipo de destino es Linux o macOS, estas conexiones se convertirán en conexiones JDBC cuando la asignación se implemente en FlowForce o cuando se compile en un archivo de ejecución de MapForce Server. Cuando esto ocurra, antes de implementar o compilar la asignación podrá elegir entre dos opciones:

- Crear la conexión JDBC con la base de datos (véase *Configurar una conexión JDBC*)
- Rellenar los datos de conexión de base de datos JDBC en la sección *Configuración JDBC* del componente de base de datos

Si la asignación utiliza una conexión nativa a una base de datos PostgreSQL o SQLite, la conexión nativa se conserva y la conversión en JDBC no tiene lugar . Si la asignación se conecta a una base de datos basada en archivos (como Microsoft Access y SQLite) deberá configurar algunas opciones más (véase *el punto siguiente*).

Para ejecutar asignaciones con conexiones JDBC es necesario tener instalado Java Runtime Environment o Java Development Kit en el equipo servidor. Este último puede ser Oracle JDK o uno de código abierto, como Oracle OpenJDK.

- La variable de entorno `JAVA_HOME` debe apuntar al directorio de instalación de JDK.
- En Windows tiene prioridad la ruta de acceso a un equipo virtual Java del registro frente a la variable `JAVA_HOME`.
- La plataforma JDK (32 o 64 bits) debe ser la misma que la de MapForce Server. De lo contrario es posible que reciba un error de tipo "No se puede acceder al equipo virtual Java".

Para configurar una conexión JDBC en Linux o macOS:

1. Descargue el controlador JDBC que ofrece el proveedor de la BD e instálelo en el sistema. Recuerde que debe seleccionar la versión de 32 bits si su sistema es de 32 bits y la versión de 64 bits si el sistema es de 64 bits.
2. Configure las variables de entorno con la ubicación donde está instalado el controlador JDBC. Por lo general será necesario configurar la variable `CLASSPATH`, pero quizá alguna más. Consulte la documentación del controlador JDBC para obtener más información.

Nota: En macOS, el sistema espera que las bibliotecas JDBC instaladas estén en el directorio `/Library/Java/Extensions`. Por tanto, se recomienda desempaquetar el controlador JDBC en esta ubicación. De lo contrario, deberá configurar el sistema para que busque la biblioteca JDBC en una ruta de acceso diferente.

Conexiones de cliente instantáneo de Oracle en macOS

Siga estas instrucciones si se va a conectar a bases de datos Oracle con el cliente **Oracle Database Instant Client** en macOS. Los requisitos son:

- Debe tener instalada la versión de Java 8.0 o posterior. Recuerde que si usa una versión de Java anterior a Java 8, también se podrá conectar con la biblioteca **JDBC Thin for All Platforms**, en cuyo caso puede ignorar las instrucciones que aparecen en este tema.
- Debe tener instalado Oracle Instant Client. El cliente Oracle Instant Client se puede descargar del sitio web oficial de Oracle. Tenga en cuenta que en la página de descargas hay varios paquetes y deberá seleccionar un paquete con compatibilidad con Oracle Call Interface (OCI), como el paquete Instant Client Basic, por ejemplo. Además, recuerde que debe seleccionar la versión de 32 bits si su sistema es de 32 bits o la versión de 64 bits si su sistema es de 64 bits.

Tras descargar y desempaquetar el cliente Oracle Instant Client, edite el archivo de lista de propiedades (`.plist`) que viene con el instalador para que las variables de la tabla apunten a la ruta de acceso del controlador correspondiente. Por ejemplo:

Variable	Valor de muestra
<code>CLASSPATH</code>	<code>/opt/oracle/instantclient_11_2/ojdbc6.jar:/opt/oracle/instantclient_11_2/ojdbc5.jar</code>
<code>TNS_ADMIN</code>	<code>/opt/oracle/NETWORK_ADMIN</code>
<code>ORACLE_HOME</code>	<code>/opt/oracle/instantclient_11_2</code>
<code>DYLD_LIBRARY_PATH</code>	<code>/opt/oracle/instantclient_11_2</code>
<code>PATH</code>	<code>\$PATH:/opt/oracle/instantclient_11_2</code>

Nota: Los valores de las variables deben ajustarse a la ruta de acceso donde están instalados los archivos del cliente Oracle Instant Client.

Bases de datos basadas en archivos

Las bases de datos basadas en archivos (como Microsoft Access y SQLite) no se incluyen en el paquete que

se implementa en FlowForce Server ni en el archivo de ejecución que se compila para MapForce Server. Por tanto, si el equipo de origen y de destino no son el mismo, deberá seguir estas instrucciones:

1. En MapForce haga clic con el botón derecho en la asignación y en el cuadro de diálogo "Configurar asignación" desactive la casilla *Convertir las rutas de acceso en absolutas en el código generado*.
2. Haga clic con el botón derecho en el componente de base de datos de la asignación y añada una conexión al archivo de base de datos usando una ruta de acceso relativa. La manera más sencilla de evitar problemas relacionados con la ruta de acceso es guardar el diseño de asignación (archivo .mfd) en el mismo directorio que el archivo de base de datos y hacer referencia a este último desde la asignación usando el nombre de archivo (es decir, una ruta de acceso relativa).
3. Copie el archivo de base de datos en un directorio del equipo de destino (que llamaremos *directorio de trabajo*). Recuerde el nombre de este directorio porque lo necesitará para ejecutar la asignación en el servidor (ver más abajo).

Hay varias maneras de ejecutar este tipo de asignaciones en el servidor:

- Si tiene pensado ejecutar la asignación con MapForce Server bajo control de FlowForce Server, configure el trabajo de FlowForce Server para que apunte al directorio de trabajo creado previamente. El archivo de base de datos debe residir en este directorio de trabajo (puede ver un ejemplo en https://www.altova.com/manual/es/flowforceserveradvanced/2024.2/index.html?fs_example_web_service.htm).
- Si tiene pensado ejecutar la asignación con MapForce Server de forma independiente y desde la línea de comandos, cambie del directorio actual al directorio de trabajo (p. ej. `cd ruta\de\directorio\de\trabajo`) antes de llamar al comando `run` de MapForce Server.
- Si tiene pensado ejecutar la asignación con la API de MapForce Server, especifique el directorio de trabajo mediante programación antes de ejecutar la asignación. Para esto sirve la propiedad `WorkingDirectory` del objeto MapForce Server en la API de COM y .NET. En la API de Java se utiliza el método `setWorkingDirectory`.

Si tanto el equipo de origen como el de destino es un equipo Windows que se ejecuta en una red local, también puede configurar la asignación para que lea el archivo de base de datos de un directorio compartido común:

1. Guarde el archivo de base de datos en un directorio compartido común al que pueda acceder tanto el equipo de origen como el de destino.
2. En la asignación haga clic con el botón derecho en el componente de base de datos y añada una conexión al archivo de base de datos usando una ruta de acceso absoluta.

Recursos globales

Si la asignación incluye referencias a recursos globales en lugar de rutas de acceso directas o conexiones de base de datos, también podrá usar los recursos globales en el lado servidor. Cuando compile una asignación en un archivo de ejecución de MapForce Server (.mfx), las referencias a los recursos globales se mantienen intactas para que pueda usarlas en el lado servidor cuando ejecute la asignación. Cuando implemente una asignación en FlowForce Server puede elegir si quiere que esta use recursos en el servidor.

Para que las asignaciones (o las funciones de asignación, en el caso de FlowForce Server) se puedan ejecutar correctamente, los detalles del archivo, la carpeta o la conexión de BD que suministre como recursos globales deben ser compatibles con el entorno servidor. Por ejemplo, las rutas de acceso de los archivos y las carpetas deben usar la convención de Linux para rutas si la asignación se ejecuta en Linux. Asimismo, si define recursos globales como conexiones de BD, estas tienen que ser posibles en el equipo servidor.

Para más información consulte [Recursos globales](#) ³⁸.

Paquetes de taxonomías XBRL

Cuando implemente una asignación que remita a paquetes de taxonomías XBRL en FlowForce Server, MapForce recopila todas las referencias externas que existen en la asignación y las resuelve usando la configuración actual y los paquetes de taxonomías que están instalados en ese momento. Si hay referencias externas sin resolver que apunten a un paquete de taxonomías, dicho paquete se implementará junto con la asignación. FlowForce Server utilizará ese paquete para ejecutar la asignación. Para actualizar el paquete de taxonomías que utiliza FlowForce Server deberá cambiarlo primero en MapForce y volver a implementar la asignación.

Recuerde que el catálogo raíz de MapForce Server influye en el modo en que se resuelven las taxonomías en el equipo de destino. El catálogo raíz se encuentra en esta ruta de acceso relativa al directorio de instalación de MapForce Server: **etc/RootCatalog.xml**.

Los paquetes de taxonomías que se implementaron con la asignación se usarán si el catálogo raíz de MapForce Server no incluye ya dichos paquetes o no contiene un paquete que se definiera para el mismo prefijo URL. El catálogo raíz de MapForce Server tiene prioridad sobre la taxonomía que se implementó.

Si ejecuta MapForce Server de forma independiente (y no bajo control de FlowForce Server), tiene la opción de especificar qué catálogo raíz debe utilizar la asignación:

- En la línea de comandos se hace agregando la opción `-catalog` al comando `run`.
- En la API de MapForce Server se hace llamando al método `SetOption` y dando la cadena "catalog" como primer argumento y la ruta de acceso del catálogo raíz como segundo argumento.

Si la asignación utiliza componentes XBRL con bases de enlaces de tabla, entonces deberá aportar el paquete de taxonomías o el archivo de configuración del paquete de taxonomías a la asignación en tiempo de ejecución:

- En la línea de comandos de MapForce Server esto se hace agregando la opción `--taxonomy-package` o `--taxonomy-packages-config-file` al comando `run`.
- En la API de MapForce Server esto se hace llamando al método `SetOption`. El primer argumento debe ser "taxonomy-package" o "taxonomy-packages-config-file". El segundo argumento debe ser la ruta de acceso real del paquete de taxonomías (o del archivo de configuración del paquete de taxonomías).

3.2 Recursos globales

Los recursos globales de Altova son alias para recursos de archivo, carpeta y base de datos. Cada alias puede tener varias configuraciones y cada configuración se corresponde con un solo recurso. Esto quiere decir que puede alternar entre configuraciones al usar recursos globales. Por ejemplo, podría crear un recurso de base de datos con dos configuraciones: `desarrollo` y `producción`. Puede cambiar de una configuración a otra en función de sus necesidades. Por ejemplo, en MapForce podría recuperar la información tanto desde la base de datos de `desarrollo` como desde la de `producción`; para ello debe indicar, como parámetro de la línea de comandos y en tiempo de ejecución, qué configuración quiere usar.

Los recursos globales se pueden usar en distintas aplicaciones de Altova (*véase la lista más abajo*).

Recursos globales en otros productos de Altova

Una vez los haya guardado como recursos globales, puede usar esos archivos, carpetas o conexiones de base de datos en varias de las aplicaciones de Altova. Por ejemplo, si suele tener que abrir el mismo archivo (o carpeta o conexión de base de datos) en distintas aplicaciones de escritorio de Altova puede que le resulte más cómodo definirlo como un recurso global. Así, si quiere cambiar la ruta de acceso solo tiene que hacerlo en un sitio. Por el momento, puede definir y usar recursos globales en estos productos de Altova:

- [Altova Authentic](#)
- [DatabaseSpy](#)
- [MobileTogether Designer](#)
- [MapForce](#)
- [StyleVision](#)
- [XMLSpy](#)
- [FlowForce Server](#)
- [MapForce Server](#)
- [RaptorXML Server/RaptorXML+XBRL Server](#)

Para más información sobre cómo crear recursos globales consulte la parte "Recursos globales de Altova" en la documentación de MapForce.

Recursos en MapForce Server

Al compilar una asignación en un archivo de ejecución de MapForce Server (`.mfx`), cualquier referencia a los recursos globales que haya usado la asignación se mantiene sin resolver. Esto significa que debe suministrar estas referencias en el lado servidor para poder ejecutar correctamente la asignación. En concreto, en MapForce Server son necesarios los siguientes requisitos para ejecutar un archivo `.mfx` de datos que usa recursos globales:

1. *El archivo de definiciones de recursos globales.* En el equipo en el que se instala MapForce el archivo se llama `GlobalResources.xml` y puede encontrarlo en la carpeta `Documentos\Altova`. Puede copiar este archivo en el equipo en el que se ejecuta MapForce y también puede crear varios archivos si lo necesita.
2. *El nombre de la configuración de los recursos globales.* Cada recurso global tiene una configuración predeterminada. Por supuesto, también puede crear otras configuraciones.

En MapForce la ruta de acceso al archivo de definiciones de recursos globales y el nombre de la configuración de recursos globales se define o cambia desde la interfaz gráfica del usuario. En MapForce Server esta información se indica en tiempo de ejecución (*imagen siguiente*).

- Si ejecuta la asignación con la línea de comandos, use las opciones `--globalresourceconfig` y `--globalresourcefile` después de ejecutar el comando `run`, por ejemplo:

```
C:\Archivos de programa (x86)\Altova\MapForceServer2024\bin\MapForceServer.exe run
SomeMapping.mfx --globalresourcefile="C:
\Usuarios\\Documentos\Altova\GlobalResources.xml" --
globalresourceconfig="Default"
```


- Si ejecuta la asignación a través de la API de MapForce Server debe llamar dos veces al método `SetOptions` antes de llamar al método `Run`. La primera llamada es necesaria para indicar la ruta al archivo de recursos globales como opción y la segunda es necesaria para indicar el nombre de la configuración de recursos globales. Para más información consulte [API de MapForce Server](#)⁶².

3.3 Optimización por combinación

La optimización por combinación permite acelerar la ejecución de asignaciones de datos en las que se pueden filtrar o combinar conjuntos de datos de gran tamaño.

La optimización por combinación funciona eliminando bucles anidados que ocurren durante la ejecución de la asignación de datos. Los bucles anidados se producen cuando la asignación recorre cada elemento de un conjunto tantas veces como elementos existan en otro conjunto de datos. Recuerde que es normal que el motor* de ejecución de asignaciones efectúe bucles (iteraciones) en varias secuencias de elementos. Cuando se producen bucles anidados independientes (es decir, bucles que recorren otros bucles), la asignación puede favorecerse de la optimización por combinación, que permite reducir el tiempo necesario para ejecutar la asignación. Los bucles anidados casi no se perciben durante la ejecución de asignaciones cuyos datos de entrada no son de gran tamaño. Sin embargo, cuando se trata de asignaciones que procesan archivos o bases de datos compuestos por un gran número de registros, los bucles anidados pueden dar lugar a complicaciones.

* El motor de ejecución de una asignación puede ser MapForce, MapForce Server o código de programa C#, C++ o Java generado con MapForce. **La optimización por combinación es una característica exclusiva de MapForce Server Advanced Edition.**

Para designar MapForce Server como motor de ejecución de destino, haga clic en el botón **BUILT-IN**  de la barra de herramientas de MapForce. Esta opción también activa la gran mayoría de características de MapForce. Si selecciona otro lenguaje de transformación, ciertas características de MapForce pueden ser incompatibles.

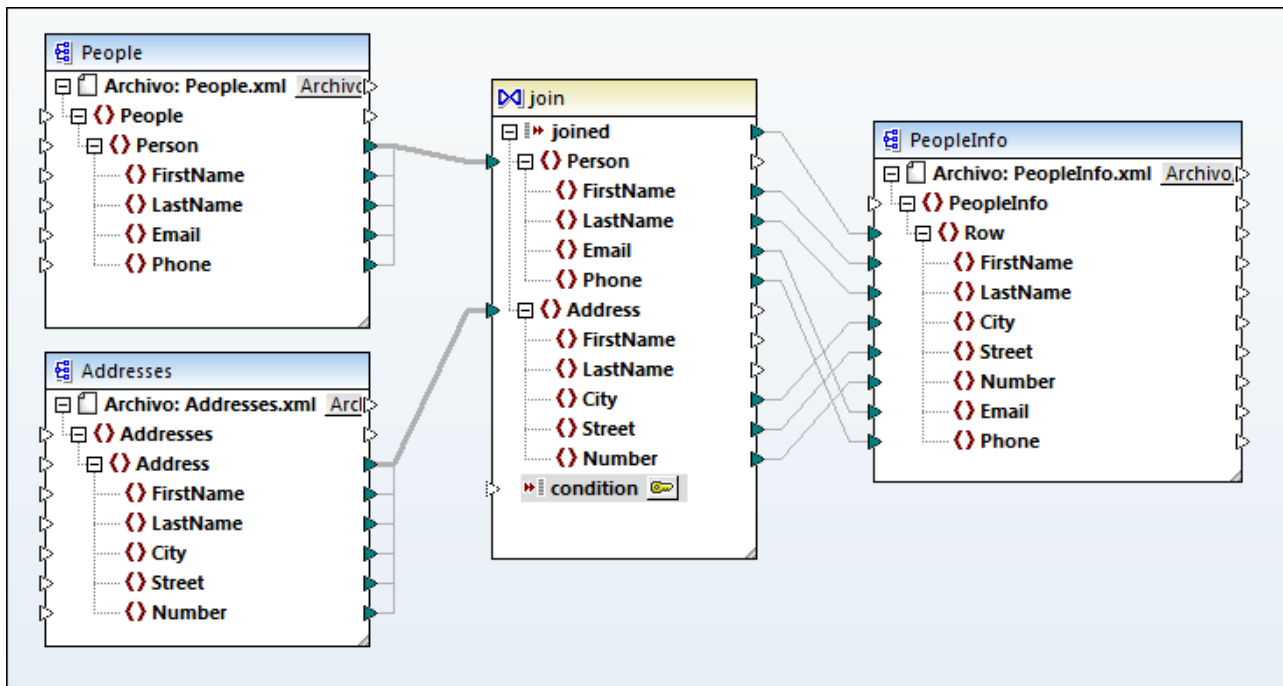
Según lo anterior, la prioridad de la optimización por combinación es ocuparse de los bucles anidados de forma eficaz. Para poder entenderlo es necesario comprender cómo se producen los bucles anidados.

La situación más común que da lugar a un bucle anidado es cuando la asignación contiene un componente de combinación y no es posible utilizar el modo de combinación JOIN SQL**.

** Cuando se dan ciertas condiciones en MapForce, las asignaciones admiten un modo de ejecución especial llamado "modo de combinación JOIN SQL" (esto solo es relevante si la asignación lee datos de una base de datos). Cuando los datos se combinan de esta forma, es la base de datos quien lleva a cabo la operación de combinación (es decir, se produce una combinación JOIN SQL) y se evitan los bucles anidados en el motor de ejecución de la asignación. Para más información sobre el modo de combinación JOIN SQL consulte la documentación de MapForce (<https://www.altova.com/es/documentation.html>).

Por ejemplo, a continuación puede ver una asignación diseñada con Altova MapForce que combina datos de dos archivos XML mediante un componente de combinación. En el equipo donde está instalado MapForce, esta asignación está en la ruta de acceso ..

\\Documentos\Altova\MapForce2024\MapForceExamples\Tutorial\JoinPeopleInfo.mfd. Los datos de algunas personas solamente están disponibles en el primer archivo XML (**Email, Phone**), mientras que otros datos solamente están en el segundo archivo XML (**City, Street, Number**). El objetivo de la asignación es escribir en el archivo XML de destino los datos combinados de todas las personas cuyos elementos **FirstName** y **LastName** sean idénticos en ambas estructuras.



JoinPeopleInfo.mfd

En MapForce el componente de combinación agrupa los elementos en dos conjuntos en base a determinada condición, lo cual implica comparar cada elemento del conjunto 1 con cada elemento del conjunto 2. El número total de comparaciones representa la combinación cruzada (producto cartesiano) de ambos conjuntos. Por ejemplo, si el primer conjunto contiene 50 elementos y el segundo contiene 100 elementos, entonces se producirá un total de 5000 comparaciones (50 x 100). En la asignación de la imagen anterior, los conjuntos que se comparan equivalen a todos los elementos de instancia de las dos estructuras XML que están conectadas al componente de combinación.

Nota: la optimización por combinación (característica exclusiva de MapForce Server Advanced Edition) no debe confundirse con los componentes de combinación (característica de Altova MapForce). Para más información sobre los componentes de combinación consulte la documentación de Altova MapForce (<https://www.altova.com/es/documentation.html>).

Desde el punto de vista del rendimiento, las asignaciones que contienen bucles anidados tardan más en ejecutarse. Imagine un caso en el que ambos conjuntos combinados contienen millones de registros. Esto afectaría lógicamente al rendimiento y la optimización por combinación sería de gran ayuda. Por lo general, la optimización por combinación funciona como cualquier motor de base de datos optimizado para consultar (indizar) conjuntos de datos de enorme tamaño. La excepción es que, como puede verse en la imagen anterior, la optimización por combinación no solo se ocupa de datos procedentes de bases de datos. La optimización por combinación elimina los bucles anidados, independientemente del tipo de datos, mediante la generación de tablas de consulta internas que se consultan en tiempo de ejecución. Esto mejora considerablemente el rendimiento de la asignación y en definitiva reduce el tiempo necesario para ejecutar la asignación.

Nota: cuando se produce la optimización por combinación, la ejecución de la asignación tardará menos pero necesitará más memoria. Tenga en cuenta que los patrones de uso de memoria dependen de varios factores complejos y, por tanto, el funcionamiento exacto dependerá de cada caso.

La optimización por combinación no solo puede acelerar las asignaciones de datos que incluyan combinaciones, sino también las que incluyan componentes de filtrado. En MapForce, los filtros procesan secuencias de elementos (es decir, comprueban si se cumple cierta condición binaria en cada instancia del elemento que está conectado a la entrada **node/row**). Si la condición binaria está conectada a una función que, a su vez, debe recorrer otra secuencia de elementos y si el contexto de asignación así lo requiere, entonces tendremos una situación parecida a cuando se produce una combinación. Si el filtro debe realizar una comparación cruzada de cada elemento de los dos conjuntos, entonces puede optimizarse por combinación.

Para poder favorecerse de la optimización por combinación, la asignación debe ejecutarse con MapForce Server Advanced Edition. Para ejecutar una asignación con MapForce Server Advanced Edition abra la asignación en MapForce y compílela en un archivo de ejecución (.mfx) (con el comando de menú **Archivo | Compilar en archivo de ejecución de MapForce Server**). Después ejecute el archivo .mfx con ayuda de un método de la API en el lenguaje que prefiera (o con el comando `run` de la interfaz de la línea de comandos). Consulte el apartado [How It Works](#)²⁹ para obtener más información.

3.4 Credenciales

Los objetos de credencial permiten usar la información de autenticación (como nombres de usuario, contraseñas o detalles de autenticación OAuth) de forma segura en varios entornos de ejecución de asignaciones. Es útil usar credenciales para asignaciones que requieren autenticación HTTP básica o autorización OAuth 2.0. Puede definir credenciales en MapForce y también en FlowForce Server. Las credenciales que defina en MapForce se pueden implementar en FlowForce Server de forma parecida a como se implementan las asignaciones.

Una vez haya compilado la asignación en un archivo de ejecución de MapForce Server (.mfx), MapForce Server lo ejecutará en función de las opciones que haya escogido al diseñar la asignación.

Si marcó la casilla *Incluir credencial cifrada en el archivo .mfx y en la implementación de la asignación* al crear la credencial en MapForce, al ejecutar la asignación MapForce Server usará cualquier credencial que haya almacenada en el archivo .mfx. Esto significa que puede ejecutar la asignación con un comando como este:

```
<exec> run mapping.mfx
```

Donde `<exec>` es la ruta de acceso al ejecutable de MapForce Server. Esta ruta puede ser absoluta o, si el directorio actual es el mismo que el del ejecutable, también puede introducir solamente el nombre del ejecutable.

Si solamente introdujo el nombre de la credencial (sin nombre de usuario y contraseña) en MapForce, entonces debe suministrar estos detalles explícitamente en el momento de ejecución con la opción de la línea de comandos `--credential` para el comando `run`. De esta manera puede, por ejemplo, usar en producción unas credenciales distintas a las que usó al diseñar la asignación. La opción `--credential` tiene la forma `--credential=KEY:VALUE`, donde

- `KEY` es el nombre de la credencial tal y como se definió en MapForce.
- `VALUE` puede ser una propiedad de credencial o una lista de propiedades separadas por el símbolo `&`. Las propiedades que se admiten para credenciales de tipo "contraseña" pueden ser `username` y `password`. Para credenciales de tipo OAuth 2.0 la única propiedad que se admite es `oauth:token`.
- Los valores que se vayan a usar para la propiedad se suministran igual que los parámetros de consulta en una URL, con el símbolo `"="`.

Por ejemplo:

```
<exec> run mapping.mfx --credential="mycredential:username=admin&password=4xJ38dnx7"
```

En el fragmento de código anterior, el valor de la opción `--credential` está entre comillas para tratar al valor de forma literal, ya que el nombre de usuario y la contraseña están separados por un símbolo `"&"`.

Si necesita varios pares de credenciales para ejecutar la asignación puede indicar la opción `--credential` varias veces.

Las credenciales suministradas como opciones de la línea de comandos tienen prioridad frente a las credenciales almacenadas.

Si no marcó la casilla *Incluir credencial cifrada en el archivo .mfx* y en la implementación de la asignación, entonces no aparecerán los campos de información sensible. Esto significa que debe suministrar la contraseña en la línea de comandos y hacer referencia a la credencial por su nombre, por ejemplo:

```
<exec> run mapping.mfx --credential=mycredential:password=4xJ38dnx7
```

La información de estos campos se considera información sensible:

- **Contraseña** (para credenciales de tipo "contraseña")
- **Secreto del cliente, Token de acceso y Actualizar token** (para credenciales de tipo "OAuth 2.0")

En el caso de las asignaciones que necesitan OAuth 2.0, la línea de comandos de MapForce Server admite un token de acceso como entrada en el momento de ejecutar la asignación. Si va a trabajar con la línea de comandos debe obtener el token de acceso OAuth 2.0 de forma externa (por ejemplo, solicitándolo a MapForce), ya que la línea de comandos de MapForce Server no cuenta con una IGU. Sin embargo, si trabaja con MapForce bajo las órdenes de FlowForce Server, este último puede obtener por sí mismo un token de acceso OAuth 2.0 nuevo en tiempo de ejecución.

Se puede ejecutar la asignación en la línea de comandos con credenciales almacenadas siempre que el token OAuth 2.0 almacenado no haya expirado ni haya sido revocado por el proveedor de servicios web. Para usar esta opción debe suministrar un token de acceso OAuth 2.0 nuevo (obtenido por medios externos) usando `--credential`, por ejemplo:

```
<exec> run mapping.mfx --  
credential=my_oauth_credential:oauth:token=jdsaf1kajlkewsaiurthczv904215-jhd
```

Donde:

- `my_oauth_credential` es el nombre de la credencial OAuth creada desde MapForce.
- `oauth:token` es la forma de indicar a MapForce Server que se suministrará un token de acceso OAuth 2.0 nuevo en tiempo de ejecución.

API de MapForce Server

La API de MapForce Server permite crear credenciales, añadir propiedades a credenciales y cerrar credenciales una vez haya terminado de declararlas. El fragmento de código siguiente ilustra la forma más común de declarar credenciales de contraseña en un programa en C# que ejecuta una asignación:

```
//Crear un objeto de MapForce Server  
Altova.MapForceServer.Server objMFS = new Altova.MapForceServer.Server();  
// Definir el nombre de la credencial como se definió en MapForce  
objMFS.BeginCredential("mycredential");  
// Añadir las propiedades de la credencial  
objMFS.AddCredentialProperty("username", "altova");  
objMFS.AddCredentialProperty("password", "b45ax78!");  
// Cerrar la credencial  
objMFS.EndCredential();
```

Para llevar a cabo autorizaciones OAuth 2.0 desde un programa que ejecuta una asignación, el nombre de la propiedad de la credencial se debe definir como `oauth:token`, como se ilustra a continuación:

```
//Crear un objeto de MapForce Server  
Altova.MapForceServer.Server objMFS = new Altova.MapForceServer.Server();  
// Definir el nombre de la credencial como se definió en MapForce  
objMFS.BeginCredential("my_oauth_credencial");  
// Añadir las propiedades de la credencial  
objMFS.AddCredentialProperty("oauth:token", "jdsaf1kajlkewsaiurthczv904215-jhd");  
// Cerrar la credencial  
objMFS.EndCredential();
```

Si una asignación necesita varios conjuntos de credenciales, use los métodos que hemos descrito más arriba para añadir cuantos conjuntos de credenciales necesite. Una vez haya declarado todas las credenciales, puede ejecutar el archivo de ejecución de la asignación como siempre, es decir llamando al método `Run()`. Para más información consulte la [referencia de la API](#)⁶².

3.5 Autenticación dinámica

En MapForce se pueden configurar asignaciones que hagan llamadas a servicios web para autenticación HTTP básica. Una de las formas de hacer esto es con la autenticación dinámica, que es una alternativa a las credenciales. La autenticación dinámica consiste en diseñar la asignación de manera que acepte el nombre de usuario y la contraseña como parámetros de entrada. Para más detalles sobre cómo configurar la autenticación dinámica consulte la documentación de MapForce (<https://www.altova.com/es/documentation>).

Si configuró la asignación para que funcione con autenticación dinámica, entonces el nombre de usuario y la contraseña correspondientes se deben introducir como parámetros en el momento de la ejecución. El proceso es el mismo que cuando introduce cualquier otro tipo de parámetro en la asignación. Por ejemplo, si llama a MapForce Server con la línea de comandos la sintaxis de una asignación como la anterior es:

```
<exec> run mapping.mfx --p=username:admin --p=password:dj9JaVax
```

Donde:

- **<exec>** es la ruta de acceso al ejecutable de MapForce Server. Esta ruta puede ser absoluta o relativa o, si el directorio actual es el mismo que el del ejecutable, también puede introducir solamente el nombre del ejecutable.
- **username** y **password** son los nombres de los parámetros de entrada respectivos en la asignación de MapForce.

Al llamar a la API de MapForce Server puede autenticar la asignación llamando al método `AddParameter` antes de llamar al método `Run`. Por ejemplo, en C# el código equivalente tendría este aspecto:

```
try
{
    Altova.MapForceServer.Server mfs = new Altova.MapForceServer.Server();
    mfs.AddParameter("username", "admin");
    mfs.AddParameter("password", "dj9JaVax");
    mfs.WorkingDirectory = "C:\\Work";
    if(mfs.Run("C:\\Work\\mapping.mfx"))
    {
        Console.WriteLine("Success");
    }
    else
    {
        Console.WriteLine(mfs.LastExecutionMessage);
    }
    Console.ReadLine();
}
catch(Exception ex)
{
    Console.WriteLine(ex);
}
```

Aquí también se debe usar los mismos **username** y **password** (primer argumento para el método `AddParameter`) que los nombres de los parámetros de entrada correspondientes en la asignación de MapForce.

4 Línea de comandos de MapForce Server

MapForce Server cuenta con una interfaz de la línea de comandos que puede usar para tareas administrativas como asignar licencias o ejecutar archivos de ejecución de asignaciones (.mfx). A continuación enumeramos los comandos disponibles.

- [assignlicense](#)⁵⁰: (solo para Windows) carga una licencia en LicenseServer y la asigna a MapForce Server.
- [exportresourcestrings](#)⁵¹: exporta todas las cadenas de recursos de la aplicación a un archivo XML.
- [help](#)⁵³: muestra información sobre el comando que se da como argumento (o sobre todos los comandos si no se da ningún argumento).
- [licenseserver](#)⁵⁴: registra MapForce Server con un LicenseServer de la red local.
- [run](#)⁵⁵: ejecuta un archivo de ejecución de asignación compilado con MapForce.
- [setdeflang](#)⁵⁹: establece el idioma predeterminado de MapForce Server.
- [verifylicense](#)⁶⁰: (solo para Windows) comprueba si la versión actual de MapForce Server tiene asignada una licencia o, en su caso, si la clave de licencia dada es la que corresponde.
- [version](#)⁶¹: muestra el número de versión de MapForce Server.

Para llamar a MapForce Server desde la línea de comandos necesita saber la ruta de acceso al ejecutable, que es distinta según el sistema operativo que use. Por defecto, el ejecutable de MapForce Server está instalado en:

<i>Linux</i>	/opt/Altova/MapForceServer2024/bin/mapforceserver
<i>macOS</i>	/usr/local/Altova/MapForceServer2024/bin/mapforceserver
<i>Windows</i>	C:\Archivos de programa\Altova\MapForceServer2024\bin\MapForceServer.exe

Nota: Si instala la versión de MapForce Server de 32 bits en una versión de Windows de 64 bits, cambie `c:\Archivos de programa` a `C:\Archivos de programa (x86)`.

Por convención, en esta documentación usamos una forma abreviada de la ruta de acceso del ejecutable al describir los comandos y usamos `mapforceserver` en lugar del nombre del ejecutable, por ejemplo:

```
mapforceserver help
```

Donde `mapforceserver` es la ruta de acceso o el nombre del ejecutable. Observe que si usa una ruta absoluta podrá ejecutar comandos independientemente del directorio actual en el que esté su ventana (terminal) de la línea de comandos. Sin embargo, si quiere llamar al ejecutable tecleando su nombre, primero debe seguir uno de estos pasos:

- Cambie el directorio actual de la terminal al directorio de instalación de MapForce Server.
- Añada el directorio en el que está el ejecutable a la variable de entorno PATH.

A continuación describimos estos dos escenarios con más detalle.

Consejos

Si no está familiarizado con la línea de comandos, estos son algunos consejos que le pueden ayudar:

- Para averiguar el directorio en el que está actualmente la ventana de la línea de comandos introduzca `pwd` en Linux y en MacOS. En Windows debe introducir `echo %CD%`.
- Puede usar la tecla **Tabulador** para introducir varias rutas de acceso a archivos o directorios sin tener que escribirlos por completo. Por ejemplo, si teclea `cd c:\prog` en la línea de comandos y pulse **Tabulador** para que el comando se complete automáticamente a `C:\Archivos de programa` (o a otro directorio de C:\ que comience por "Prog").
- Al introducir en Windows una ruta que contiene espacios en blanco, como `C:\Archivos de programa`, debe escribirlos entre comillas.
- Si ve un mensaje parecido a "No se reconoce este comando como comando interno o externo" es probable que haya escrito mal alguna ruta o algún comando.
- En Linux debe asegurarse de que escribe la ruta o el nombre del directorio con mayúsculas si es que las contienen. Por ejemplo, si escribe la ruta `/home/nikita/downloads` verá que aparece un error si el nombre del directorio en realidad es `/home/nikita/Downloads`.
- Al escribir una ruta en Linux o en macOS debe usar barra en lugar de barra invertida, como hace en Windows.

Ejecutar un comando

1. Abra una ventana de la línea de comandos.
 - a. Para abrir la línea de comandos en Windows, pulse la tecla **Windows** y escriba `cmd`. Haga clic en la sugerencia que aparece: **Símbolo del sistema**.
 - b. Para abrirla en Mac, haga clic en el icono **Finder** y en el menú seleccione **Ir > Utilidades**. Haga doble clic en el icono **Terminal** de la ventana Utilidades.
 - c. Si usa Linux con una interfaz gráfica del usuario, elija y ejecute el comando **Terminal** según su distribución de Linux. Si ejecuta Linux desde la línea de comandos puede ignorar este paso.
2. Introduzca la ruta de acceso completa al ejecutable, seguida del comando que quiere ejecutar. Por ejemplo, el comando siguiente abre la ayuda en la línea de comandos.

<i>Linux</i>	<code>/opt/Altova/MapForceServer2024/bin/mapforceserver help</code>
<i>macOS</i>	<code>/usr/local/Altova/MapForceServer2024/bin/mapforceserver help</code>
<i>Windows</i>	<code>C:\Archivos de programa (x86)\Altova\MapForceServer2024\bin\MapForceServer.exe help</code>

En el ejemplo anterior el comando `help` se ejecuta sin más opciones o argumentos. Otros comandos pueden tener argumentos y opciones, que pueden ser obligatorios u opcionales. Por ejemplo, el comando `run` tiene un argumento obligatorio que permite indicar la ruta de acceso o el nombre de archivo del archivo `.mfx` que quiere ejecutar. Consulte la sección de referencia para más detalles sobre cada uno de los comandos.

Llamar a MapForce Server en el directorio de instalación

Para llamar al ejecutable sin tener que teclear toda la ruta, cambie el directorio actual por el directorio en el que está instalado MapForce Server, por ejemplo:

<i>Linux</i>	<code>cd /opt/Altova/MapForceServer2024/bin</code>
<i>macOS</i>	<code>cd /usr/local/Altova/MapForceServer2024/bin</code>

<i>Windows</i>	<code>cd C:\Archivos de programa (x86)\Altova\MapForceServer2024\bin</code>
----------------	---

Ahora para ejecutar cualquier comando solo tiene que teclear el nombre del ejecutable, por ejemplo:

<i>Linux</i>	<code>./mapforceserver help</code>
<i>macOS</i>	<code>./mapforceserver help</code>
<i>Windows</i>	<code>MapForceServer.exe help</code>

Nota: En Linux y macOS el prefijo `./` indica que el ejecutable está en el directorio actual.

Llamar a MapForce Server desde cualquier directorio

Para llamar al ejecutable desde cualquier directorio debe hacer referencia a este con una ruta absoluta. Otra alternativa para llamar al programa es teclear solamente el nombre del ejecutable si primero edita la variable de entorno de su sistema operativo PATH para que incluya la ruta completa de acceso al directorio de instalación de MapForce Server. Hay varias maneras de cambiar la variable PATH, por ejemplo:

Nota: Una vez modificada la variable de entorno PATH, para que los cambios surtan efecto puede que sea necesario cerrar la ventana de la línea de comandos y abrir una nueva.

4.1 assignlicense

Sintaxis y descripción

Al ejecutarse el comando `assignlicense` carga el archivo de licencia indicado por el argumento `ARCHIVO` en el servidor `LicenseServer` con el que está registrado MapForce Server (véase el comando `licenseserver`) y asigna esa licencia a MapForce 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.

```
mapforceserver assignlicense [opciones] ARCHIVO
```

- 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 MapForce Server.

Para más información sobre el proceso de asignación de licencias consulte la documentación de Altova `LicenseServer` (<https://www.altova.com/manual/es/licenseserver/3.14/>).

Ejemplos

Ejemplos del comando `assignlicense`:

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

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

```
mapforceserver exportresourcestrings [opciones] CódigoIdioma 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.

Ejemplos

Ejemplos del comando `exportresourcestrings`:

```
mapforceserver 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 **MapForce Server** en inglés.

Localizar MapForce Server en otros idiomas

Si quiere puede localizar MapForce 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:\Archivos de programa (x86)\Altova\MapForceServer2024\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 MapForce 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 programa (x86)\Altova\MapForceServer2024\bin`. El DLL tendrá un nombre similar a este `MapForceServer2024_ci.dll`. La parte `_ci` del nombre contiene el código del idioma. Por ejemplo, en `MapForceServer2024_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 MapForce Server predeterminada. Use el código de idioma del nombre del archivo DLL como argumento del comando `setdeflang`

Nota: Altova ya ofrece MapForce 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 MapForce Server.

4.3 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.

```
mapforceserver help Command
```

Ejemplo

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

```
mapforceserver 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:

```
mapforceserver licenseserver --help
```

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

```
mapforceserver help licenseserver
```

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

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

4.4 licenseserver

Sintaxis y descripción

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

```
mapforceserver 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.

Cuando MapForce 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 MapForce Server. Consulte la documentación de Altova LicenseServer para obtener más información (<https://www.altova.com/manual/es/licenseserver/3.14/>).

Ejemplo

Ejemplo del comando `licenseserver`:

```
mapforceserver licenseserver DOC.altova.com
mapforceserver licenseserver localhost
mapforceserver 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 MapForce 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.

4.5 run

El comando `run` ejecuta un archivo de ejecución de MapForce Server (archivo `.mfx`) suministrado como argumento. El archivo de ejecución de MapForce Server se crea con MapForce; este archivo representa una asignación compilada para su ejecución en servidor.

Los archivos de entrada que necesita la asignación se espera que estén en la ruta indicada al diseñar la asignación en MapForce. Si MapForce Server no se ejecuta en el mismo sistema operativo que MapForce, entonces los archivos de entrada para la asignación se deben copiar en el equipo de destino junto con el archivo `.mfx` y después se debe hacer referencia a ellos con una ruta relativa. Para información sobre cómo configurar una asignación con respecto a rutas relativas o absolutas, consulte la documentación de MapForce (<https://www.altova.com/es/documentation#mapforce>). Puede que existan otros requisitos en función de cómo haya diseñado la asignación (véase [Preparar asignaciones para ejecución en servidor](#)³²).

Si la asignación devuelve un valor simple, como una cadena, este resultado se escribe en la secuencia `stdout` (resultado estándar). Los mensajes de éxito y error, por su parte, se encuentran en la secuencia `stderr` (error estándar). Si no quiere que la secuencia de resultado estándar se muestre en pantalla junto con el mensaje de éxito o error puede redirigir la secuencia de resultado estándar o la secuencia de error estándar (o ambas) a archivos. Si no las redirige, las secuencias `stdout` y `stderr` se mostrarán juntas en pantalla.

Por ejemplo, para redirigir la secuencia de resultado estándar a un archivo, use:

```
mapforceserver run MyMapping.mfx > MyOutput.txt
```

Para redirigir la secuencia de error estándar a un archivo, use:

```
mapforceserver run MyMapping.mfx 2> Diagnostics.log
```

Para redirigir las dos secuencias al mismo tiempo, use:

```
mapforceserver run MyMapping.mfx > MyOutput.txt 2> Diagnostics.log
```

Para más información sobre cómo redirigir secuencias consulte la documentación de la shell de comandos de su sistema operativo.

Sintaxis

Windows `MapForceServer run [opciones] MfxFile`

Linux `mapforceserver run [opciones] MfxFile`

Mac `mapforceserver run [opciones] MfxFile`

Opciones

	<code>--catalog</code>	<p>Especifica la ruta absoluta a un archivo de catálogo raíz que no es el archivo de catálogo raíz instalado. El valor por defecto es la ruta absoluta al archivo de catálogo raíz instalado.</p> <p>Formulario: <code>--catalog=FILE</code></p>
<code>--cert</code>	<code>--certificatespath</code>	<p>Esta opción se puede aplicar si MapForce se está ejecutando en Linux. Indica la ruta de acceso al directorio donde se almacenan los archivos de certificado requeridos por la asignación.</p> <p>Formulario: <code>--certificatespath=DIRECTORY</code></p> <p>Véase también el apartado Gestión de certificados digitales ¹²⁵.</p>
<code>--cred</code>	<code>--credential=KEY:VALUE</code>	<p>Esta opción se puede usar si la asignación contiene credenciales que definió al diseñar la asignación en MapForce. KEY es el nombre de la credencial como se definió en MapForce. VALUE consiste en uno o más pares nombre-valor, por ejemplo:</p> <p><code>name1=value1&name2=value2</code></p> <p>Donde name es cualquiera de estas opciones:</p> <ul style="list-style-type: none"> • <code>username</code> • <code>password</code> • <code>oauth:token</code> <p>Si usa varios pares nombre-valor separados por el símbolo & debe usar <code>KEY:VALUE</code> entre comillas. Para más información consulte Credenciales ⁴³.</p>
<code>--gc</code>	<code>--globalresourceconfig</code>	<p>Esta opción se puede usar si la aplicación consume recursos globales ³⁸. Indica el nombre de la configuración de los recursos globales (alias) y se debe usar junto con la opción <code>--globalresourcefile</code>.</p> <p>Formulario: <code>--gc=VALUE</code></p>
<code>--gr</code>	<code>--globalresourcefile</code>	<p>Esta opción se puede usar si la aplicación consume recursos globales ³⁸. Indica la ruta de acceso al archivo de definición de los recursos globales y se debe usar junto con la opción <code>--globalresourceconfig</code>.</p> <p>Formulario: <code>--gr=FILE</code>.</p>
<code>--l</code>	<code>--lang</code>	<p>El lenguaje usado para mostrar mensajes.</p> <p>Formulario: <code>--lang=VALUE</code> (<code>en,de,ja,es,fr</code>)</p>
<code>--p</code>	<code>--param</code>	<p>Esta opción se puede aplicar si la asignación se diseñó para aceptar parámetros de entrada. Asigna un valor a un parámetro definido en la asignación. Formulario <code>--param=ParamName:ParamValue</code>. Antes de cada parámetro se debe usar el conmutador <code>--param</code>. Use comillas si</p>

		<p><i>ParamName</i> o <i>ParamValue</i> contienen algún espacio. Por ejemplo: <code>--p=company: "Nanonull Inc"</code>.</p> <p>Para más información sobre asignaciones que aceptan parámetros de entrada consulte la documentación de MapForce (https://www.altova.com/es/documentation#mapforce).</p>
	<code>--taxonomy-package</code>	<p>Indica la ruta de acceso absoluta a un paquete de taxonomías XBRL según se describe en la recomendación Taxonomy Packages 1.0. El valor de FILE suministra la ubicación del paquete de taxonomía. Añada la opción varias veces para indicar más de un paquete de taxonomía.</p> <p>Formulario: <code>--taxonomy-package=FILE</code></p>
	<code>--taxonomy-packages-config-file</code>	<p>Indica la ruta de acceso a un archivo de configuración llamado TaxonomyPackagesConfig.json que se usa para cargar paquetes de taxonomía XBRL. Este archivo de configuración se actualiza cada vez que añade, elimina, activa o desactiva paquetes de taxonomías XBRL desde la interfaz gráfica de los productos de Altova XMLSpy, MapForce o StyleVision. Si añadió paquetes de taxonomías XBRL personales con uno de esos productos el archivo está en C:</p> <p>\Users\<username>\Documents\Altova.</p> <p>Formulario: <code>--taxonomy-packages-config-file=FILE</code></p>

Ejemplos

Este ejemplo muestra cómo ejecutar el archivo de ejecución de una asignación (.mfx) con MapForce en Windows. La asignación usada en este ejemplo lee un archivo de entrada, **Employees.xml**, y produce dos archivos de salida (**PersonList.xml** y **Contacts.xml**).

Primero, genere el archivo de ejecución de MapForce Server (.mfx):

1. Ejecute MapForce y abra el siguiente archivo de diseño de MapForce (archivo .mfd): **C:\Users\<user>\Documents\Altova\MapForce2024\MapForceExamples\ChainedPersonList.mfd**.
2. En el menú **Archivo** haga clic en **Compilar en archivo de ejecución de MapForce Server**.
3. Cuando el programa lo solicite, guarde el archivo .mfx en el directorio **C:\temp**. Este es el directorio de trabajo donde MapForce Server ejecuta la asignación.

A continuación abra la línea de comandos y cambie el directorio de trabajo a **C:\temp**.

```
cd C:\temp
```

Por último, ejecute este comando para ejecutar **ChainedPersonList.mfx**. En este ejemplo se llama a MapForce Server con una ruta absoluta. (Para llamarlo con una ruta relativa debe añadir la ruta de acceso del ejecutable a la variable de entorno `PATH` de su sistema).

```
"C:\Program Files (x86)\Altova\MapForceServer2024\bin\MapForceServer.exe" run
```

```
ChainedPersonList.mfx
```

Los dos archivos de salida (**PersonList.xml** y **Contacts.xml**) se generan en el directorio de trabajo. Es importante destacar que esta asignación está configurada para usar rutas absolutas, razón por la cual esta asignación se ejecutó exitosamente y no necesitó que el archivo **Employees.xml** de entrada existiera en el directorio de trabajo. El archivo **Employees.xml** existe en la carpeta MapForce Examples que hemos mencionado antes. Para hacer referencia a él se usa una ruta absoluta. Para indicar si las rutas deben tratarse como absolutas o relativas haga clic con el botón derecho en la asignación en MapForce, seleccione **Configurar asignación** y después active o desactive la casilla *Convertir las rutas de acceso en absolutas en el código generado*. Siempre que cambie la configuración de la asignación debe asegurarse de que vuelve a compilar la asignación a .mfx. Para más información consulte el apartado [Preparar asignaciones para ejecución en servidor](#)³².

4.6 setdeflang

Sintaxis y descripción

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

```
mapforceserver setdeflang [opciones] CódigoIdioma
```

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

Ejemplo

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

```
mapforceserver sdl de
mapforceserver setdeflang es
```

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

Opciones

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

4.7 verifylicense

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.

```
mapforceserver verifylicense [opciones]
```

- Para comprobar si una licencia en concreto está asignada a MapForce 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/licenseserver/3.14/>).

Ejemplos

Ejemplos del comando `verifylicense`:

```
mapforceserver verifylicense
mapforceserver verifylicense --license-key=ABCD123-ABCD123-ABCD123-ABCD123-ABCD123-ABCD123
```

- El primer comando comprueba si MapForce Server tiene licencia.
- El segundo comando comprueba si MapForce 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 [l]

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

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

4.8 version

Sintaxis y descripción

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

```
mapforceserver version
```

Ejemplo

Ejemplo del comando `version`:

```
mapforceserver version
```

5 API de MapForce Server

MapForce 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 cada una de estas tres plataformas:

- [Interfaz .NET](#) ⁶³
- [Interfaz COM](#) ⁷¹
- [Interfaz Java](#) ⁸⁰

Por último, puede consultar la descripción técnica de la API en estos apartados:

- [Referencia de la API \(.NET y COM\)](#) ⁹³
- [Referencia de la API \(Java\)](#) ¹¹¹

5.1 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.MapForceServer`.

Durante el proceso de instalación, MapForce Server se registra automáticamente como objeto de servidor COM, por lo que no es necesario registrarlo a mano.

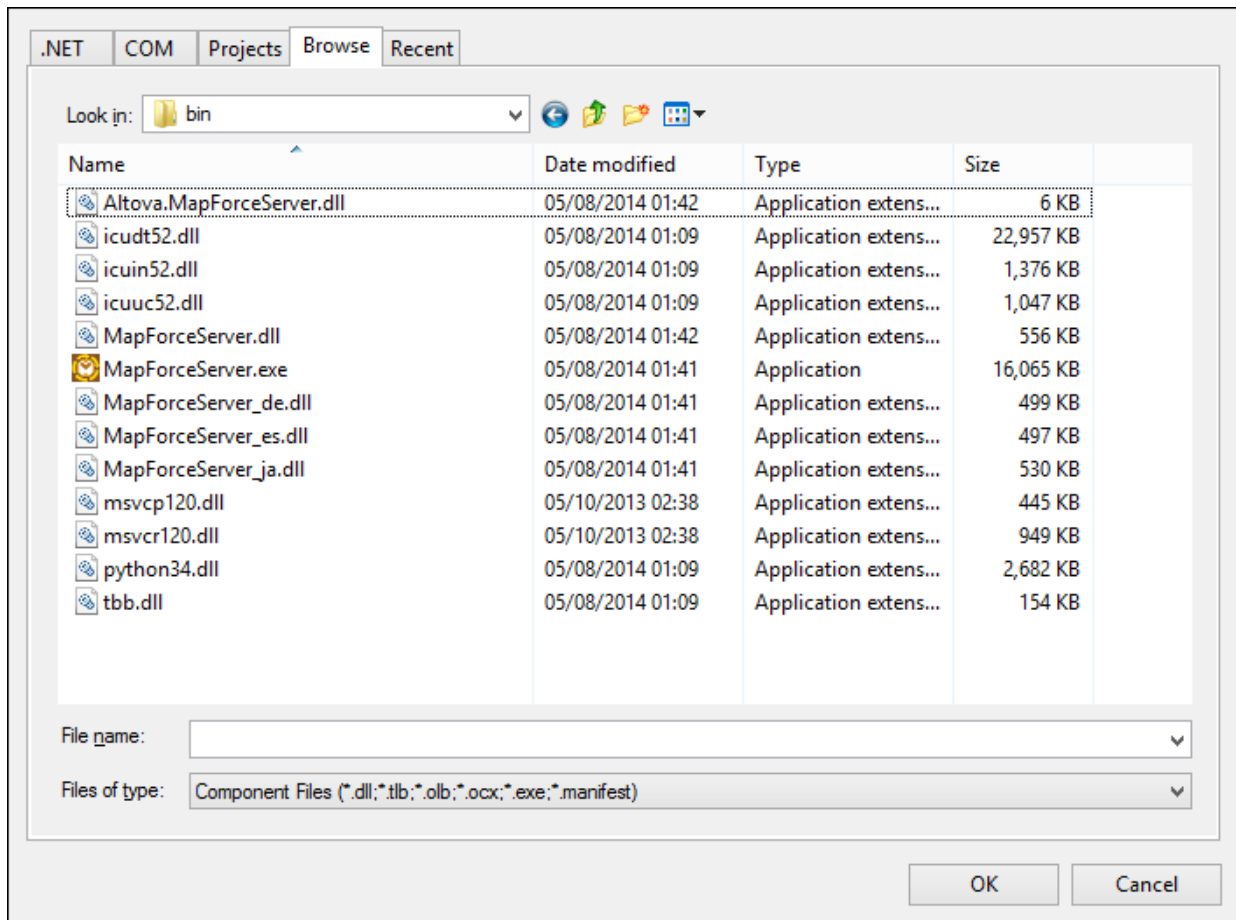
Para poder usar MapForce Server en su proyecto .NET basta con añadir una referencia al archivo `Altova.MapForceServer.dll` (*ver más abajo*). El archivo `Altova.MapForceServer.dll` está en la carpeta `bin` de la carpeta de instalación de MapForce Server. Este archivo DLL se añade automáticamente al caché global de ensamblados (GAC) durante la instalación de MapForce Server.

Nota: si instaló una versión de 64 bits de MapForce Server, los archivos de la versión de 32 bits de `Altova.MapForceServer.dll` estarán en la carpeta `bin\API_32bit`. Ocurre igual al contrario, si instaló una versión de 32 bits de MapForce Server, los archivos de la versión de 64 bits de `Altova.MapForceServer.dll` estarán en la carpeta `bin\API_64bit`.

Nota: antes de la versión 4.0 de .NET Framework, el GAC estaba ubicado en el directorio `%windir%\assembly`. A partir de la versión 4.0 de .NET Framework, el GAC se encuentra en el directorio `%windir%\Microsoft.NET\assembly`. La parte `%windir%` representa el directorio del sistema operativo Windows, que suele ser `C:\Windows`.

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

1. Abra el proyecto .NET en Visual Studio y haga clic en **Proyecto | Agregar referencia**.



- Abra la pestaña *Examinar*, navegue hasta la carpeta *<carpeta de aplicación de MapForceServer>/bin*, seleccione el archivo *Altova.MapForceServer.dll* y haga clic en **Aceptar**.

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

5.1.1 Ejemplo en C#

Este ejemplo muestra cómo ejecutar un archivo de ejecución de MapForce (.mfx) desde código C#. En Windows, los archivos de ejemplo están en: **C:\Program Files\Altova\MapForceServer2024\etc\Examples**.

Requisitos

- MapForce Server está instalado y tiene asignada una licencia válida.
- Si crea un proyecto de Visual Studio nuevo, añada una referencia al ensamblado de MapForce Server (véase [Interfaz .NET](#)⁶³). Puede saltarse este paso si está ejecutando el ejemplo existente de la API de MapForce, ya que el ejemplo ya contiene esa referencia.
- En el menú **Compilar** de Visual Studio, haga clic en **Administrador de configuración** y escoja la plataforma correcta, por ejemplo **Debug | x86** (o **Debug | x64**, en su caso). No use **Any CPU** como plataforma.
- Si tiene instalada la versión de 64 bits de MapForce Server, la aplicación que llama a la API (como la

del ejemplo que aparece más abajo) deberá estar diseñada para la plataforma de 64 bits en Visual Studio. Además, la ruta de acceso del ejecutable de MapForce Server también debe adaptarse según corresponda en el código.

La solución de ejemplo está en el directorio "Archivos de programa", que necesita derechos de administrador. Puede ejecutar Visual Studio como administrador o copiar la solución en una carpeta distinta que no necesite derechos de administrador.

Ejecutar el código de la asignación

El código siguiente ejecuta tres archivos de ejecución de servidor (.mfx). La tabla siguiente enumera los archivos de entrada que espera cada archivo .mfx, así como la salida que se genera tras la ejecución.

Archivo de ejecución (.mfx)	Entrada	Salida
TokenizeString.mfx	AltovaTools.xml	AltovaToolsFeatures.csv
SimpleTotal.mfx	ipo.xml	<i>String</i>
ClassifyTemperatures.mfx	Temperatures.xml	Temperatures_out.xml

Si tiene Altova MapForce puede examinar las asignaciones originales a partir de las cuales se crearon los archivos .mfx para entenderlas mejor. Se trata, respectivamente, de **TokenizeString1.mfd**, **SimpleTotal.mfd** y **ClassifyTemperatures.mfd**. Puede encontrar estas asignaciones en el directorio: **C:\users\<user>\Altova\MapForce2024\MapForceExamples**.

El ejemplo siguiente:

- Crea una instancia nueva de `Altova.MapForceServer.Server`, que es el objeto con el que trabajará de ahora en adelante.
- Define el directorio de trabajo en el que se lleva a cabo la ejecución. Se espera que los archivos de entrada estén en este directorio si hace referencia a ellos con una ruta relativa. Los archivos de salida también se crean en este directorio (*véase la tabla anterior*).
- Ejecuta **TokenizeString.mfx**. La ruta de acceso al archivo se da como argumento para el método `Run` (tenga en cuenta que la ruta es relativa al directorio de trabajo que se definió antes). Si el archivo se ejecuta correctamente, en el directorio de trabajo se crea un archivo .csv que representa el resultado de la asignación.
- Ejecuta **SimpleTotal.mfx**. En este caso la ruta también es relativa al directorio de trabajo. Esta asignación produce una cadena de salida; para obtenerla debe llamar al método `GetOutputParameter`.
- Ejecuta **ClassifyTemperatures.mfx**. Esta asignación espera un parámetro como entrada; puede indicarlo con el método `AddParameter`.

```
namespace MapForceServerAPI_sample
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
```

```

        // Create a MapForce Server object
        Altova.MapForceServer.Server objMFS = new Altova.MapForceServer.Server();

        // Set a working directory - used as a base for relative paths (you may
        need to adapt the path to the installation folder)
        objMFS.WorkingDirectory = "..\\..\\..";

        // Default path to the MapForce Server executable is the installation
        path (same dir with the MapForceServer.dll)
        // In case you moved the binaries on the disk, you need to explicitly set
        the path to the .exe file
        // objMFS.ServerPath = "C:\\Program Files (x86)\\Altova\\
        \\MapForceServer2020\\bin\\MapForceServer.exe";
        // objMFS.ServerPath = "C:\\Program Files\\Altova\\MapForceServer2020\\
        \\bin\\MapForceServer.exe";

        System.Console.WriteLine("Running " + objMFS.ProductNameAndVersion + ".
        \\n");

        // Set global resource file and configuration, if your mapping uses
        global resources
        // objMFS.SetOption( "globalresourcefile", "GlobalResources.xml" ); //
        "gr" can be used as short name for "globalresourcefile"
        // objMFS.SetOption( "globalresourceconfig", "Default" ); // "gc" can be
        used as short name for "globalresourceconfig"

        //
        -----
        // An example with input and output paths stored inside the MFX file
        System.Console.WriteLine("\\nExecuting TokenizeString.mfx...");
        if (objMFS.Run("TokenizeString.mfx"))
            System.Console.WriteLine("Successfully generated file
        'AltovaToolFeatures.csv'.");
        else
        {
            // execution failed. maybe no write permissions in working directory?
            Run this program as administrator.
            System.Console.WriteLine(objMFS.LastExecutionMessage);
        }

        //
        -----
        // An example creating a simple output so that we can retrieve the result
        explicitly
        System.Console.WriteLine("\\nExecuting SimpleTotal.mfx...");
        if (objMFS.Run("SimpleTotal.mfx"))
            System.Console.WriteLine("Mapping result is: " +
        objMFS.GetOutputParameter("total"));
        else
        {
            // execution failed (e.g. somebody deleted file ipo.xml)
            System.Console.WriteLine(objMFS.LastExecutionMessage);
        }

        //
    
```

```
-----  
        // an example with parameterized input  
        // the default of 'lower = 5' gets changed to the value '10'  
        // mfx reads file Temperatures.xml and writes its output to  
Temperatures_out.xml.  
        System.Console.WriteLine("\nExecuting ClassifyTemperatures.mfx with  
parameter 'lower' set to '10' ...");  
        objMFS.AddParameter("lower", "10");  
        if (objMFS.Run("ClassifyTemperatures.mfx"))  
            System.Console.WriteLine("File Temperatures_out.xml has been written  
successfully.");  
        else  
        {  
            // execution failed. maybe no write permissions in working directory?  
Run this program as administrator.  
            System.Console.WriteLine(objMFS.LastExecutionMessage);  
        }  
    }  
    catch (System.Runtime.InteropServices.COMException ex)  
    {  
        System.Console.WriteLine("Internal Error - " + ex.Message);  
    }  
}  
}
```

5.1.2 Ejemplo en Visual Basic .NET

Este ejemplo muestra cómo ejecutar un archivo de ejecución de MapForce (.mfx) desde código Visual Basic. En Windows, los archivos de ejemplo están en: **C:\Program Files\Altova\MapForceServer2024\etc\Examples**.

Requisitos

- MapForce Server está instalado y tiene asignada una licencia válida.
- Si crea un proyecto de Visual Studio nuevo, añada una referencia al ensamblado de MapForce Server (véase [Interfaz .NET](#)⁶³). Puede saltarse este paso si está ejecutando el ejemplo existente de la API de MapForce, ya que el ejemplo ya contiene esa referencia.
- En el menú **Compilar** de Visual Studio, haga clic en **Administrador de configuración** y escoja la plataforma correcta, por ejemplo **Debug | x86** (o **Debug | x64**, en su caso). No use **Any CPU** como plataforma.
- Si tiene instalada la versión de 64 bits de MapForce Server, la aplicación que llama a la API (como la del ejemplo que aparece más abajo) deberá estar diseñada para la plataforma de 64 bits en Visual Studio. Además, la ruta de acceso del ejecutable de MapForce Server también debe adaptarse según corresponda en el código.

La solución de ejemplo está en el directorio "Archivos de programa", que necesita derechos de administrador. Puede ejecutar Visual Studio como administrador o copiar la solución en una carpeta

distinta que no necesite derechos de administrador.

Ejecutar el código de la asignación

El código siguiente ejecuta tres archivos de ejecución de servidor (.mfx). La tabla siguiente enumera los archivos de entrada que espera cada archivo .mfx, así como la salida que se genera tras la ejecución.

Archivo de ejecución (.mfx)	Entrada	Salida
TokenizeString.mfx	AltovaTools.xml	AltovaToolsFeatures.csv
SimpleTotal.mfx	ipo.xml	<i>String</i>
ClassifyTemperatures.mfx	Temperatures.xml	Temperatures_out.xml

Si tiene Altova MapForce puede examinar las asignaciones originales a partir de las cuales se crearon los archivos .mfx para entenderlas mejor. Se trata, respectivamente, de **TokenizeString1.mfd**, **SimpleTotal.mfd** y **ClassifyTemperatures.mfd**. Puede encontrar estas asignaciones en el directorio: **C:\users\<user>\Altova\MapForce2024\MapForceExamples**.

El ejemplo siguiente:

- Crea una instancia nueva de `Altova.MapForceServer.Server`, que es el objeto con el que trabajará de ahora en adelante.
- Define el directorio de trabajo en el que se lleva a cabo la ejecución. Se espera que los archivos de entrada estén en este directorio si hace referencia a ellos con una ruta relativa. Los archivos de salida también se crean en este directorio (véase la tabla anterior).
- Ejecuta **TokenizeString.mfx**. La ruta de acceso al archivo se da como argumento para el método `Run` (tenga en cuenta que la ruta es relativa al directorio de trabajo que se definió antes). Si el archivo se ejecuta correctamente, en el directorio de trabajo se crea un archivo .csv que representa el resultado de la asignación.
- Ejecuta **SimpleTotal.mfx**. En este caso la ruta también es relativa al directorio de trabajo. Esta asignación produce una cadena de salida; para obtenerla debe llamar al método `GetOutputParameter`.
- Ejecuta **ClassifyTemperatures.mfx**. Esta asignación espera un parámetro como entrada; puede indicarlo con el método `AddParameter`.

```
Option Explicit On
Option Explicit On

Option Explicit On

Module Program

    Sub Main()

        Try
            'Create a MapForce Server object;
            Dim objMFS As Altova.MapForceServer.Server = New Altova.MapForceServer.Server

            'Set a working directory - used as a base for relative paths for the MapForce
```

```

server execution (.mfx) file.
    'objMFS.WorkingDirectory = "C:\Program Files (x86)
\Altova\MapForceServer2020\etc\Examples"
    objMFS.WorkingDirectory = "..\..\.."

    'Default path to the MapForce Server executable is the installation path
(same dir with the MapForceServer.dll)
    'In case you moved the binaries on the disk, you need to explicitly set the
path to the .exe file
    'objMFS.ServerPath = "C:\Program Files (x86)
\Altova\MapForceServer2024\bin\MapForceServer.exe"
    'objMFS.ServerPath = "C:\Program
Files\Altova\MapForceServer2024\bin\MapForceServer.exe"

    'Set global resource file and configuration, if your mapping uses global
resources
    'objMFS.SetOption("globalresourcefile", "GlobalResources.xml") '"gr" can be
used as short name for "globalresourcefile"
    'objMFS.SetOption("globalresourceconfig", "Config2") '"gc" can be used as
short name for "globalresourceconfig"

-----

    'An example with input and output paths stored inside the MFX file
System.Console.WriteLine(vbCrLf & "Executing TokenizeString.mfx...")
If (objMFS.Run("TokenizeString.mfx")) Then
    System.Console.WriteLine("Successfully generated file
'AltovaToolFeatures.csv'.")
Else
    'execution failed. maybe no write permissions in working directory? Run
this program as administrator.
    System.Console.WriteLine(objMFS.LastExecutionMessage)
End If

-----

    'An example creating a simple output so that we can retrieve the result
explicitly
System.Console.WriteLine(vbCrLf & "Executing SimpleTotal.mfx...")
If (objMFS.Run("SimpleTotal.mfx")) Then
    System.Console.WriteLine("Mapping result is: " &
objMFS.GetOutputParameter("total"))
Else
    'execution failed (e.g. somebody deleted file ipo.xml)
    System.Console.WriteLine(objMFS.LastExecutionMessage)
End If

-----

    'an example with parameterized input
    ' the default of 'lower=5' gets changed to the value '10'
    ' mfx reads file Temperatures.xml and writes its output to

```

```
Temperatures_out.xml.  
    System.Console.WriteLine(vbCrLf & "Executing ClassifyTemperatures.mfx with  
parameter 'lower' set to '10' ...")  
    objMFS.AddParameter("lower", "10")  
    If (objMFS.Run("ClassifyTemperatures.mfx")) Then  
        System.Console.WriteLine("File Temperatures_out.xml has been written  
successfully.")  
    Else  
        'execution failed. maybe no write permissions in working directory? Run  
this program as administrator.  
        System.Console.WriteLine(objMFS.LastExecutionMessage)  
    End If  
  
    Catch ex As Exception  
        System.Console.WriteLine("Internal Error - " & ex.Message())  
    End Try  
  
End Sub  
  
End Module
```

5.2 Interfaz COM

MapForce 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:

- **MapForce.Server** (for 32-bit MapForce Server)
- **MapForce_x64.Server** (for 64-bit MapForce Server)

Puede encontrar estas clases 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 MapForce Server, lo mejor es desinstalar MapForce 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.

5.2.1 Ejemplo en C++

Este ejemplo muestra cómo ejecutar un archivo de ejecución de MapForce (`.mfx`) desde código C++. En Windows, los archivos de ejemplo están en: **C:\Program Files\Altova\MapForceServer2024\etc\Examples**.

Requisitos

Antes de ejecutar el siguiente código, compruebe que cumple con estos requisitos:

- MapForce Server está instalado y tiene asignada una licencia válida.
- MapForce Server está disponible como objeto de servidor COM (este proceso suele tener lugar automáticamente durante la instalación de MapForce Server. Consulte el apartado [Información sobre la interfaz COM](#)⁷¹ para obtener más información).

Ejecutar el código de la asignación

El código siguiente ejecuta tres archivos de ejecución de servidor (`.mfx`). La tabla siguiente enumera los archivos de entrada que espera cada archivo `.mfx`, así como la salida que se genera tras la ejecución.

Archivo de ejecución (.mfx)	Entrada	Salida
TokenizeString.mfx	AltovaTools.xml	AltovaToolsFeatures.csv
SimpleTotal.mfx	ipo.xml	<i>String</i>
ClassifyTemperatures.mfx	Temperatures.xml	Temperatures_out.xml

Si tiene Altova MapForce puede examinar las asignaciones originales a partir de las cuales se crearon los archivos `.mfx` para entenderlas mejor. Se trata, respectivamente, de **TokenizeString1.mfd**, **SimpleTotal.mfd** y **ClassifyTemperatures.mfd**. Puede encontrar estas asignaciones en el directorio: **C:\users\<user>\Altova\MapForce2024\MapForceExamples**.

El ejemplo siguiente:

- Crea una instancia nueva de `Altova.MapForceServer.Server`, que es el objeto con el que trabajará de ahora en adelante.
- Define el directorio de trabajo en el que se lleva a cabo la ejecución. Se espera que los archivos de entrada estén en este directorio si hace referencia a ellos con una ruta relativa. Los archivos de salida también se crean en este directorio (véase *la tabla anterior*).
- Ejecuta **TokenizeString.mfx**. La ruta de acceso al archivo se da como argumento para el método `Run` (tenga en cuenta que la ruta es relativa al directorio de trabajo que se definió antes). Si el archivo se ejecuta correctamente, en el directorio de trabajo se crea un archivo `.csv` que representa el resultado de la asignación.
- Ejecuta **SimpleTotal.mfx**. En este caso la ruta también es relativa al directorio de trabajo. Esta asignación produce una cadena de salida; para obtenerla debe llamar al método `GetOutputParameter`.
- Ejecuta **ClassifyTemperatures.mfx**. Esta asignación espera un parámetro como entrada; puede indicarlo con el método `AddParameter`.

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

#ifdef _WIN64
// 32-bit MapForce Server
#import "progid:MapForce.Server"
#else
// 64-bit MapForce Server
#import "progid:MapForce_x64.Server"
#endif

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

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

        //Set a working directory - used as a base for relative paths (you may need to
        adapt the path to the installation folder)
        pMFS->WorkingDirectory = ".."; // this is relative to this applications' working
        directory (the project folder)

        // Default path to the MapForce Server executable is the installation path (same
        dir with the MapForceServer.dll)
        // In case you moved the binaries on the disk, you need to explicitly set the path
        to the .exe file
        // pMFS.ServerPath = "C:\\Program Files (x86)\\Altova\\MapForceServer2024\\bin\\
        \\MapForceServer.exe";
        // pMFS.ServerPath = "C:\\Program Files\\Altova\\MapForceServer2024\\bin\\
```



```

\MapForceServer.exe";

    //Set global resource file and configuration, if your mapping uses global resources
    //pMFS->SetOption( "globalresourcefile", "GlobalResources.xml" ); // "gr" can be
used as short name for "globalresourcefile"
    //pMFS->SetOption( "globalresourceconfig", "Default" ); // "gc" can be used as
short name for "globalresourceconfig"

    //
-----

    // An example with input and output paths stored inside the MFX file
std::cout << "\nExecuting TokenizeString.mfx..." << std::endl;
if ( pMFS->Run( "TokenizeString.mfx" ) == VARIANT_TRUE )
    std::cout << "Successfully generated file 'AltovaToolFeatures.csv'." <<
std::endl;
else
{
    // execution failed. maybe no write permissions in working directory? Run this
program as administrator.
    std::cout << pMFS->LastExecutionMessage << std::endl;
}

//
-----

    // An example creating a simple output so that we can retrieve the result
explicitly
std::cout << "\nExecuting SimpleTotal.mfx..." << std::endl;
if ( pMFS->Run( "SimpleTotal.mfx" ) )
    std::cout << "Mapping result is: " + pMFS->GetOutputParameter( "total" ) <<
std::endl;
else
{
    // execution failed (e.g. somebody deleted file ipo.xml)
    std::cout << pMFS->LastExecutionMessage << std::endl;
}

//
-----

    // an example with parameterized input
    // the default of 'lower = 5' gets changed to the value '10'
    // mfx reads file Temperatures.xml and writes its output to Temperatures_out.xml.
std::cout << "\nExecuting ClassifyTemperatures.mfx with parameter 'lower' set to
'10' ..." << std::endl;
pMFS->AddParameter("lower", "10");
if ( pMFS->Run( "ClassifyTemperatures.mfx" ) )
    std::cout << "File Temperatures_out.xml has been written successfully." <<
std::endl;
else
{
    // execution failed. maybe no write permissions in working directory? Run this
program as administrator.
    std::cout << pMFS->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;
}

```

5.2.2 Ejemplo en VBScript

Este ejemplo muestra cómo ejecutar un archivo de ejecución de MapForce (.mfx) desde código VBScript. En Windows, los archivos de ejemplo están en: **C:\Program Files\Altova\MapForceServer2024\etc\Examples**.

Antes de ejecutar el siguiente código, compruebe que cumple con estos requisitos:

- MapForce Server está instalado y tiene asignada una licencia válida.
- MapForce Server está disponible como objeto de servidor COM (este proceso suele tener lugar automáticamente durante la instalación de MapForce Server. Consulte el apartado [Información sobre la interfaz COM](#)⁷¹ para obtener más información).

```

Option Explicit

REM This script produces extensive output.
REM It is best called from a cmd.exe console with "cscript MapForceServerAPI_sample.vbs"

'Create the MapForce Server object
Dim objMFS
' Since we load a COM-DLL we need care about the process architecture
On Error Resume Next ' ignore any COM errors avoiding uncontrolled script termination
Dim WshShell
Dim WshProcEnv
Set WshShell = CreateObject("WScript.Shell")
Set WshProcEnv = WshShell.Environment("Process")
Dim process_architecture
process_architecture= WshProcEnv("PROCESSOR_ARCHITECTURE")
If process_architecture = "x86" Then
    Set objMFS = WScript.GetObject( "", "MapForce.Server" )
    If Err.Number <> 0 then
        WScript.Echo("You are running in a 32-bit process but MapForce Server COM-API 32-bit seems not to be installed on your system.")
        WScript.Quit -1
    End If
Else
    Set objMFS = WScript.GetObject( "", "MapForce_x64.Server" )

```

```

If Err.Number <> 0 then
    WScript.Echo("You are running in a 64-bit process but MapForce Server COM-API 64-
bit seems not to be installed on your system.")
    WScript.Echo("If you have installed 32-bit MapForce 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 as a base for relative paths (you may need to adapt the
path to the installation folder)
REM objMFS.WorkingDirectory = "C:\Program Files (x86)
\Altova\MapForceServer2020\etc\Examples"
Dim currDir
Dim fso
Set fso = CreateObject("Scripting.FileSystemObject")
currDir = fso.GetParentFolderName(Wscript.ScriptFullName)
' set working folder to parent of this script
objMFS.WorkingDirectory = fso.GetParentFolderName( currDir )

' Default path to the MapForce Server executable is the installation path (same dir with
the MapForceServer.dll)
' In case you moved the binaries on the disk, you need to explicitly set the path to the
.exe file
' objMFS.ServerPath = "C:\Program Files (x86)
\Altova\MapForceServer2024\bin\MapForceServer.exe"

' Set global resource file and configuration, if your mapping uses global resources
' Call objMFS.SetOption("globalresourcefile", "GlobalResources.xml") "gr" can be used as
short name for "globalresourcefile"
' Call objMFS.SetOption("globalresourceconfig", "Config2") "gc" can be used as short name
for "globalresourceconfig"

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

' The Run method will return 'True' if the execution of the mfx file 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

REM -----
REM run an example with input and output paths stored inside the MFX file
' the path to the mfx file can be absolute or relative to the working directory
' depends on existence of file AltovaTools.xml in working directory
' creates output file AltovaToolFeatures.csv in working directory
WScript.Echo( "Processing TokenizeString.mfx..." )
If ( objMFS.Run( "TokenizeString.mfx" ) ) Then
    'WScript.Echo( objMFS.LastExecutionMessage )      ' execution log

```

```

WScript.Echo( "Successfully generated file AltovaToolFeatures.csv." )
Else
  'execution failed (e.g. somebody deleted file AltovaTools.xml)
  WScript.Echo( objMFS.LastExecutionMessage )
End If
WScript.Echo("")
' handle COM errors
If Err.Number <> 0 Then
  WScript.Echo("Internal error - " & Err.Description )
  WScript.Quit -1
End If

REM
-----
REM this is an example creating a simple output so that we can retrieve the result
explicitly
' depends on input XML file ipo.xml
WScript.Echo( "Processing SimpleTotal.mfx..." )
If ( objMFS.Run( "SimpleTotal.mfx" ) ) Then
  'WScript.Echo( objMFS.LastExecutionMessage )
  WScript.Echo( "Mapping result is: " & objMFS.GetOutputParameter("total") )
Else
  'execution failed (e.g. somebody deleted file ipo.xml)
  WScript.Echo( objMFS.LastExecutionMessage )
End If
WScript.Echo("")
' handle COM errors
If Err.Number <> 0 Then
  WScript.Echo("Internal error - " & Err.Description )
  WScript.Quit -1
End If

REM -----
REM This is an example with parameterized input
' the default of 'lower=5' gets changed to the value '10'
' mfx reads file Temperatures.xml and writes its output to Temperatures_out.xml.
WScript.Echo( "Processing ClassifyTemperatures.mfx with parameter 'lower' set to '10'
..." )
call objMFS.AddParameter("lower", "10")
If ( objMFS.Run( "ClassifyTemperatures.mfx" ) ) Then
  'WScript.Echo( objMFS.LastExecutionMessage )
  WScript.Echo( "File Temperatures_out.xml has been written successfully." )
Else
  'execution failed (e.g. somebody locks file Temperatures_out.xml)
  WScript.Echo( objMFS.LastExecutionMessage )
End If
call objMFS.ClearParameterList()
WScript.Echo("")

' 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
```

5.2.3 Ejemplo en Visual Basic for Applications (VBA)

Microsoft Visual Basic for Applications (VBA) se usa sobre todo para automatizar tareas en Microsoft Office. Sin embargo, desde VBA también puede llamar a la API de MapForce Server para ejecutar asignaciones. Las instrucciones que aparecen a continuación se han probado en MapForce Server y VBA para Microsoft Office 2013. Dependiendo del entorno de desarrollo puede que los pasos del proceso cambien ligeramente.

Requisitos previos

Antes de llamar a las funciones de la API de MapForce Server desde su proyecto VBA debe tener en cuenta estos requisitos previos:

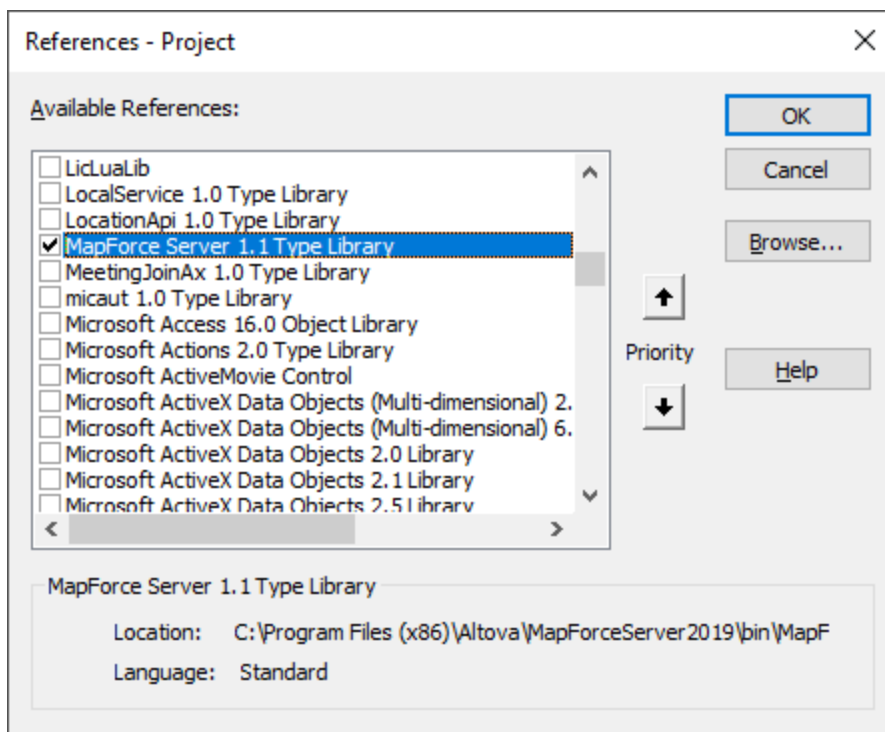
1. Microsoft Office y MapForce Server deben estar instalados en el mismo equipo.
2. La arquitectura de MapForce Server (32 o 64 bits) debe coincidir con la de Microsoft Office. Por ejemplo, si ejecuta VBA en Microsoft Office de 32 bits, entonces debe usar MapForce Server de 32 bits. Para averiguar si su producto Office es de 32 o 64 bits haga clic en la pestaña **Archivo** y después en **Cuenta** y en *Acerca de Excel/Word*.
3. El proyecto VBA debe hacer referencia a la biblioteca de MapForce Server (ver instrucciones más abajo).

Cómo añadir una referencia a la biblioteca de MapForce Server desde el proyecto VBA

1. En un documento Microsoft Office que tenga habilitadas las macros, haga clic en la pestaña **Programador** y después en **Visual Basic**.

La pestaña **Programador** no está habilitada por defecto en Microsoft Office. Para habilitarla en un programa de Office 2013 haga clic con el botón derecho en la cinta y seleccione **Personalizar la cinta de opciones** en el menú contextual. Después, en el cuadro de diálogo "Opciones", marque la casilla *Programador* del panel "Pestañas principales".

2. En el entorno de desarrollo VBA, en el menú **Herramientas**, haga clic en **Referencias**.



3. Seleccione la **biblioteca de tipos de MapForce Server**.

Para llamar a la API de MapForce Server

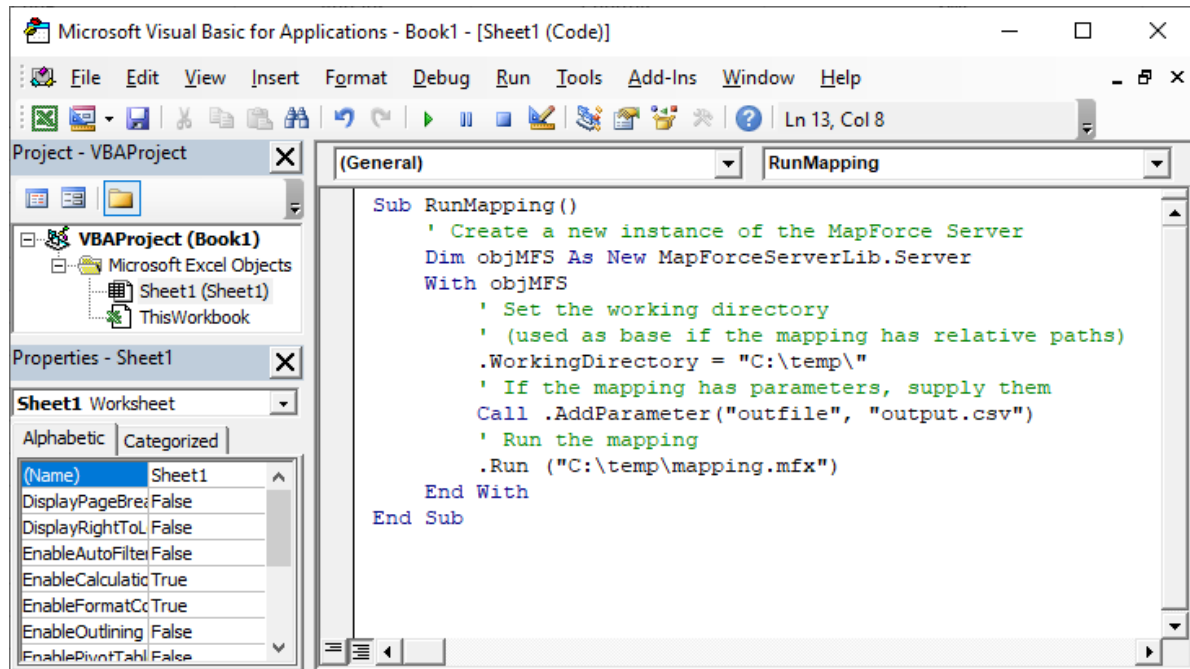
Tras añadir la referencia a la biblioteca de MapForce Server al proyecto VBA, podrá introducir código VBA en la ventana del editor de código. Por ejemplo, el código que aparece a continuación llama a MapForce Server y ejecuta un archivo de asignación ejecutable (`mapping.mfx`) que toma un parámetro de entrada llamado `outfile` cuyo valor es `output.csv`.

```

Sub RunMapping()
    ' Crear una instancia nueva de MapForce Server
    Dim objMFS As New MapForceServerLib.Server
    With objMFS
        ' Establecer el directorio de trabajo, que sirve de base para las rutas de
        acceso relativas
        .WorkingDirectory = "C:\temp\"
        ' Aportar parámetros si la asignación los utiliza
        Call .AddParameter("archivosalida", "salida.csv")
        ' Ejecutar la asignación
        .Run ("C:\temp\asignación.mfx")
    End With
End Sub

```

Pulse **F5** para depurar el código VBA y ejecutar la asignación.



5.3 Interfaz Java

La API consiste en un archivo JAR (`MapForceServer.jar`) y un archivo JNI (`MapForceServer.dll`). Estos dos archivos, así como otros archivos relacionados con la API, están disponibles en la carpeta `bin` de la carpeta de instalación de MapForce Server. Puede hacer referencia a esos archivos desde su ubicación original o copiarlos a otra ubicación si eso encaja en su configuración del proyecto. (En sistemas Windows necesitará derechos de administrador para ejecutar el programa desde su ubicación original.)

Nota: si instaló una versión de 64 bits de MapForce Server, los archivos de la versión de 32 bits de `MapForceServer.jar` y (`MapForceServer.dll` estarán en la carpeta `bin\API_32bit` de la carpeta de instalación de MapForce Server. Necesitará estos archivos si está usando una versión de 32 bits. Ocurre igual al contrario, si instaló una versión de 32 bits de MapForce Server, los archivos de la versión de 64 bits de `APPNAME-UL%>.jar` y (`MapForceServer.dll` estarán en la carpeta `bin\API_64bit`. Necesitará estos archivos si está usando una versión de 64 bits.

Para acceder al código Java de la API de MapForce Server agregue estas referencias al archivo `.classpath` de su proyecto Java.

<code>MapForceServer.jar</code>	La biblioteca que se comunica con MapForce Server
<code>MapForceServer_JavaDoc.zip</code>	Documentación de la API de MapForce Server

Además, `java.library.path` debe incluir la carpeta en la que se encuentra el archivo de biblioteca JNI (`MapForceServer.dll`).

Si implementa su proyecto en una aplicación de servidor, asegúrese de que `MapForceServer.jar` y `MapForceServer.dll` están configurados correctamente con Java en el equipo servidor.

Para ver un ejemplo de cómo usar los archivos de biblioteca de la API consulte el archivo `buildAndRun.bat` (véase *más abajo*), que se encuentra en la carpeta `etc\Examples\Java` de la carpeta de instalación de MapForce Server.

Generar y ejecutar un programa Java con la API

Para aprender a generar y ejecutar un programa Java que use la API de MapForce Server consulte el archivo de ejemplo `buildAndRun.bat`. Puede reutilizar este archivo para ejecutar sus propios programas de Java; para ello modifíquelo como necesite.

Inicie el archivo de lote en una interfaz de la línea de comandos con este comando:

```
buildAndRun "path_to_Java_bin_folder"
```

Nota: para comprobar si Java es su classpath puede ejecutar el comando `java --version`. Si Java no es su classpath debe proporcionar la ruta de acceso relevante como parámetro del comando `buildAndRun`. Si la ruta contiene espacios debe usar comillas entorno a la ruta.

Extracto de `buildAndRun.bat`

```
@echo off
if %1.==. goto error
```



```
REM The location of the JAVA API binaries, the JAR file and the JNI library.
REM Adapt to your needs.
SETLOCAL
Set JavaAPIBinPath=%PROGRAMFILES%\Altova\MapForceServer2024\bin

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

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

@echo off
goto end

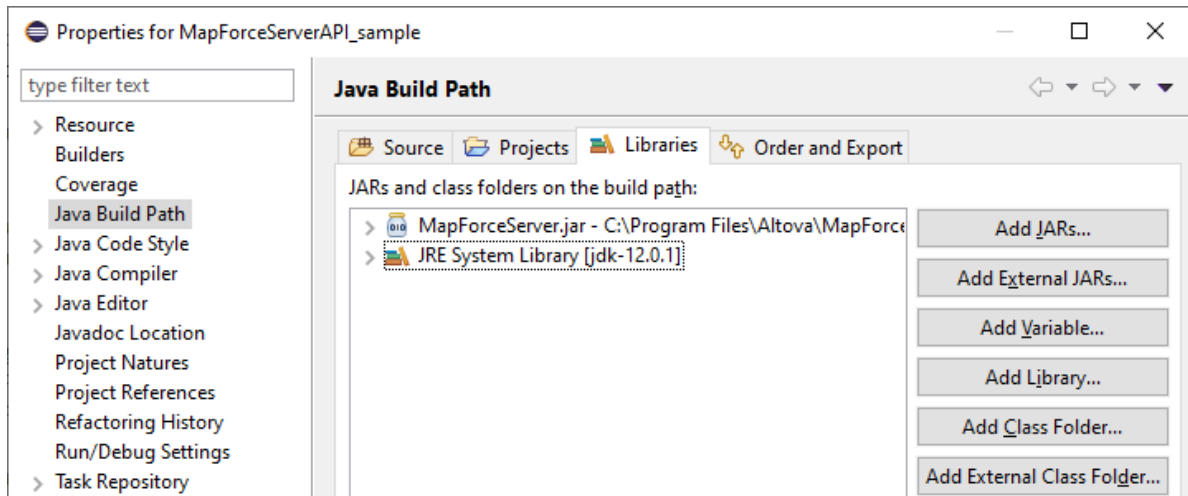
:error
echo Usage: buildAndRun "<path_to_java_bin_folder>"

:end
```

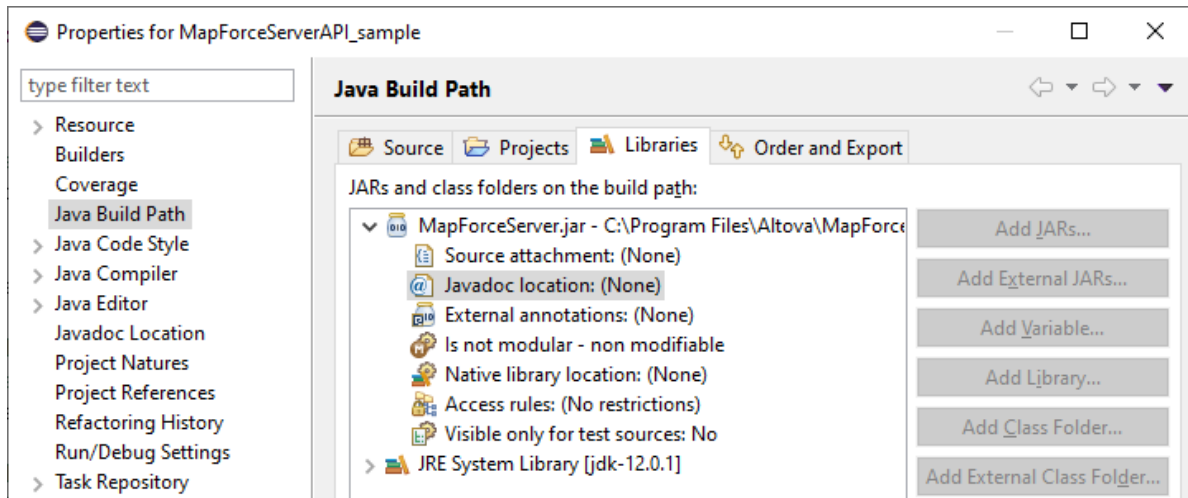
Para añadir las referencias de la biblioteca de MapForce Server en Eclipse

Si está usando Eclipse como entorno de desarrollo Java, puede agregar las referencias a la biblioteca que necesite a CLASSPATH editando las propiedades del proyecto Java, como explicamos a continuación. También puede adjuntar documentación en formato JavaDoc a la biblioteca .jar. Puede encontrar el JavaDoc en la carpeta **bin** de la carpeta de instalación de MapForce Server; a continuación explicamos cómo hacer que el JavaDoc sea visible desde Eclipse.

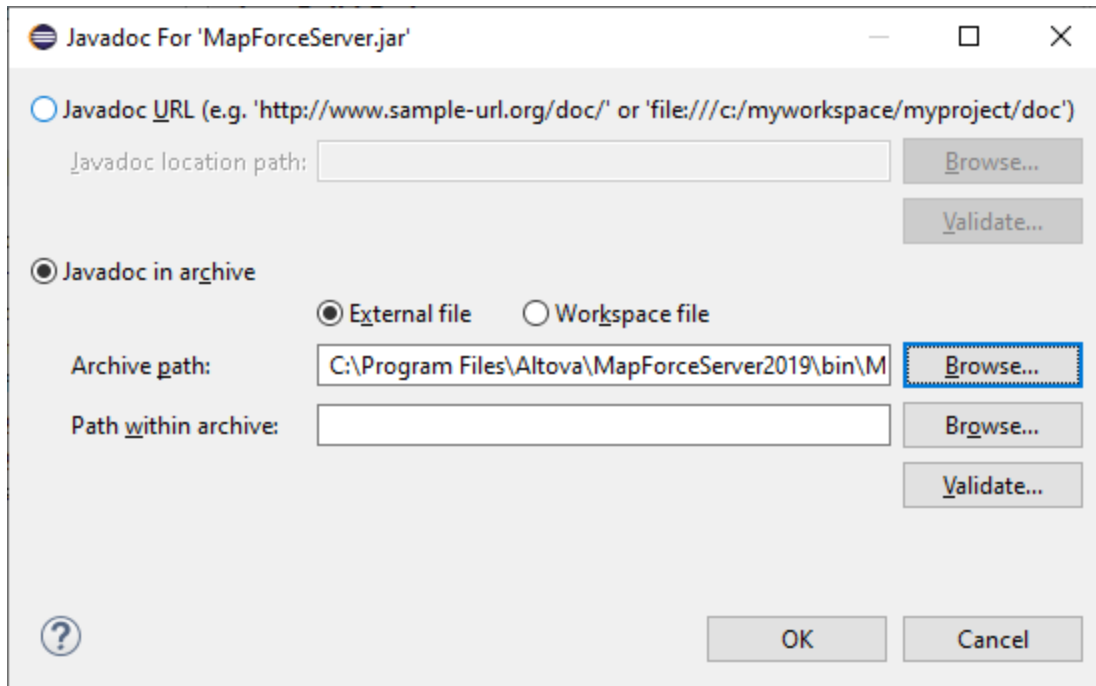
1. Abra el proyecto en Eclipse y haga clic en **Project | Properties**.
2. Haga clic en **Java Build Path**.
3. En la pestaña *Libraries* de bibliotecas haga clic en el botón **Add External JARs** para añadir archivos JAR externos y navegue hasta el archivo `MapForceServer.jar` (situado en la carpeta de instalación de MapForce Server).



4. En el panel *JARs and class folders on the build path*, expanda el nodo **MapForceServer.jar** y después haga doble clic en el registro **Javadoc location: (None)**.



5. Compruebe que están marcadas las opciones **Javadoc in archive** y **External file** y en el campo *Archive path* seleccione el archivo `MapForceServer_JavaDoc.zip` (situado en la carpeta de instalación de MapForce Server).



- Haga clic en **OK** para terminar. La referencia a la biblioteca de MapForce Server y el archivo de almacenamiento Javadoc se añade ahora al archivo `.classpath` del proyecto.

A continuación puede ver un ejemplo de archivo `.classpath` que hace referencia a los archivos de la carpeta de instalación en un sistema operativo de 64 bits que ejecuta la versión de 32 bits de MapForce Server (las líneas relevantes están resaltadas en amarillo):

```
<?xml version="1.0" encoding="UTF-8"?>
<classpath>
  <classpathentry kind="src" path=""/>
  <classpathentry kind="lib" path="C:/Program
Files/Altova/MapForceServer2024/bin/MapForceServer.jar">
    <attributes>
      <attribute name="javadoc_location" value="jar:file:/C:/Program%
20Files/Altova/MapForceServer2024/bin/MapForceServer_JavaDoc.zip!/">
    </attributes>
  </classpathentry>
  <classpathentry kind="con" path="org.eclipse.jdt.launching.JRE_CONTAINER">
    <attributes>
      <attribute name="module" value="true"/>
    </attributes>
  </classpathentry>
  <classpathentry kind="output" path=""/>
</classpath>
```

5.3.1 Ejemplo en Java

Este ejemplo muestra cómo ejecutar un archivo de ejecución de MapForce (.mfx) desde código Java. En Windows, los archivos de ejemplo están en: **C:\Program Files\Altova\MapForceServer2024\etc\Examples**.

- **TokenizeString.mfx**: un archivo de ejecución de MapForce Server que ejecuta el programa Java.
- **AltovaTools.xml**: un archivo XML que proporciona datos de entrada al programa de la asignación.

En las plataformas Linux y Mac no hay ningún ejemplo ya instalado, pero puede preparar un archivo .mfx ejecutable como el de este ejemplo siguiendo estos pasos:

1. Abra un archivo de diseño de asignación (.mfd) con MapForce en Windows.
2. En el menú **Archivo**, seleccione **Configurar asignación** y desmarque la casilla *Convertir las rutas de acceso en absolutas en el código generado* si estaba marcada.
3. En cada componente de la asignación abra el cuadro de diálogo "Propiedades" (haga doble clic en la barra de título del componente, por ejemplo) y cambie todas las rutas de los archivos de absolutas a relativas. Marque también la casilla *Guardar todas las rutas de acceso de archivos como relativas al archivo MFD*. Si le resulta más cómodo, puede copiar todos los archivos y esquemas de entrada en la misma carpeta en la que se encuentra la asignación para poder hacer referencia a ellos usando solamente su nombre. Consulte la documentación de MapForce para más información sobre las rutas relativas y absolutas en el diseño de asignaciones.
4. En el menú **Archivo**, seleccione **Compilar en archivo de ejecución de MapForce Server** para generar el archivo .mfx que después ejecutará con MapForce Server, como se muestra en el extracto de código de más abajo.

Requisitos

Antes de ejecutar el código siguiente, asegúrese de que cumple estos requisitos:

- MapForce Server está instalado y posee una licencia
- Java CLASSPATH incluye una referencia a la biblioteca **MapForceServer.jar** (para ver un ejemplo consulte [Interfaz Java](#)⁸⁰).

Si está usando un archivo .mfx personalizado como el de más arriba, puede que deba cumplir también otros requisitos en función de los datos que procese la asignación. Para más información consulte [Preparar asignaciones para ejecución en servidor](#)³².

En Windows, el proyecto Java de ejemplo está en el directorio "Archivos de programa", que necesita derechos de administrador, por lo que deberá ejecutar su entorno de desarrollo Java (por ejemplo, Eclipse) como administrador o copiar el ejemplo a una carpeta distinta para la que no necesite esos derechos.

Ejecutar la asignación en Java

El extracto de código siguiente primero crea un objeto de MapForce Server. A continuación define el directorio de trabajo en el que la aplicación debe buscar los archivos de entrada de la asignación y donde también se generan los archivos de salida de la asignación. Como hemos mencionado anteriormente, el archivo de asignación de ejemplo lee datos desde un archivo XML de origen, por lo que debe asegurarse de el directorio de trabajo contiene tanto el archivo XML como el esquema.

El método `setServerPath` indica la ruta de acceso al ejecutable de MapForce Server. Por ejemplo, en Ubuntu, este sería `/opt/Altova/MapForceServer2024/bin/mapforceserver`. Puede omitir `setServerPath` si no ha movido **MapForceServer.jar** de su ubicación predeterminada.

Por último, el método `run` ejecuta una asignación (archivo `.mfx`) compilado con MapForce (en este ejemplo, **TokenizeString.mfx**). Si se ejecuta correctamente, el programa siguiente genera un archivo CSV y un archivo XML en el directorio de trabajo. Si la ejecución falla, el programa intenta imprimir el último mensaje de ejecución generado por MapForce Server.

```
import com.altova.mapforceserver.MapForceServer;

public class Program {

    public static void main(String[] args)
    {
        MapForceServer objMFS;
        try
        {
            // set up the server
            objMFS = new MapForceServer();

            // The default location of the server is the directory that the java native
library is in.
            // With the following line you could select a different server binary.
            // objMFS.setServerPath(strServerPath);

            // The sample data is located in the parent folder of the Java sample code
            objMFS.setWorkingDirectory("..");

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

            //Set global resource file and configuration, if your mapping uses global
resources
            //objMFS.setOption( "globalresourcefile", "GlobalResources.xml" ); // "gr"
can be used as short name for "globalresourcefile"
            //objMFS.setOption( "globalresourceconfig", "Default" ); // "gc" can be used
as short name for "globalresourceconfig"

            //
            -----
            // An example with input and output paths stored inside the MFX file
            System.out.println("\nExecuting TokenizeString.mfx...");
            if (objMFS.run("TokenizeString.mfx"))
                System.out.println("Success: " + objMFS.getLastExecutionMessage());
            else
                System.out.println("Unsuccessful: " +
objMFS.getLastExecutionMessage());

            //
            -----
            // An example creating a simple output so that we can retrieve the result
explicitly
            System.out.println("\nExecuting SimpleTotal.mfx...");
            if (objMFS.run("SimpleTotal.mfx"))
```

```
        System.out.println("Mapping result is: " +
objMFS.getOutputParameter("total"));
        else
        {
            // execution failed (e.g. somebody deleted file ipo.xml)
            System.out.println(objMFS.getLastExecutionMessage());
        }
        //
-----
        // An example with parameterized input
        // the default of 'lower = 5' gets changed to the value '10'
        // mfx reads file Temperatures.xml and writes its output to
Temperatures_out.xml.
        System.out.println("\nExecuting ClassifyTemperatures.mfx with parameter
'lower' set to '10' ...");
        objMFS.addParameter("lower", "10");
        if (objMFS.run("ClassifyTemperatures.mfx"))
            System.out.println("File Temperatures_out.xml has been written
successfully.");
        else
        {
            // execution failed. maybe no write permissions in working directory? Run
this program as administrator.
            System.out.println(objMFS.getLastExecutionMessage());
        }

        // You can stop the server explicitly by invoking the 'stop' method if you
don't want to let the garbage collector decide.
        objMFS.stop();
    }

    catch (Exception e)
    {
        System.out.println("ERROR: " + e.getMessage());
    }
}
}
```

5.4 Ejemplo: ejecutar asignaciones con parámetros

En este ejemplo aprenderá a compilar una asignación de MapForce en un archivo de ejecución de MapForce Server (.mfx) y a ejecutarlo desde la API de MapForce. Más concretamente, en el ejemplo la asignación toma el nombre del archivo de entrada como parámetro. Consulte también los ejemplos anteriores: [C#](#)⁶⁴, [C++](#)⁷¹, [VB.NET](#)⁶⁷, [VBScript](#)⁷⁴ o [Java](#)⁸⁴.

En este ejemplo usamos MapForce para que pueda ver y entender el diseño original de la asignación. También usamos MapForce para compilar la asignación en un archivo de ejecución de MapForce Server (.mfx) y configurar opciones, como las rutas relativas o absolutas.

La plataforma de servidor que se usa en este ejemplo es Windows. Puede usar un servidor que esté en el mismo equipo en el que está instalado MapForce o en uno distinto. También puede ejecutar este ejemplo en un equipo con Linux o con Mac (en Java), siempre que ajuste las rutas según la plataforma que use.

Requisitos

Para ejecutar esta asignación los requisitos son los mismos que los que se han descrito en los ejemplos anteriores: [C#](#)⁶⁴, [C++](#)⁷¹, [VB.NET](#)⁶⁷, [VBScript](#)⁷⁴ o [Java](#)⁸⁴.

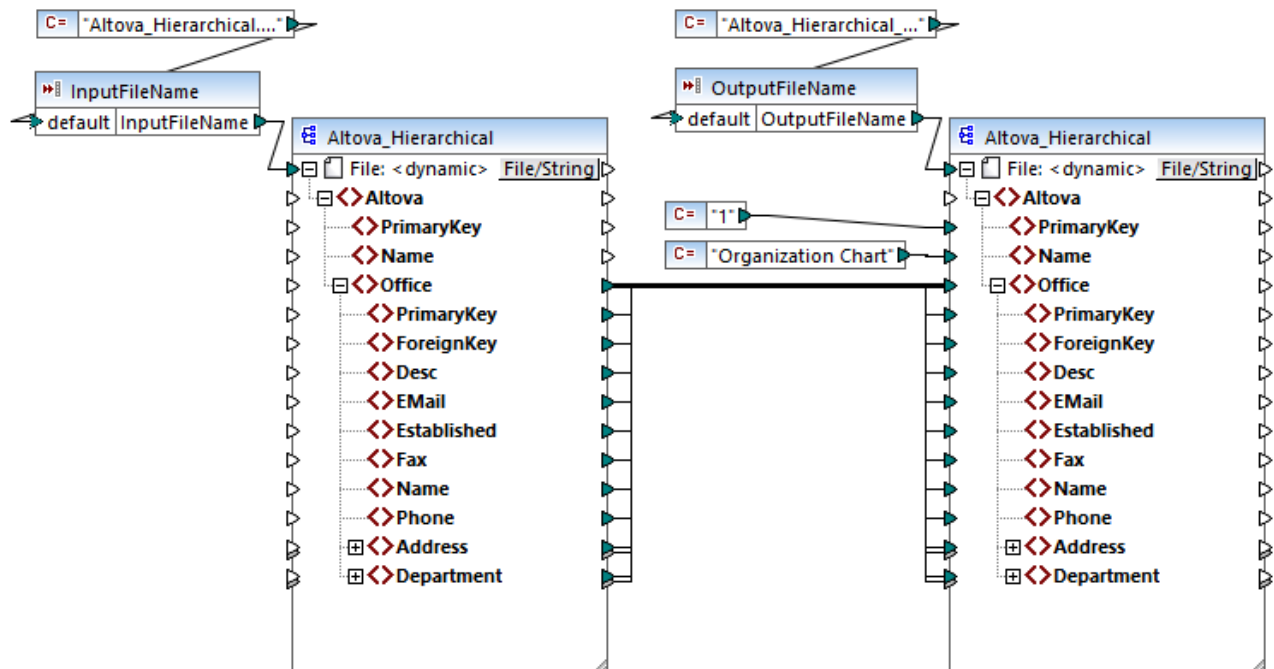
Preparar la asignación para ejecutarla en servidor

Una vez MapForce esté instalado y lo ejecute por primera vez, verá que tiene disponibles varios archivos de asignación de ejemplo en este directorio:

C:\Users\<<username>\Documents\Altova\MapForce2024\MapForceExamples

Asegúrese de que cambia esta ruta según la versión de MapForce que tenga.

El diseño de la asignación que usamos en este ejemplo es **FileNamesAsParameters.mfd**. Como puede ver a continuación, se trata de una asignación directa que copia datos de un origen en un archivo XML de destino prácticamente sin cambiarlos. Solamente los campos **PrimaryKey** y **Name** del archivo XML de destino se rellenan con constantes de la asignación.



FileNamesAsParameters.mfd

La asignación tiene dos parámetros de entrada: **InputFileName** y **OutputFileName**. Para que sea posible acceder a una vista previa de la asignación en MapForce los valores de los parámetros los dan dos constantes. Como se describe más abajo, los valores de los parámetros se pueden reemplazar con otros al ejecutar la asignación con MapForce Server.

Observe cómo los dos componentes, el de entrada y el de salida, están configurados para que obtengan el nombre del archivo de forma dinámica desde la asignación. Es decir, hay un conector *File: <dynamic>* disponible en la parte superior de cada uno de los componentes; este conector lee datos de los respectivos parámetros de asignación (**InputFileName** u **OutputFileName**). En MapForce puede definir o cambiar esta opción haciendo clic en el botón Archivo/cadena que hay en la parte superior derecha de los componentes. Tenga en cuenta que el nombre del archivo de entrada dado como parámetro debe ser una referencia a un archivo XML físico que proporcione datos de entrada a la asignación (en este ejemplo, **Altova_Hierarchical.xml** de la misma carpeta que el archivo de diseño de la asignación). El nombre del archivo de salida debe ser válido, por ejemplo **Output.xml**.

Antes de compilar la asignación en un archivo de ejecución de MapForce Server (.mfx) se recomienda que compruebe la configuración de la asignación. Para ello haga clic con el botón derecho en un área vacía de la asignación y seleccione **Configurar asignación** en el menú contextual.

En este ejemplo vamos a cambiar la configuración que ve más arriba. En concreto, al desmarcar la casilla **Convertir las rutas de acceso en absolutas en el código generado**, las rutas absolutas de los archivos de entrada o de salida que se usen en la asignación se resuelven como relativas al archivo de diseño de la asignación (.mfd). Al ejecutar la asignación, MapForce Server busca esas rutas en el directorio de trabajo del programa. El directorio de trabajo predeterminado depende de la plataforma que use para acceder a la API de MapForce Server. Por ejemplo, en el caso de las aplicaciones en C#, es el mismo directorio que el ejecutable. Además, como puede ver a continuación, puede cambiar el directorio de trabajo con la propiedad de la API [WorkingDirectory](#)¹⁰² (o con el método Java [setWorkingDirectory](#)¹²³, en su caso).

Notas:

- En el cuadro de diálogo **Configurar asignación**, las únicas opciones que afectan directamente a la compilación del archivo .mfx son **Convertir las rutas de acceso en absolutas en el código generado** y **Extremos de línea**.
- En esta asignación no hizo falta editar cada uno de los componentes en MapForce para cambiar las rutas de absolutas a relativas porque las rutas ya eran relativas. De lo contrario tendría que cambiarlas primero, véase [Preparar asignaciones para ejecución en servidor](#)³².

En este ejemplo usaremos **C:\MappingExample** como directorio de trabajo. Primero debe copiar el archivo de entrada **Altova_Hierarchica.xml** al que hace referencia la asignación desde **C:\Users\<username>\Documents\Altova\MapForce2024\MapForceExamples** al directorio de trabajo.

En este ejemplo tanto la entrada como la salida son archivos XML, por lo que no hay necesidad de copiar el archivo de esquema XML, ya que la información que de él se deriva está incrustada en el archivo .mfx durante la compilación. Si su asignación usa otros tipos de componentes (como bases de datos), puede que deba cumplir también otros requisitos, como explicamos en [Preparar asignaciones para ejecución en servidor](#)³².

Por último, para compilar la asignación en un archivo de ejecución de servidor:

- En el menú **Archivo** haga clic en **Compilar en archivo de ejecución de MapForce** y seleccione un directorio de destino. En este ejemplo ese directorio de destino es el mismo que el directorio de trabajo, **C:\MappingExample**.

En el extracto de código siguiente puede ver cómo ejecutar la asignación desde distintos entornos usando la API de MapForce Server. Observe que los dos parámetros son referenciados por el mismo nombre que tienen

en el diseño de la asignación. Asimismo, el valor de parámetro tiene el mismo tipo de datos que el definido en la asignación (en este caso, *cadena*).

Ejecutar una asignación desde C++

Puede ejecutar el archivo .mfx usando código C++ como el que ve a continuación. Para ejecutarlo correctamente asegúrese de que cuenta con una licencia válida para MapForce Server, consulte también el comando de la línea de comandos [verifylicense](#) ⁶⁰.

C++

```
#include <iostream>
#include "atlbase.h"

// 32-bit MapForce Server
#import "progid:MapForce.Server"

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

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

        //Set a working directory - used as a base for relative paths
        pMFS->WorkingDirectory = "C:\\\\MappingExample";

        //Prepare the parameters
        pMFS->AddParameter( "InputFileName", "Altova_Hierarchical.xml" );
        pMFS->AddParameter( "OutputFileName", "Output.xml" );

        //Run the mapping
        if ( pMFS->Run( "FileNamesAsParameters.mfx" ) )
            std::cout << "Success - finished execution" << std::endl;
        else
            std::cout << pMFS->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;
}
```

```
}
```

Ejecutar la asignación desde C#

Puede ejecutar el archivo .mfx usando código C# como el que ve a continuación. Para ejecutarlo correctamente primero debe añadir una referencia al DLL de MapForce Server en Visual Studio, como se explica en [Interfaz .NET](#)⁶³, y asegurarse de que cuenta con una licencia válida para MapForce Server, consulte también el comando de la línea de comandos [verifylicense](#)⁶⁰.

C#

```
static void Main(string[] args)
{
    try
    {
        // Create a MapForce Server object
        Altova.MapForceServer.Server objMFS = new Altova.MapForceServer.Server();

        // Set the working directory - all relative paths will be resolved against it
        objMFS.WorkingDirectory = "C:\\MappingExample";

        //Prepare the parameters
        objMFS.AddParameter("InputFileName", "Altova_Hierarchical.xml");
        objMFS.AddParameter("OutputFileName", "Output.xml");

        //Run the mapping
        if (objMFS.Run("FileNamesAsParameters.mfx"))
            System.Console.WriteLine("Success - finished execution");
        else
            System.Console.WriteLine(objMFS.LastExecutionMessage);
    }
    catch(System.Runtime.InteropServices.COMException ex)
    {
        System.Console.WriteLine("Internal Error: " + ex.Message);
    }
}
```

Ejecutar la asignación desde Java

Puede ejecutar el archivo .mfx usando código C# como el que ve a continuación. Para ejecutarlo correctamente asegúrese de que:

- la variable de entorno Java CLASSPATH incluye una referencia a la biblioteca **MapForceServer.jar**, como se describe en [Interfaz Java](#)⁸⁰
- usted cuenta con una licencia válida para MapForce Server.

Java

```
public static void main(String[] args) {
    try {
        // Create a MapForce Server object
        com.altova.mapforceserver.MapForceServer objMFS = new
```

```
com.altova.mapforceserver.MapForceServer();

    // Set the working directory - all relative paths will be resolved against it
    // objMFS.setWorkingDirectory("/home/ubuntu/Downloads/MappingExample");
    objMFS.setWorkingDirectory("C:\\MappingExample");

    // Add the mapping parameters
    objMFS.addParameter("InputFileName", "Altova_Hierarchical.xml");
    objMFS.addParameter("OutputFileName", "Output.xml");

    // Run the mapping
    if ( objMFS.run( "FileNamesAsParameters.mfx" ) )
        System.out.println( "Success - finished execution" );
    else
        System.out.println( objMFS.getLastExecutionMessage() );

} catch (Exception e) {
    e.printStackTrace();
}

System.out.println("Finished execution");
}
```

5.5 Referencia de la API (.NET y COM)

Este apartado ofrece una referencia general de los elementos de la API de MapForce Server (como las interfaces y los métodos) que sirven para escribir código para las plataformas COM o .NET.

5.5.1 Interfaces

5.5.1.1 IServer

La interfaz `IServer` crea una instancia nueva del objeto MapForce Server y ofrece acceso al motor de MapForce Server.

Si está usando C++ en una plataforma COM, el nombre de la interfaz principal es `IServer`. Si está usando un lenguaje .NET, como C# o VB.NET, el nombre de la interfaz principal es `Server`.

Propiedades

Nombre	Descripción
APIMajorVersion ⁹⁶	De solo lectura. Obtiene la versión principal de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
APIMinorVersion ⁹⁶	De solo lectura. Obtiene la versión secundaria de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
APIServicePackVersion ⁹⁷	De solo lectura. Obtiene la versión de service pack de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
Is64Bit ⁹⁸	De solo lectura. Devuelve true si el motor de MapForce Server es un ejecutable de 64 bits.
LastExecutionMessage ⁹⁸	De solo lectura. Obtiene el mensaje obtenido durante la última ejecución del comando Run .
MajorVersion ⁹⁹	De solo lectura. Obtiene la versión principal del producto como número de años a partir de 1998 (p. ej. "20" para Altova MapForce Server 2018).
MinorVersion ⁹⁹	De solo lectura. Obtiene la versión secundaria del producto (p. ej. "2" para Altova MapForce Server 2018 r2).

Nombre	Descripción
ProductName ¹⁰⁰	De solo lectura. Obtiene la versión secundaria del producto (p. ej. "para Altova MapForce Server").
ProductNameAndVersion ¹⁰⁰	De solo lectura. Obtiene el nombre completo y la versión del producto (p. ej. "Altova MapForce Server 2018 r2 sp1 (x64)").
ServerPath ¹⁰¹	Obtiene o establece la ruta de acceso del ejecutable de MapForce Server.
ServicePackVersion ¹⁰¹	De solo lectura. Obtiene la versión de service pack del producto (p. ej. "1" para Altova MapForce Server 2018 r2 sp1).
WorkingDirectory ¹⁰²	Obtiene o establece el directorio actual para los trabajos que están en ejecución (las rutas de acceso relativas se evaluarán en función del directorio de trabajo).

Métodos

Nombre	Descripción
AddCredentialProperty ¹⁰²	Añade una propiedad a la credencial actual (por ejemplo, el nombre de usuario, la contraseña o ambos). El primer argumento indica el nombre de la propiedad y el segundo su valor. Son nombres válidos de propiedad: <code>username</code> , <code>password</code> . En MapForce Server Advanced Edition también se admite el nombre de propiedad <code>oauth:token</code> . Se llama a este método después de llamar a <code>BeginCredential()</code> y antes de llamar a <code>EndCredential()</code> .
AddParameter ¹⁰³	Asigna un valor al parámetro definido en la asignación. El primer argumento especifica el nombre del parámetro tal y como se definió en la asignación. El segundo argumento especifica el valor del parámetro.
BeginCredential ¹⁰⁴	Crea una credencial nueva con el nombre dado como argumento. Si llama a este método también debe añadirle propiedades usando <code>AddCredentialProperty()</code> y cerrar la credencial llamando a <code>EndCredential()</code> .
ClearCredentialList ¹⁰⁵	Borra la lista de credenciales definida previamente. Todas las credenciales son válidas para el objeto mientras este dure. Llame a este método si necesita borrar explícitamente todas las credenciales que haya definido previamente.
ClearOptions ¹⁰⁵	Borra la lista de opciones previamente establecida con el método <code>SetOption</code> . Todas las opciones establecidas con el método <code>SetOption</code> son válidas para la vida útil del objeto. Llame

Nombre	Descripción
	a este método si necesita borrar de forma explícita todas las opciones previamente establecidas.
ClearParameterList ¹⁰⁶	Borra la lista de parámetros que se definieron previamente con el método <code>AddParameter</code> .
EndCredential ¹⁰⁶	Cierra un objeto de credencial creado previamente con el método <code>BeginCredential</code> .
GetOutputParameter ¹⁰⁷	Obtiene la cadena de salida generada por el último comando <code>run</code> . Devuelve <code>null</code> si no se generó ningún resultado. Esta función requiere un parámetro de cadena que identifique el nombre que tiene el componente de salida en Altova MapForce.
Run ¹⁰⁷	Ejecuta un archivo de ejecución de MapForce Server Execution (archivo <code>.mfx</code>). Devuelve <code>true</code> si se ejecuta correctamente. De lo contrario devuelve <code>false</code> .
SetOption ¹⁰⁸	<p>Establece una opción antes de ejecutar la asignación. El primer argumento especifica el nombre de la opción, mientras que el segundo argumento especifica el valor de la opción. Este método es especialmente útil para asignaciones diseñadas para que consuman recursos globales (véase Recursos globales de Altova ³⁸). Estas son las opciones compatibles actualmente:</p> <ul style="list-style-type: none"> • globalresourcefile (or gr): una ruta de acceso a los archivos de los recursos globales. (Si se indica esta opción también se debe indicar un nombre de configuración de los recursos globales, véase el siguiente punto). • globalresourcefile (or gr): un nombre de configuración de los recursos globales. (Si se indica esta opción también se debe indicar una ruta de acceso a los archivos de los recursos globales, véase el siguiente punto). • catalog: la ruta de acceso a un archivo RootCatalog.xml personalizado. Esta opción permite indicar un archivo de catálogo personalizado para resolver las URL que usa la asignación. El catálogo predeterminado está en el subdirectorio <code>etc</code> del directorio del programa de instalación. • taxonomy-package: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación. • taxonomy-packages-config-file: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación. <p>Todas las opciones establecidas son válidas para la vida útil del objeto. Si establece una opción dos veces con el mismo</p>

Nombre	Descripción
	nombre se sobrescribirá la opción anterior. Para borrar de forma explícita todas las opciones, llame al método <code>ClearOptions()</code> .

Ejemplos

En los siguientes ejemplos se muestra cómo inicializar y ejecutar MapForce Server en varios idiomas:

- [Ejemplo C++](#) ⁷¹
- [Ejemplo C#](#) ⁶⁴
- [Ejemplo VBScript](#) ⁷⁴
- [Ejemplo VB.NET](#) ⁶⁷.

5.5.1.1.1 Propiedades

5.5.1.1.1.1 *APIMajorVersion*

Obtiene la versión principal de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
APIMajorVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property APIMajorVersion As Integer
```

VB.NET

5.5.1.1.1.2 *APIMinorVersion*

Obtiene la versión secundaria de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
APIMinorVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property APIMinorVersion As Integer
```

VB.NET

5.5.1.1.1.3 *APIServicePackVersion*

Obtiene la versión de service pack de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
APIServicePackVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property APIServicePackVersion As Integer
```

VB.NET

5.5.1.1.1.4 *Is64Bit*

Devuelve **true** si el motor de MapForce Server es un ejecutable de 64 bits.

Firma

```
Is64Bit : Boolean
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property Is64Bit As Boolean
```

VB.NET

5.5.1.1.1.5 *LastExecutionMessage*

Obtiene el mensaje obtenido durante la última ejecución del comando **Run**.

Firma

```
LastExecutionMessage : String
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property LastExecutionMessage As String
```

VB.NET

5.5.1.1.1.6 *MajorVersion*

Obtiene la versión principal del producto como número de años a partir de 1998 (p. ej. "20" para Altova MapForce Server 2018).

Firma

```
MajorVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property MajorVersion As Integer
```

VB.NET

5.5.1.1.1.7 *MinorVersion*

Obtiene la versión secundaria del producto (p. ej. "2" para Altova MapForce Server 2018 r2).

Firma

```
MinorVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property MinorVersion As Integer
```

VB.NET

5.5.1.1.1.8 *ProductName*

Obtiene la versión secundaria del producto (p. ej. "para Altova MapForce Server").

Firma

```
ProductName : String
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property ProductName As String
```

VB.NET

5.5.1.1.1.9 *ProductNameAndVersion*

Obtiene el nombre completo y la versión del producto (p. ej. "Altova MapForce Server 2018 r2 sp1 (x64)").

Firma

```
ProductNameAndVersion : String
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property ProductNameAndVersion As String
```

VB.NET

5.5.1.1.10 *ServerPath*

Obtiene o establece la ruta de acceso del ejecutable de MapForce Server.

Firma

```
ServerPath : String
```

Firma genérica

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

C#

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

C++

```
Property ServerPath As String
```

VB.NET

5.5.1.1.11 *ServicePackVersion*

Obtiene la versión de service pack del producto (p. ej. "1" para Altova MapForce Server 2018 r2 sp1).

Firma

```
ServicePackVersion : Integer
```

Firma genérica

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

C#

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

C++

```
ReadOnly Property ServicePackVersion As Integer
```

VB.NET

5.5.1.1.1.12 WorkingDirectory

Obtiene o establece el directorio actual para los trabajos que están en ejecución (las rutas de acceso relativas se evaluarán en función del directorio de trabajo).

Firma

```
WorkingDirectory : String
```

Firma genérica

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

C#

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

C++

```
Property WorkingDirectory As String
```

VB.NET

5.5.1.1.2 Métodos

5.5.1.1.2.1 AddCredentialProperty

Añade una propiedad a la credencial actual (por ejemplo, el nombre de usuario, la contraseña o ambos). El primer argumento indica el nombre de la propiedad y el segundo su valor. Son nombres válidos de propiedad: `username`, `password`. En MapForce Server Advanced Edition también se admite el nombre de propiedad `oauth:token`.

Se llama a este método después de llamar a `BeginCredential()` y antes de llamar a `EndCredential()`.

Firma

```
AddCredentialProperty(in bstrName: System.String, in bstrValue: System.String) -> Void
```

Firma genérica

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

C#

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

C++

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

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrName	<code>System.String</code>	Indica el nombre de la propiedad de la credencial.
bstrValue	<code>System.String</code>	Indica el valor de la propiedad de la credencial.

Ejemplos

El fragmento de código siguiente ilustra cómo declarar una credencial llamada "mycredential" en C#. El nombre de la credencial debe ser el mismo que se le dio a la credencial en MapForce en el momento de crearla.

```
//Create a MapForce Server object
Altova.MapForceServer.Server objMFS = new Altova.MapForceServer.Server();

objMFS.BeginCredential("mycredential");
objMFS.AddCredentialProperty("username", "altova");
objMFS.AddCredentialProperty("password", "b45ax78!");
objMFS.EndCredential();
```

5.5.1.1.2.2 AddParameter

Asigna un valor al parámetro definido en la asignación. El primer argumento especifica el nombre del parámetro tal y como se definió en la asignación. El segundo argumento especifica el valor del parámetro.

Firma

```
AddParameter(in bstrName:String, in bstrValue:String) -> Void
```

Firma genérica

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

C#

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

C++

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

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrName	String	Especifica el nombre del parámetro.
bstrValue	String	Especifica el valor del parámetro.

5.5.1.1.2.3 BeginCredential

Crea una credencial nueva con el nombre dado como argumento. Si llama a este método también debe añadirle propiedades usando `AddCredentialProperty()` y cerrar la credencial llamando a `EndCredential()`.

Firma

```
BeginCredential(in bstrCredentialName:String) -> Void
```

Firma genérica

```
void BeginCredential(string bstrCredentialName)
```

C#

```
HRESULT BeginCredential([in] BSTR bstrCredentialName);
```

C++

```
Sub BeginCredential(ByVal bstrCredentialName As String)
```

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrCredentialName	String	Indica el nombre de la credencial como se definió en MapForce.

5.5.1.1.2.4 *ClearCredentialList*

Borra la lista de credenciales definida previamente. Todas las credenciales son válidas para el objeto mientras este dure. Llame a este método si necesita borrar explícitamente todas las credenciales que haya definido previamente.

Firma

```
ClearCredentialList() -> Void
```

Firma genérica

```
void ClearCredentialList()
```

C#

```
HRESULT ClearCredentialList();
```

C++

```
Sub ClearCredentialList()
```

VB.NET

5.5.1.1.2.5 *ClearOptions*

Borra la lista de opciones previamente establecida con el método `SetOption`. Todas las opciones establecidas con el método `SetOption` son válidas para la vida útil del objeto. Llame a este método si necesita borrar de forma explícita todas las opciones previamente establecidas.

Firma

```
ClearOptions() -> Void
```

Firma genérica

```
void ClearOptions()
```

C#

```
HRESULT ClearOptions();
```

C++

```
Sub ClearOptions()
```

VB.NET

5.5.1.1.2.6 *ClearParameterList*

Borra la lista de parámetros que se definieron previamente con el método `AddParameter`.

Firma

```
ClearParameterList() -> Void
```

Firma genérica

```
void ClearParameterList()
```

C#

```
HRESULT ClearParameterList();
```

C++

```
Sub ClearParameterList()
```

VB.NET

5.5.1.1.2.7 *EndCredential*

Cierra un objeto de credencial creado previamente con el método `BeginCredential`.

Firma

```
EndCredential() -> Void
```

Firma genérica

```
void EndCredential()
```

C#

```
HRESULT EndCredential();
```

C++

```
Sub EndCredential()
```

VB.NET

5.5.1.1.2.8 *GetOutputParameter*

Obtiene la cadena de salida generada por el último comando `run`. Devuelve `null` si no se generó ningún resultado. Esta función requiere un parámetro de cadena que identifique el nombre que tiene el componente de salida en Altova MapForce.

Firma

```
GetOutputParameter(in bstrName:String) -> String
```

Firma genérica

```
string GetOutputParameter(string bstrName)
```

C#

```
HRESULT GetOutputParameter([in] BSTR bstrName, [out, retval] BSTR* pbstrValue );
```

C++

```
Function GetOutputParameter(bstrName As String) As String
```

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrName	String	Especifica el nombre del componente de salida como aparece en MapForce. Este nombre se muestra en la barra del título de cada componente en la asignación (o si hace clic con el botón derecho en el encabezado del componente y selecciona Propiedades).

5.5.1.1.2.9 *Run*

Ejecuta un archivo de ejecución de MapForce Server Execution (archivo `.mfx`). Devuelve **true** si se ejecuta correctamente. De lo contrario devuelve **false**.

Firma

```
Run(in bstrMappingPath:String) -> Boolean
```

Firma genérica

```
Run(string bstrMappingPath)
```

C#

```
Run( [in] BSTR bstrMappingPath, [out, retval] VARIANT_BOOL* pbSuccess );
```

C++

```
Function Run(ByVal bstrMappingPath As String) As Boolean
```

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrMappingPath	String	Especifica la ruta de acceso al archivo de ejecución de MapForce Server (.mfx). Si indica una ruta relativa, esta lo es con respecto al directorio de trabajo. Puede definir el directorio de trabajo desde la propiedad <code>WorkingDirectory</code> .

5.5.1.1.2.10 *SetOption*

Establece una opción antes de ejecutar la asignación. El primer argumento especifica el nombre de la opción, mientras que el segundo argumento especifica el valor de la opción. Este método es especialmente útil para asignaciones diseñadas para que consuman recursos globales (véase [Recursos globales de Altova](#)³⁸). Estas son las opciones compatibles actualmente:

- **globalresourcefile (or gr)**: una ruta de acceso a los archivos de los recursos globales. (Si se indica esta opción también se debe indicar un nombre de configuración de los recursos globales, véase el siguiente punto).
- **globalresourcefile (or gr)**: un nombre de configuración de los recursos globales. (Si se indica esta opción también se debe indicar una ruta de acceso a los archivos de los recursos globales, véase el siguiente punto).
- **catalog**: la ruta de acceso a un archivo **RootCatalog.xml** personalizado. Esta opción permite indicar un archivo de catálogo personalizado para resolver las URL que usa la asignación. El catálogo predeterminado está en el subdirectorio etc del directorio del programa de instalación.

- **taxonomy-package**: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación.
- **taxonomy-packages-config-file**: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación.

Todas las opciones establecidas son válidas para la vida útil del objeto. Si establece una opción dos veces con el mismo nombre se sobrescribirá la opción anterior. Para borrar de forma explícita todas las opciones, llame al método `ClearOptions()`.

Firma

```
SetOption(in bstrName:String, in bstrValue:String) -> Void
```

Firma genérica

```
void SetOption(ByVal bstrName As String, ByVal bstrValue As String)
```

C#

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

C++

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

VB.NET

Parámetros

Nombre	Tipo	Descripción
bstrName	String	Indica el nombre de la opción que se quiere añadir.
bstrValue	String	Indica el valor del servicio web que se quiere añadir.

5.5.1.1.2.11 StopServerProcess

Este método detiene de forma explícita el proceso conectado con el objeto COM pero sin liberar el objeto. El proceso se detiene de forma implícita cuando se libera el objeto COM.

Firma

```
StopServerProcess() -> System.Void
```

Firma genérica

```
void StopServerProcess()
```

C#

```
Sub StopServerProcess()
```

VB.NET

5.6 Referencia de la API (Java)

Este apartado ofrece una referencia general de los elementos de la API de MapForce Server (como las clases y los métodos) que sirven para escribir código para la plataforma Java.

5.6.1 Clases

5.6.1.1 MapForceServer

La clase `MapForceServer` crea una instancia de objeto de MapForce Server nueva y ofrece acceso al motor de MapForce Server.

Métodos

Nombre	Descripción
addCredentialPropertiesFromMap ¹¹⁴	Añade propiedades de un esquema de propiedades a la credencial actual. Este método toma como argumento un esquema de propiedades de credencial (<i>property_name</i> , <i>property_value</i>). Se llama a este método después de llamar a <code>beginCredential()</code> y antes de llamar a <code>endCredential()</code> . En vez de llamar a este método también puede llamar a <code>AddCredentialProperty()</code> .
addCredentialProperty ¹¹⁴	Añade una propiedad a la credencial actual (por ejemplo, el nombre de usuario, la contraseña o ambos). El primer argumento indica el nombre de la propiedad y el segundo su valor. Son nombres válidos de propiedad: <code>username</code> , <code>password</code> . En MapForce Server Advanced Edition también se admite el nombre de propiedad <code>oauth:token</code> . Se llama a este método después de llamar a <code>beginCredential()</code> y antes de llamar a <code>endCredential()</code> . En vez de llamar a este método también puede llamar a <code>addCredentialProperty()</code>
addParameter ¹¹⁵	Asigna un valor al parámetro definido en la asignación.
beginCredential ¹¹⁶	Crea una credencial nueva con el nombre dado como argumento. Si llama a este método también debe añadirle propiedades usando <code>addCredentialProperty()</code> or <code>addCredentialPropertiesFromMap()</code> y cerrar la credencial llamando a <code>endCredential()</code> .
clearCredentialList ¹¹⁶	Borra la lista de credenciales definida previamente. Todas las credenciales son válidas para el objeto mientras este dure. Llame a este método si necesita borrar explícitamente todas las credenciales que haya definido previamente.

Nombre	Descripción
clearOptions ¹¹⁶	Borra la lista de opciones previamente establecida con el método <code>setOption</code> . Todas las opciones establecidas con el método <code>setOption</code> son válidas para la vida útil del objeto. Llame a este método si necesita borrar de forma explícita todas las opciones previamente establecidas.
clearParameterList ¹¹⁷	Borra la lista de parámetros que se definieron previamente con el método <code>addParameter</code> .
endCredential ¹¹⁷	Cierra un objeto de credencial creado previamente con el método <code>beginCredential</code> .
getAPIMajorVersion ¹¹⁷	Obtiene la versión principal de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
getAPIMinorVersion ¹¹⁷	Obtiene la versión secundaria de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
getAPIServicePackVersion ¹¹⁸	Obtiene la versión de service pack de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.
getLastExecutionMessage ¹¹⁸	Obtiene el mensaje obtenido durante la última ejecución del comando <code>run</code> .
getMajorVersion ¹¹⁸	Obtiene la versión principal del producto como número de años a partir de 1998 (p. ej. "20" para Altova MapForce Server 2018).
getMinorVersion ¹¹⁹	Obtiene la versión secundaria del producto (p. ej. "2" para Altova MapForce Server 2018 r2).
getOutputParameter ¹¹⁹	Obtiene la cadena de salida generada por el último comando <code>run</code> . Devuelve <code>null</code> si no se generó ningún resultado. Esta función requiere un parámetro de cadena que identifique el nombre que tiene el componente de salida en Altova MapForce.
getProductName ¹²⁰	Obtiene la versión secundaria del producto (p. ej. "para Altova MapForce Server").
getProductNameAndVersion ¹²⁰	Obtiene el nombre completo y la versión del producto (p. ej. "Altova MapForce Server 2018 r2 sp1 (x64)").
getServicePackVersion ¹²⁰	Obtiene la versión de service pack del producto (p. ej. "1" para Altova MapForce Server 2018 r2 sp1 (x64)).
is64bit ¹²¹	Devuelve true si el motor de MapForce Server es un ejecutable de 64 bits.
run ¹²¹	Ejecuta un archivo de ejecución de MapForce Server Execution (archivo <code>.mfx</code>). Devuelve true si se ejecuta correctamente. De lo contrario devuelve false .

Nombre	Descripción
setOption ¹²²	<p>Establece una opción antes de ejecutar la asignación. El primer argumento especifica el nombre de la opción, mientras que el segundo argumento especifica el valor de la opción. Este método es especialmente útil para asignaciones diseñadas para que consuman recursos globales (véase Recursos globales de Altova ³⁸). Estas son las opciones compatibles actualmente:</p> <ul style="list-style-type: none"> • globalresourcefile (o gr): una ruta de acceso a los archivos de los recursos globales. (Si se indica esta opción también se debe indicar un nombre de configuración de los recursos globales, véase el siguiente punto). • globalresourcefile (o gr): un nombre de configuración de los recursos globales. (Si se indica esta opción también se debe indicar una ruta de acceso a los archivos de los recursos globales, véase el siguiente punto). • catalog: la ruta de acceso a un archivo RootCatalog.xml personalizado. Esta opción permite indicar un archivo de catálogo personalizado para resolver las URL que usa la asignación. El catálogo predeterminado está en el subdirectorio etc del directorio del programa de instalación. • taxonomy-package: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación. • taxonomy-packages-config-file: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación. <p>Todas las opciones establecidas son válidas para la vida útil del objeto. Si establece una opción dos veces con el mismo nombre se sobrescribirá la opción anterior. Para borrar de forma explícita todas las opciones, llame al método <code>clearOptions()</code>.</p>
setServerPath ¹²³	<p>Establece la ruta de acceso del ejecutable de MapForce Server.</p>
setWorkingDirectory ¹²³	<p>Establece el directorio actual para los trabajos que están en ejecución (las rutas de acceso relativas se evaluarán en función del directorio de trabajo).</p>
stop ¹²⁴	<p>Detiene el proceso</p>

Ejemplos

Para ver un ejemplo de cómo crear una nueva instancia de MapForce Server en Java, consulte el [Ejemplo para Java](#) ⁸⁴.

5.6.1.1.1 Métodos

5.6.1.1.1.1 *addCredentialPropertiesFromMap*

Añade propiedades de un esquema de propiedades a la credencial actual. Este método toma como argumento un esquema de propiedades de credencial (*property_name*, *property_value*). Se llama a este método después de llamar a `beginCredential()` y antes de llamar a `endCredential()`. En vez de llamar a este método también puede llamar a `AddCredentialProperty()`.

Firma

```
addCredentialPropertiesFromMap(arg0:Map) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	Map	Un esquema que proporciona el nombre de la propiedad de la credencial y su valor.

Ejemplos

El fragmento de código siguiente ilustra cómo añadir una credencial llamada "mycredential" al contexto actual usando el método `addCredentialPropertiesFromMap`.

```
//Create a MapForce Server object
com.altova.mapforceserver.MapForceServer objMFS = new com.altova.mapforceserver.MapForceServer();

objMFS.beginCredential("mycredential");
java.util.Map<String, String> credentialMap = new java.util.HashMap<String,String>();
credentialMap.put("username", "altova");
credentialMap.put("password", "b45ax78!");
objMFS.addCredentialPropertiesFromMap(credentialMap);
objMFS.endCredential();
```

5.6.1.1.1.2 *addCredentialProperty*

Añade una propiedad a la credencial actual (por ejemplo, el nombre de usuario, la contraseña o ambos). El primer argumento indica el nombre de la propiedad y el segundo su valor. Son nombres válidos de propiedad: `username`, `password`. En MapForce Server Advanced Edition también se admite el nombre de propiedad `oauth:token`.

Se llama a este método después de llamar a `beginCredential()` y antes de llamar a `endCredential()`. vez de llamar a este método también puede llamar a `addCredentialProperty()`

Firma

```
addCredentialProperty(arg0:String, arg1:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	El nombre de la propiedad de la credencial (por ejemplo "nombre de usuario" o "contraseña").
arg1	<code>String</code>	El valor de la propiedad de la credencial.

Ejemplos

El fragmento de código siguiente ilustra cómo declarar una credencial llamada "mycredential". El nombre de la credencial debe ser el mismo que se le dio a la credencial en MapForce en el momento de crearla.

```
//Create a MapForce Server object
com.altova.mapforceserver.MapForceServer objMFS = new com.altova.mapforceserver.MapForceServer();

objMFS.beginCredential("mycredential");
objMFS.addCredentialProperty("username", "altova");
objMFS.addCredentialProperty("password", "b45ax78!");
objMFS.endCredential();
```

5.6.1.1.1.3 addParameter

Asigna un valor al parámetro definido en la asignación.

Firma

```
addParameter(arg0:String, arg1:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
--------	------	-------------

arg0	<code>String</code>	Especifica el nombre del parámetro.
arg1	<code>String</code>	Especifica el valor del parámetro.

5.6.1.1.1.4 *beginCredential*

Crea una credencial nueva con el nombre dado como argumento. Si llama a este método también debe añadirle propiedades usando `addCredentialProperty()` or `addCredentialPropertiesFromMap()` y cerrar la credencial llamando a `endCredential()`.

Firma

```
beginCredential(arg0:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	El nombre de la credencial tal y como se definió en MapForce.

5.6.1.1.1.5 *clearCredentialList*

Borra la lista de credenciales definida previamente. Todas las credenciales son válidas para el objeto mientras este dure. Llame a este método si necesita borrar explícitamente todas las credenciales que haya definido previamente.

Firma

```
clearCredentialList() -> void
```

Firma genérica

5.6.1.1.1.6 *clearOptions*

Borra la lista de opciones previamente establecida con el método `setOption()`. Todas las opciones establecidas con el método `setOption()` son válidas para la vida útil del objeto. Llame a este método si necesita borrar de forma explícita todas las opciones previamente establecidas.

Firma

```
clearOptions() -> void
```

Firma genérica

5.6.1.1.1.7 *clearParameterList*

Borra la lista de parámetros que se definieron previamente con el método `addParameter`.

Firma

```
clearParameterList() -> void
```

Firma genérica

5.6.1.1.1.8 *endCredential*

Cierra un objeto de credencial creado previamente con el método `beginCredential`.

Firma

```
endCredential() -> void
```

Firma genérica

5.6.1.1.1.9 *getAPIMajorVersion*

Obtiene la versión principal de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
getAPIMajorVersion() -> int
```

Firma genérica

5.6.1.1.1.10 *getAPIMinorVersion*

Obtiene la versión secundaria de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
getAPIMinorVersion() -> int
```

Firma genérica

5.6.1.1.11 *getAPIServicePackVersion*

Obtiene la versión de service pack de la API de MapForce Server. Si la API está conectada a otro servidor, la versión puede diferir de la versión del producto.

Firma

```
getAPIServicePackVersion() -> int
```

Firma genérica

5.6.1.1.12 *getLastExecutionMessage*

Obtiene el mensaje obtenido durante la última ejecución del comando `run`.

Firma

```
getLastExecutionMessage() -> java.lang.String
```

Firma genérica

5.6.1.1.13 *getMajorVersion*

Obtiene la versión principal del producto como número de años a partir de 1998 (p. ej. "20" para Altova MapForce Server 2018).

Firma

```
getMajorVersion() -> int
```

Firma genérica

5.6.1.1.1.14 *getMinorVersion*

Obtiene la versión secundaria del producto (p. ej. "2" para Altova MapForce Server 2018 r2).

Firma

```
getMinorVersion() -> int
```

Firma genérica

5.6.1.1.1.15 *getOutputParameter*

Obtiene la cadena de salida generada por el último comando `run`. Devuelve null si no se generó ningún resultado. Esta función requiere un parámetro de cadena que identifique el nombre que tiene el componente de salida en Altova MapForce.

Firma

```
getOutputParameter(arg0:String) -> java.lang.String
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	Especifica el nombre del componente de salida como aparece en MapForce. Este nombre se muestra en la barra del título de cada componente en la asignación (o si hace clic con el botón derecho en el encabezado del componente y selecciona Propiedades).

5.6.1.1.1.16 *getProductName*

Obtiene la versión secundaria del producto (p. ej. "para Altova MapForce Server").

Firma

```
getProductName() -> java.lang.String
```

Firma genérica

5.6.1.1.1.17 *getProductNameAndVersion*

Obtiene el nombre completo y la versión del producto (p. ej. "Altova MapForce Server 2018 r2 sp1 (x64)").

Firma

```
getProductNameAndVersion() -> java.lang.String
```

Firma genérica

5.6.1.1.1.18 *getServerPath*

Obtiene la ruta de acceso al archivo binario ejecutable del servidor.

Firma

```
getServerPath() -> java.lang.String
```

Firma genérica

5.6.1.1.1.19 *getServicePackVersion*

Obtiene la versión de service pack del producto (p. ej. "1" para Altova MapForce Server 2018 r2 sp1 (x64)).

Firma

```
getServicePackVersion() -> int
```

Firma genérica

5.6.1.1.1.20 *getWorkingDirectory*

Obtiene el directorio de trabajo actual.

Firma

```
getWorkingDirectory() -> java.lang.String
```

Firma genérica

5.6.1.1.1.21 *is64bit*

Devuelve **true** si el motor de MapForce Server es un ejecutable de 64 bits.

Firma

```
is64bit() -> boolean
```

Firma genérica

5.6.1.1.1.22 *run*

Ejecuta un archivo de ejecución de MapForce Server Execution (archivo .mfx). Devuelve **true** si se ejecuta correctamente. De lo contrario devuelve **false**.

Firma

```
run(arg0:String) -> boolean
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	String	Especifica la ruta de acceso al archivo de ejecución de MapForce Server (.mfx).

5.6.1.1.1.23 *setOption*

Establece una opción antes de ejecutar la asignación. El primer argumento especifica el nombre de la opción, mientras que el segundo argumento especifica el valor de la opción. Este método es especialmente útil para asignaciones diseñadas para que consuman recursos globales (véase [Recursos globales de Altova](#)³⁸). Estas son las opciones compatibles actualmente:

- **globalresourcefile (o gr)**: una ruta de acceso a los archivos de los recursos globales. (Si se indica esta opción también se debe indicar un nombre de configuración de los recursos globales, véase el siguiente punto).
- **globalresourcefile (o gr)**: un nombre de configuración de los recursos globales. (Si se indica esta opción también se debe indicar una ruta de acceso a los archivos de los recursos globales, véase el siguiente punto).
- **catalog**: la ruta de acceso a un archivo **RootCatalog.xml** personalizado. Esta opción permite indicar un archivo de catálogo personalizado para resolver las URL que usa la asignación. El catálogo predeterminado está en el subdirectorio etc del directorio del programa de instalación.
- **taxonomy-package**: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación.
- **taxonomy-packages-config-file**: la ruta a un paquete de taxonomías XBRL personalizado, si es necesario para la asignación.

Todas las opciones establecidas son válidas para la vida útil del objeto. Si establece una opción dos veces con el mismo nombre se sobrescribirá la opción anterior. Para borrar de forma explícita todas las opciones, llame al método `clearOptions()`.

Firma

```
setOption(arg0:String, arg1:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	Indica el nombre de la opción que se quiere añadir.
arg1	<code>String</code>	Indica el valor del servicio web que se quiere añadir.

5.6.1.1.1.24 *setServerPath*

Establece la ruta de acceso del ejecutable de MapForce Server.

Firma

```
setServerPath(arg0:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	Establece la ruta de acceso al ejecutable de MapForce Server.

5.6.1.1.1.25 *setWorkingDirectory*

Establece el directorio actual para los trabajos que están en ejecución (las rutas de acceso relativas se evaluarán en función del directorio de trabajo).

Firma

```
setWorkingDirectory(arg0:String) -> void
```

Firma genérica

Parámetros

Nombre	Tipo	Descripción
arg0	<code>String</code>	Establece la ruta de acceso al directorio de trabajo.

5.6.1.1.1.26 *stop*

Detiene el proceso del servidor.

Firma

```
stop() -> void
```

Firma genérica

5.6.1.2 MapForceServerException

La clase `MapForceServerException` ofrece acceso mediante programación a las excepciones que emite la clase `MapForceServer`.

```
public class MapForceServerException extends Exception
```

6 Gestión de certificados digitales

La gestión de certificados digitales es un aspecto crucial del intercambio seguro de datos entre equipos clientes y servidores web. Como las asignaciones no solo se pueden ejecutar en Windows con Altova MapForce, sino que también se pueden ejecutar en servidores Windows, Linux y macOS con MapForce Server (por separado o como parte de FlowForce Server), en esta sección explicamos cómo puede gestionar certificados HTTPS en las distintas plataformas.

En el contexto del protocolo HTTPS es importante distinguir entre certificados de cliente y de servidor.

Certificados de servidor

Un certificado de servidor identifica un servidor como entidad de confianza para una aplicación cliente como MapForce. El certificado de servidor puede estar firmado digitalmente por una entidad de certificación comercial o por la organización del usuario. En ambos casos, mientras se diseña la asignación en MapForce tiene la opción de especificar:

- Si se debe comprobar el certificado de servidor.
- Si la solicitud debe continuar incluso si se detectan diferencias entre el nombre del certificado y el nombre de host.

Esta configuración puede editarse en el cuadro de diálogo "Configuración de seguridad HTTPS" de MapForce. Si habilita la opción para comprobar el certificado de servidor, debe tener en cuenta que:

- Si llama a un servidor web cuyo certificado está firmado por una entidad de certificación de confianza, es probable que el sistema operativo ya esté configurado para confiar en el certificado de servidor, sin necesidad de realizar ajustes en el sistema.
- Si llama a un servidor web cuyo certificado está autofirmado (p.ej. un servidor de red local de su organización), deberá configurar el sistema operativo para que confíe en el certificado.

En la mayoría de los casos, puede comprobar el nivel de confianza que hay entre el sistema operativo y el servidor web con solo introducir la URL del servicio web en la barra de direcciones del explorador. Si el servidor no es de confianza o el sistema operativo no está configurado para confiar en el servidor, el explorador mostrará un mensaje parecido a "Esta conexión no es de confianza" o "Hay un problema con el certificado de este sitio web". Sin embargo, si el explorador utiliza una base de datos de certificados que no es la del sistema operativo (p. ej. Firefox 35.0.1 en Ubuntu 14.04) entonces no podrá comprobar el nivel de confianza con el explorador web.

En Windows puede establecer confianza con el servidor siguiendo las instrucciones del explorador e importando/installando los certificados necesarios en el almacén *Entidades de certificación raíz de confianza* (véase [Confiar en certificados servidor en Windows](#)¹³⁰). En macOS puede realizar una operación parecida en Acceso a Llaveros (véase [Confiar en certificados servidor en macOS](#)¹²⁹). Para Linux consulte el apartado [Confiar en certificados servidor en Linux](#)¹²⁸.

Certificados de cliente

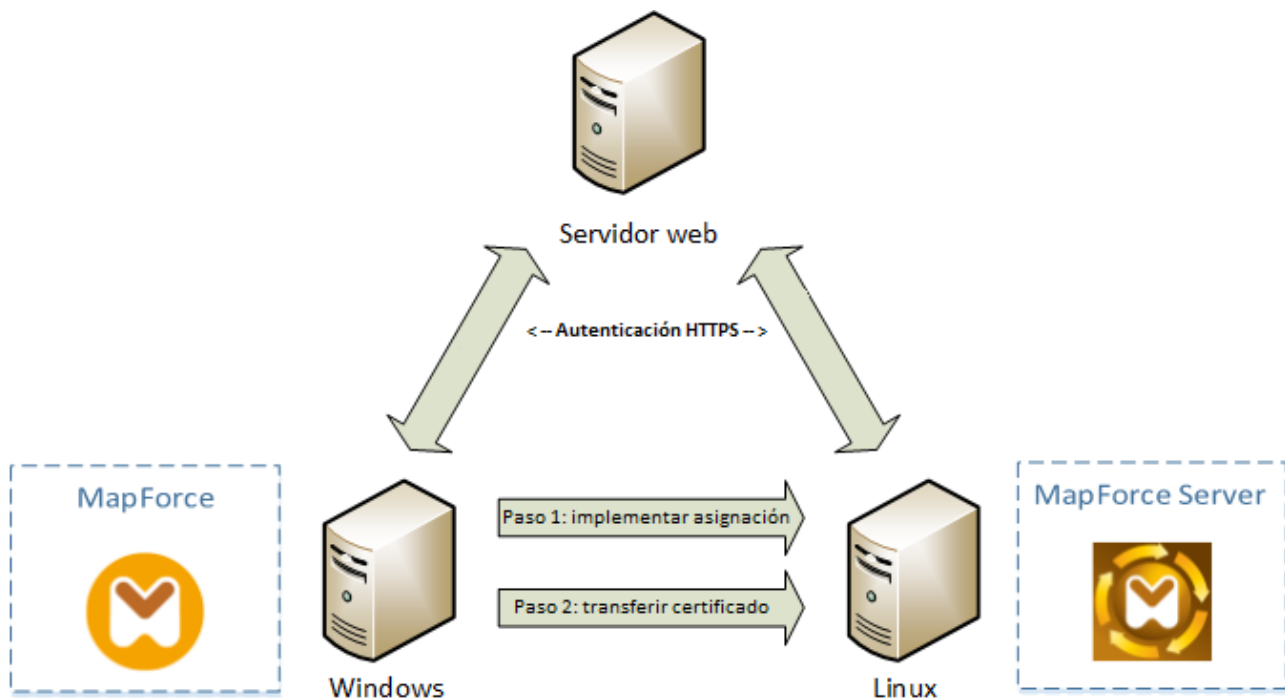
Mientras que los certificados de servidor sirven para identificar un servidor como entidad de confianza, los certificados de cliente sirven principalmente para autenticar al autor de la llamada con el servidor web. Si tiene pensado llamar a un servidor web que requiere certificados de cliente, quizás sea recomendable ponerse en contacto con el administrador del servidor web y solicitar instrucciones para configurar el cliente correctamente. Por ejemplo, en el caso de IIS, el servidor web puede estar configurado para ocuparse de conexiones HTTPS y certificados de cliente de varias maneras:

- Puede exigir HTTPS y omitir el certificado de cliente
- Puede exigir HTTPS y aceptar el certificado de cliente
- Puede exigir HTTPS y también el certificado de cliente

El hecho de que la solicitud al servicio web funcione o no dependerá tanto de la configuración del servidor web como de la aplicación cliente. Por ejemplo, si el servidor web está configurado para exigir un certificado de cliente, para que funcione la llamada la aplicación que efectúa la llamada debe presentar un certificado de cliente válido.

Desde el punto de vista del sistema MapForce, lo mismo ocurre cuando las asignaciones incluyen llamadas a servicios web por HTTPS. Para que estas asignaciones se ejecuten correctamente, es necesario que el servidor web esté configurado para aceptar o exigir el certificado de cliente y que el sistema operativo donde se ejecuta la asignación suministre el certificado de cliente adecuado al servidor web.

En el diagrama que aparece a continuación puede ver un ejemplo donde se transfiere un certificado de cliente utilizado por MapForce a un servidor Linux que ejecuta MapForce Server. Una vez transferido al sistema operativo, el certificado se podrá usar en MapForce Server para autenticarse en el servidor web y ejecutar la asignación correctamente.



Implementación de asignaciones con certificados de cliente en otros equipos

MapForce es capaz de usar cifrado TLS (*Seguridad de la capa de transporte*) además de HTTP, que sustituye al protocolo SSL (*Capa de sockets seguros*). Tenga en cuenta que si la implementación cliente o el servidor no son compatibles con TSL es posible que se recurra a SSL.

Para garantizar la compatibilidad con llamadas web con autenticación de certificado de cliente en varias plataformas, MapForce y MapForce Server requieren que la gestión de certificados esté implementada en cada plataforma. Esto garantiza que la gestión de certificados siempre esté dentro del ámbito del sistema operativo. Cada sistema operativo ofrece diferentes funciones de gestión de certificados, tal y como muestra la tabla.

Plataforma	Gestión de certificados e implementación
Windows	<p>Puede gestionar certificados con el complemento <i>Certificados</i> (véase Acceso a los almacenes de certificados en Windows¹³¹).</p> <p>La biblioteca <i>SChannel</i> ofrece compatibilidad con TLS.</p>
Linux	<p>Puede gestionar certificados con la herramienta y biblioteca OpenSSL de la línea de comandos. Si el equipo Linux que ejecuta MapForce Server todavía no es compatible con OpenSSL, deberá descargarlo e instalarlo para poder gestionar certificados.</p> <p>La biblioteca OpenSSL ofrece compatibilidad con TLS (https://www.openssl.org/).</p>
macOS	<p>Puede gestionar certificados con el administrador <i>Keychain Access Manager</i>, disponible en Finder > Applications > Utilities.</p> <p>La biblioteca nativa Secure Transport ofrece compatibilidad con TLS.</p>

Si ejecuta la asignación en un sistema operativo Windows donde ya puede consumir correctamente el mismo servicio web al que tiene pensado llamar desde MapForce, lo normal es que no necesite realizar cambios en la configuración de certificados (véase [Certificados de cliente en Windows](#)¹⁴²). Sin embargo, si diseña asignaciones con MapForce en un equipo Windows y después las implementa en otro equipo (con otro sistema operativo), el certificado de cliente no se almacena ni se copia junto con el paquete implementado. Para que la llamada a servicio web y la asignación se ejecuten correctamente, el certificado de cliente debe existir también en el sistema operativo de destino.

Para transferir un certificado del sistema Windows a otro equipo Windows, exporte el certificado (con clave privada) del sistema de origen (véase [Exportar certificados de Windows](#)¹³²). Después debe importar el mismo certificado al almacén **Usuario actual/Personal** del sistema operativo de destino (véase [Certificados de cliente en Windows](#)¹⁴²).

Consulte los apartados [Certificados de cliente en Linux](#)¹³⁸ y [Certificados de cliente en macOS](#)¹⁴⁰ para obtener más información sobre cómo transferir certificados de cliente a plataformas Linux y macOS.

6.1 Confiar en certificados servidor en Linux

En Linux puede importar un certificado de confianza en el almacén de certificados del sistema como explicamos a continuación. Siga estas instrucciones solamente si está seguro de la autenticidad del certificado en el que desea confiar.

En Debian y Ubuntu, siga estos pasos:

1. Copie el archivo de certificado del servidor web en este directorio.

```
sudo cp /home/downloads/server_cert.crt /usr/local/share/ca-certificates/
```

2. Actualice el almacén de certificados de la siguiente manera:

```
sudo update-ca-certificates
```

En Cent OS, siga estos pasos:

1. Instale el paquete `ca-certificates`:

```
yum install ca-certificates
```

2. Habilite la función de configuración dinámica de entidades de certificación:

```
update-ca-trust enable
```

3. Copie el certificado de servidor en este directorio:

```
cp server_cert.crt /etc/pki/ca-trust/source/anchors/
```

4. Utilice este comando:

```
update-ca-trust extract
```

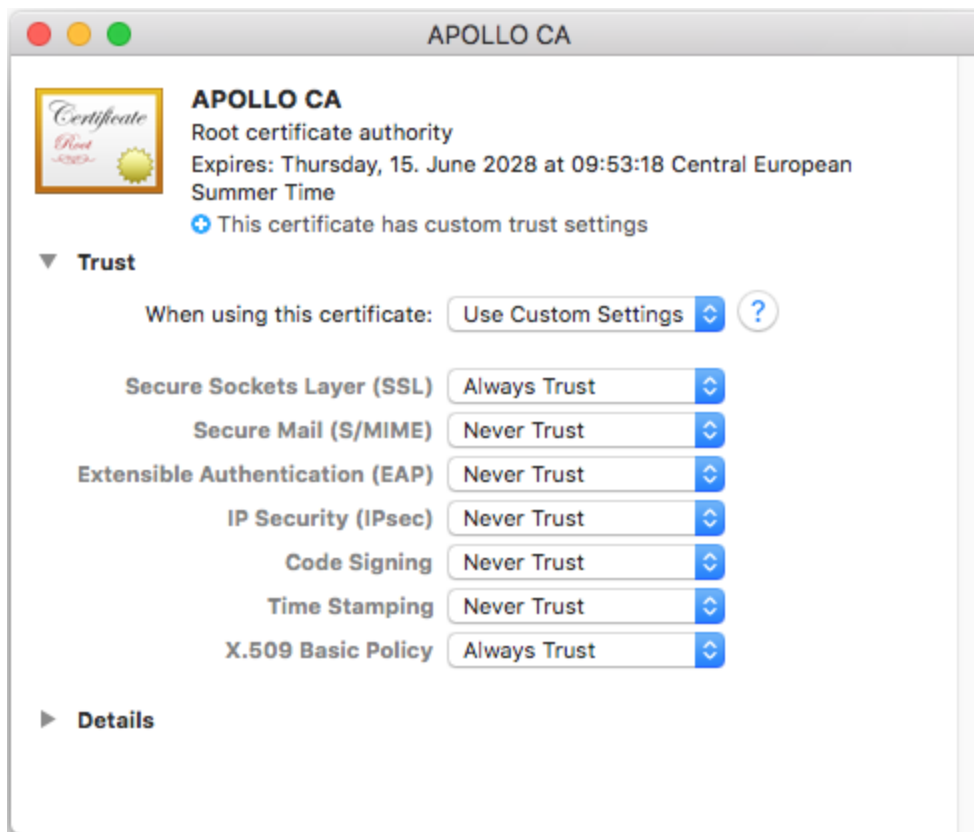
En los casos en que necesite acceder al servidor solo mediante el navegador, basta con que importe el certificado en el almacén de certificados del navegador. Las instrucciones exactas pueden ser distintas para cada navegador. Por ejemplo, en Firefox 59.0.2 puede hacer lo siguiente:

1. En **Opciones | Privacidad & seguridad**, haga clic en **Ver certificados**.
2. En la pestaña **Autoridades**, haga clic en **Importar** y navegue hasta el archivo del certificado raíz que creó previamente.
3. Cuando la aplicación lo solicite, seleccione **Confiar en este certificado al identificar páginas web** y haga clic en **Aceptar**.

6.2 Confiar en certificados servidor en OS X / macOS

En macOS puede importar un certificado de confianza en el Acceso a llaveros como explicamos a continuación:

1. Ejecute el Acceso a llaveros.
2. Haga clic en **Sistema** y después en **Certificados**.
3. En el menú **Archivo**, haga clic en **Importar elementos**.
4. Navegue hasta el certificado de confianza y haga clic en **Abrir**.
5. Cuando la aplicación lo solicite, introduzca la contraseña del Acceso a llaveros y haga clic en **Modificar llavero**.
6. Haga doble clic en el certificado, expanda la sección *Confiar* y seleccione **Confiar siempre**.



6.3 Confiar en certificados servidor en Windows

En Windows puede importar un certificado de confianza en el sistema de certificados como explicamos a continuación:

1. Abra el almacén de certificados de Windows de la *cuenta del equipo*, véase [Acceso al almacén de certificados de confianza de Windows](#)¹³¹.
2. Expanda la carpeta *Entidades de certificación raíz de confianza* de la estructura *Certificados (equipo local)*, haga clic con el botón derecho en **Certificados** y seleccione **Todas las tareas | Importar**. Después, siga las instrucciones del asistente de importación de certificados.

Para más información consulte este artículo (en inglés): [Importar un certificado en el sitio web de Microsoft](#).

6.4 Acceso a los almacenes de certificados en Windows

En Windows puede gestionar certificados en el complemento Certificados de Microsoft Management Console (MMC) tanto para su cuenta de usuario como para la cuenta del equipo.

Para abrir el complemento Certificados para el *usuario actual de Windows*, ejecute el siguiente comando en la línea de comandos:

```
certmgr.msc
```

Para abrir el complemento Certificados para la *cuenta del equipo*, siga estos pasos:

1. Ejecute `mmc` en la línea de comandos.
2. En el menú **File** (archivo) del MMC haga clic en **Add/Remove Snap-in** (agregar/eliminar complementos).
3. Haga clic en **Certificates** (certificados) y después en **Add** (agregar).
4. Haga clic en **Computer account** (cuenta del equipo) y después en **Next** (siguiente).
5. Haga clic en **Local computer** (equipo local) y después en **Finish** (finalizar).

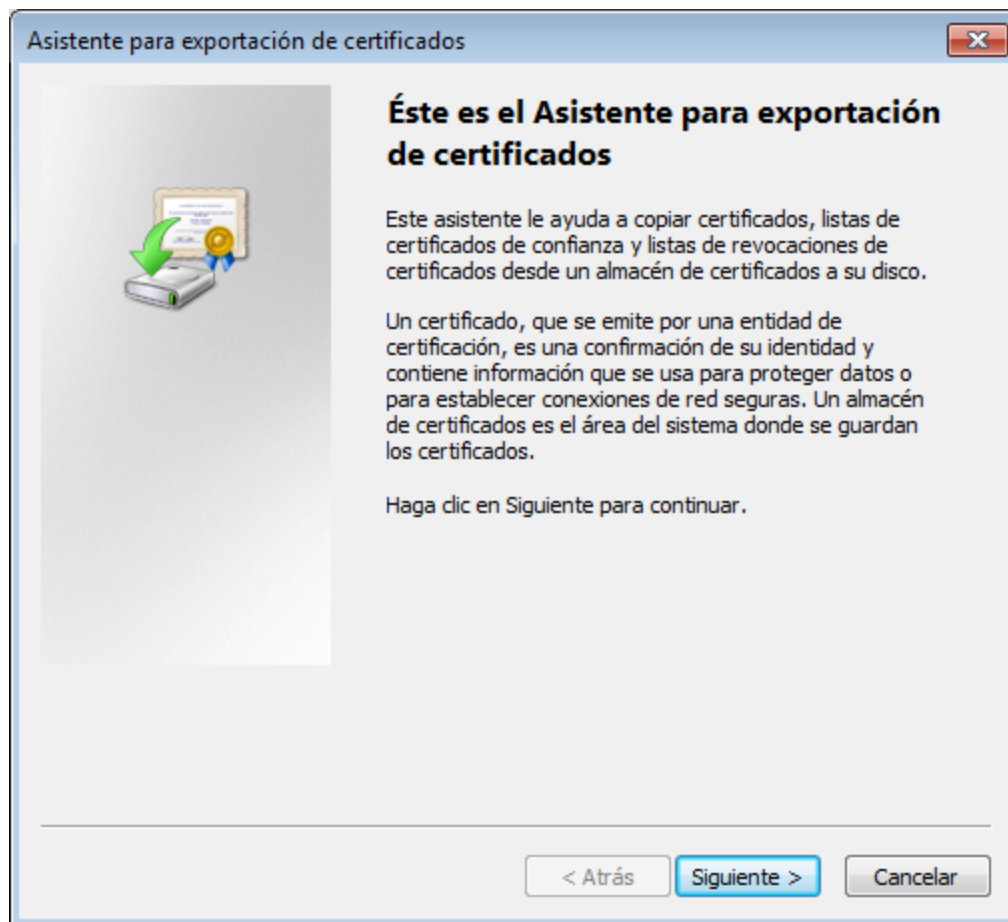
6.5 Exportar certificados de Windows

En el caso de asignaciones que llaman a servicios web por HTTPS e implementadas en servidores macOS o Linux que ejecutan MapForce Server o FlowForce Server, es necesario que el equipo no Windows tenga acceso al mismo certificado de cliente que se utilizó en Windows para diseñar y probar la asignación.

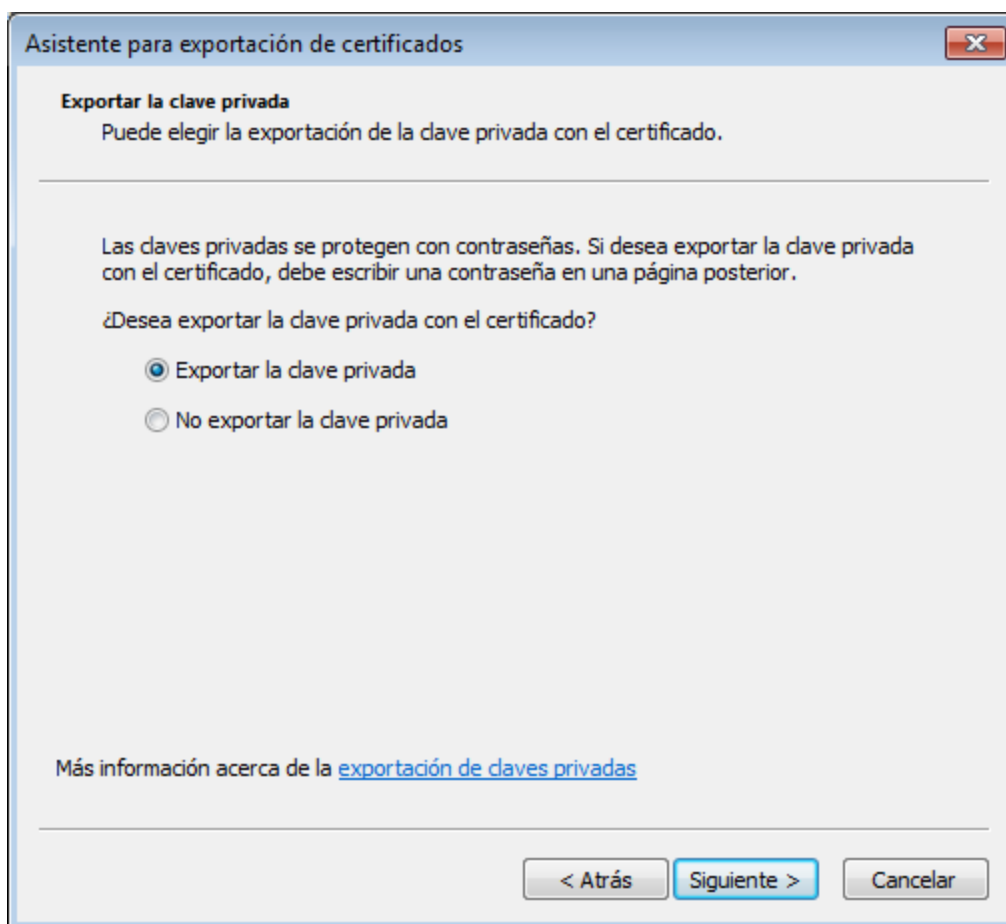
Para ejecutar estas asignaciones en sistemas operativos no Windows con MapForce Server, basta con exportar el certificado necesario con clave privada desde Windows e importarlo al sistema operativo de destino.

Para exportar un certificado con clave privada desde Windows:

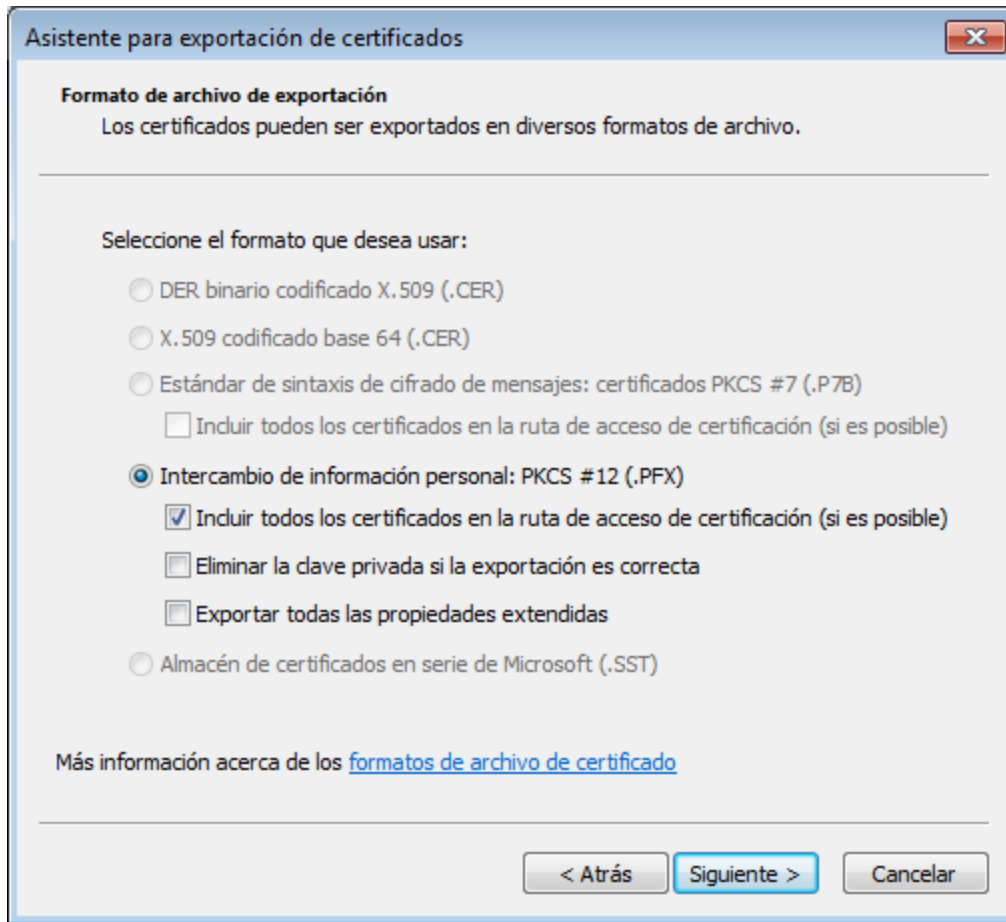
1. En Windows abra el complemento *Certificados* (véase [Acceso a los almacenes de certificados en Windows](#)¹³¹).
2. Haga clic con el botón derecho en el certificado que desea exportar, apunte a **Todas las tareas** y después haga clic en **Exportar**.
3. Ahora haga clic en **Siguiente**.



4. Elija la opción que permite exportar el certificado junto con su clave privada y después haga clic en **Siguiente**.

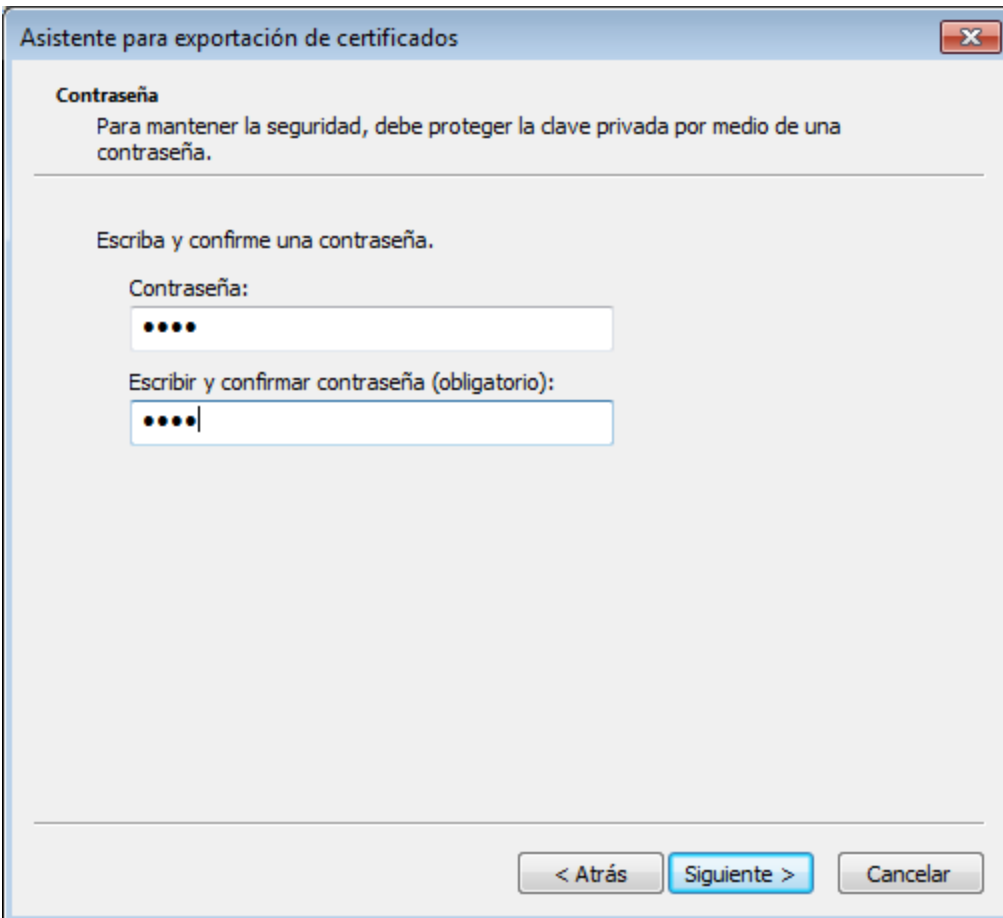


5. Elija el formato de archivo *Personal Information Exchange - PKCS #12 (.pfx)* y después haga clic en **Siguiete**.



Nota: compruebe que la opción *Eliminar la clave privada si la exportación es correcta* **no está marcada**. De lo contrario no podrá volver a usar el certificado una vez exportado.

6. Introduzca la contraseña y haga clic en **Siguiete**. Esta es la contraseña que necesitará después de copiar el certificado al sistema operativo de destino.



Asistente para exportación de certificados

Contraseña
Para mantener la seguridad, debe proteger la clave privada por medio de una contraseña.

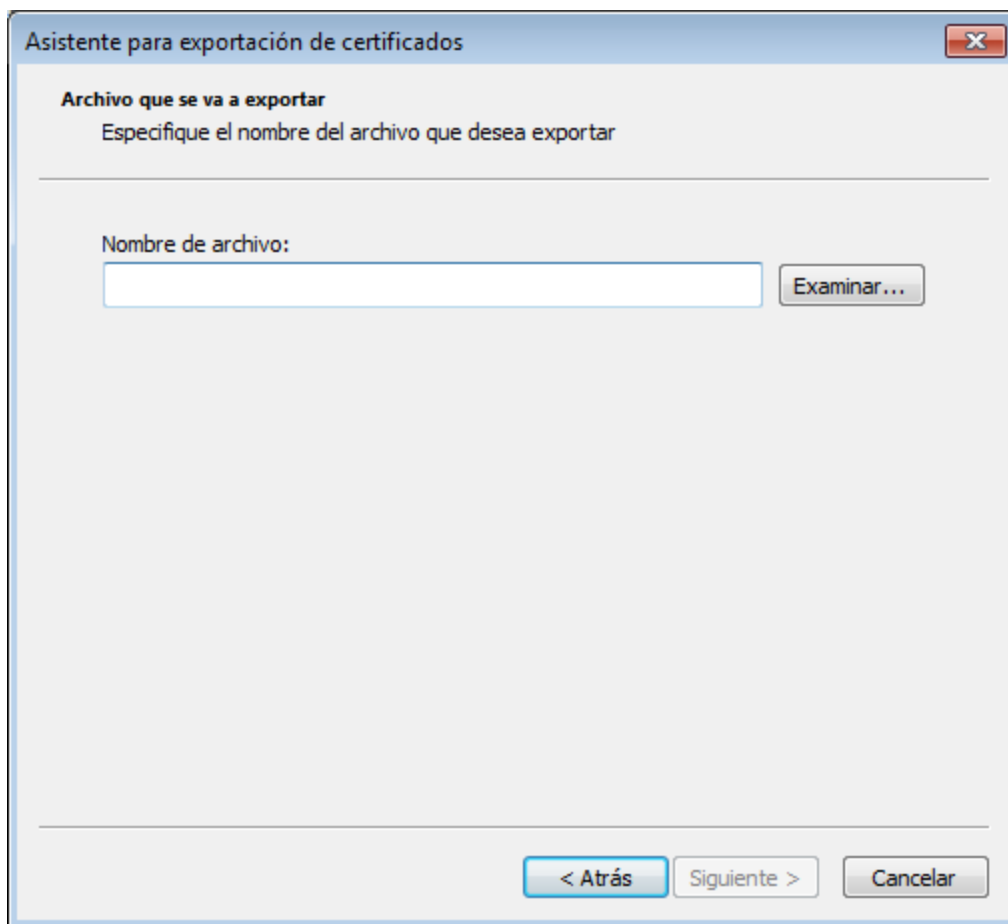
Escriba y confirme una contraseña.

Contraseña:
••••

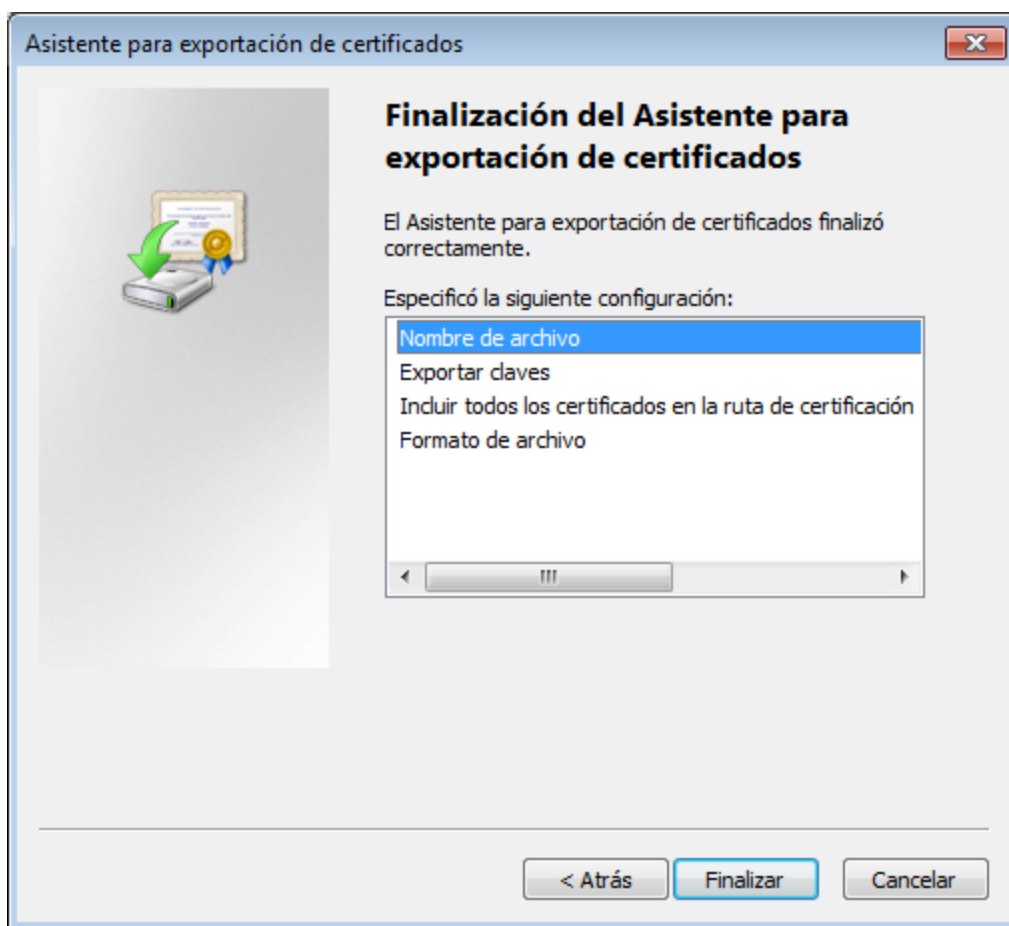
Escribir y confirmar contraseña (obligatorio):
••••|

< Atrás **Siguiete >** Cancelar

7. Busque el archivo que desea exportar y haga clic en **Siguiete**.



8. Para terminar haga clic en **Finalizar**.



6.6 Certificados de cliente en Linux

Si sus asignaciones incluyen autenticación de servicios web por HTTPS mediante certificados de cliente, siga estos pasos para implementar dichas asignaciones en el equipo Linux que ejecuta MapForce Server:

1. Abra la asignación que llama al servicio web en MapForce.
2. Haga doble clic en el título del componente de servicio web para abrir el cuadro de diálogo "Configuración del componente".
3. Haga clic en el botón **Editar** situado junto a *Configuración de seguridad HTTP*.
4. En el cuadro de diálogo "Configuración de seguridad HTTP" marque la casilla *Certificado de cliente* y después seleccione el certificado necesario en el almacén **Usuario actual/Personal** de Windows.
5. Guarde y compile la asignación en un archivo de ejecución o impleméntela en FlowForce Server.
6. Transfiera el certificado de cliente que necesita la llamada al servicio web al sistema operativo de destino. Compruebe que el certificado tiene una clave privada y que la propiedad *Uso mejorado de claves* del certificado incluye *Autenticación de cliente*.

Para transferir el certificado de cliente a Linux:

1. Exporte el certificado de cliente con clave privada desde Windows en formato de archivo *Personal Information Exchange - PKCS #12 (.pfx)* (véase [Exportar certificados de Windows](#)¹³²).
2. Copie este archivo al equipo Linux.
3. Convierta el archivo .pfx al formato .pem con este comando:

```
openssl pkcs12 -in cert.pfx -out "John Doe.pem" -nodes
```

Este comando analiza el archivo .pfx y genera un archivo .pem sin cifrar la clave privada. Los certificados con clave privada cifrada piden una contraseña y no son compatibles con ejecuciones en el servidor.

Ejecutar la asignación

Para instruir a MapForce Server que utilice el archivo .pem como certificado cliente, configure el parámetro `--certificatespath` cuando ejecute la asignación. El parámetro `--certificatespath` define la ruta de acceso del directorio donde se almacenan los certificados que necesita la asignación actual. Por ejemplo, si la ruta de acceso del certificado es `/home/John/John Doe.pem`, entonces `--certificatespath` debe configurarse con el valor `/home/John`.

Si no se indica el parámetro `--certificatespath`, el comportamiento predeterminado de MapForce Server es buscar certificados en el directorio `$HOME/.config/altova/certificates` del usuario actual.

Para que la asignación se ejecute correctamente el archivo del certificado debe tener la extensión .pem y el nombre de archivo debe coincidir con el nombre común (CN) del certificado, incluidos los espacios (p. ej. **John Doe.pem**). Si el nombre común contiene una barra diagonal (/), sustitúyala por un carácter de subrayado (_).

Si tiene pensado ejecutar la asignación como trabajo de FlowForce Server, copie el archivo del certificado en el directorio `$HOME/.config/altova/certificates`. Cuando ejecute el trabajo, FlowForce Server utilizará este directorio para buscar los archivos de certificados que necesite la asignación.

Por motivos de seguridad, compruebe que los archivos de certificado no pueden ser leídos por otros usuarios porque contienen información confidencial.

6.7 Certificados de cliente en OS X / macOS

Si sus asignaciones de datos incluyen autenticación de servicios web a través de certificados de cliente HTTPS, siga estos pasos para implementar estas asignaciones en un macOS que ejecute MapForce Server:

1. Abra la asignación que llama al servicio web en MapForce.
2. Haga doble clic en el título del componente de servicio web para abrir el cuadro de diálogo "Configuración del componente".
3. Haga clic en el botón **Editar** situado junto a *Configuración de seguridad HTTP*.
4. En el cuadro de diálogo "Configuración de seguridad HTTP" marque la casilla *Certificado de cliente* y después seleccione el certificado necesario.
5. Si el nombre del certificado no coincide totalmente con el nombre de host del servidor, marque la casilla *Permitir diferencias entre el nombre de host del certificado y el de la solicitud*.
6. Guarde e implemente la asignación en el sistema operativo de destino.
7. Transfiera el certificado de cliente que necesita la llamada al servicio web al sistema operativo de destino. Compruebe que el certificado tiene una clave privada y que la propiedad *Uso mejorado de claves* del certificado incluye *Autenticación de cliente*.

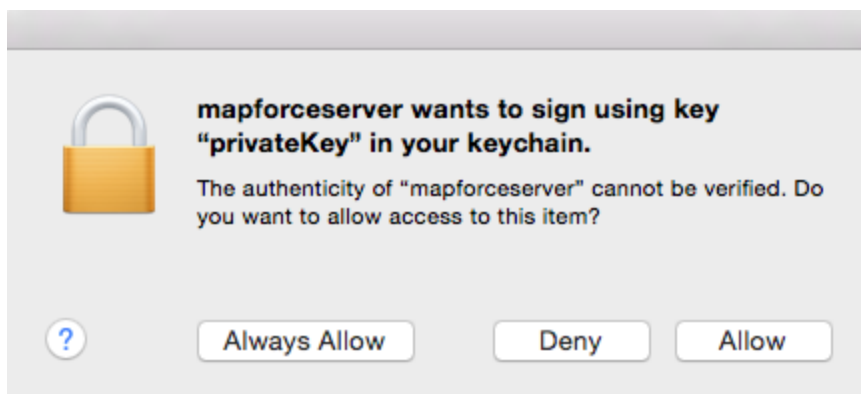
Para transferir el certificado de cliente a macOS:

1. Exporte el certificado de cliente con clave privada desde Windows en formato de archivo *Personal Information Exchange - PKCS #12 (.pfx)* (véase [Exportar certificados de Windows](#)¹³²) y copie este archivo a macOS.
2. Si todavía no lo ha hecho, compruebe que el sistema operativo confía en el certificado de servidor (véase [Confiar en certificados servidor en macOS](#)¹²⁹).
3. Ejecute Keychain Access desde **Finder > Applications > Utilities**.
4. En el menú **File** haga clic en **Import Items**.
5. Busque el certificado de cliente que exportó de Windows en el paso nº1 y seleccione las llaves de destino.
6. Haga clic en **Open** para abrir e introducir la contraseña con la que se cifró el certificado.

Ejecutar la asignación

Ahora ya puede ejecutar la asignación con el comando `run` de MapForce Server, pero tenga en cuenta estos aspectos:

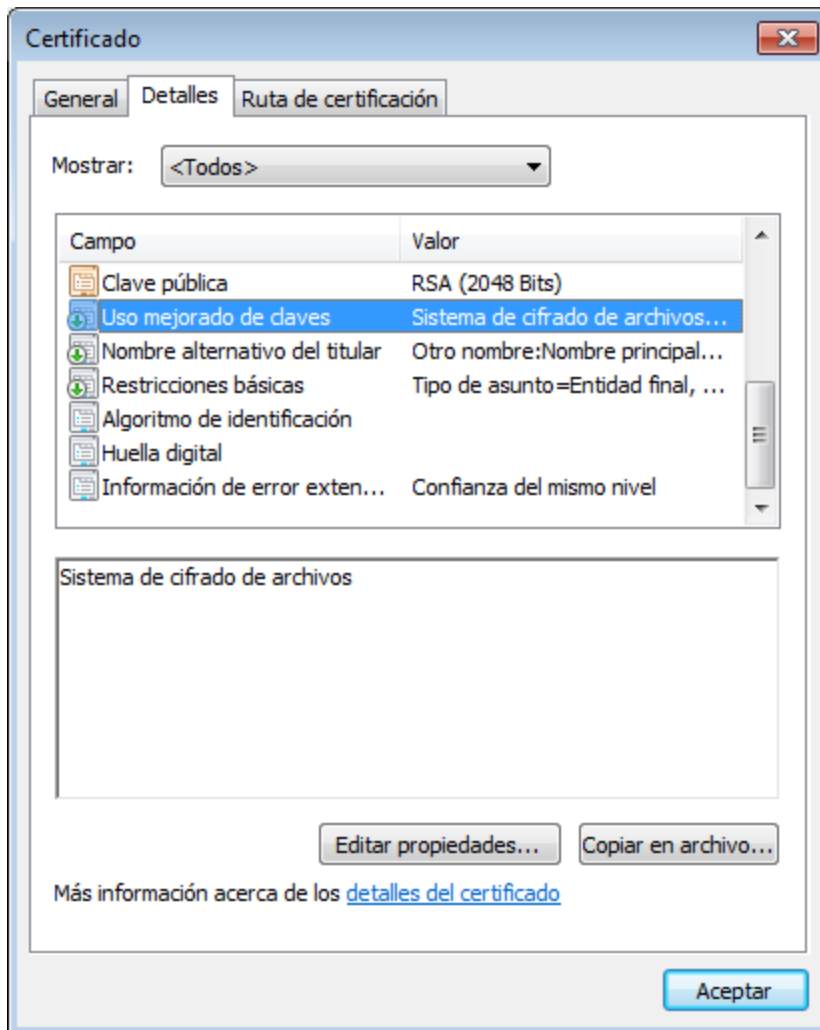
- Si ejecuta la asignación de forma remota por SSH, primero debe desbloquear las llaves con el comando `security unlock-keychain command`.
- Si ejecuta la asignación desde la interfaz del usuario gráfica de macOS, haga clic en **Allow** para permitir a MapForce Server acceder a las llaves.



6.8 Certificados de cliente en Windows

Si ejecuta en Windows una asignación que requiere certificados de cliente, los requisitos para ejecutar correctamente la asignación son:

- El certificado de cliente debe existir en el almacén de certificados **Usuario actual/Personal**. Para que el certificado exista en este almacén, debe ser importado con el asistente de importación de certificados (más información en [https://technet.microsoft.com/es-es/library/cc754489\(v=ws.11\).aspx](https://technet.microsoft.com/es-es/library/cc754489(v=ws.11).aspx)).
- El certificado debe tener una clave privada.
- La propiedad **Uso mejorado de claves** debe incluir *Autenticación de cliente*.



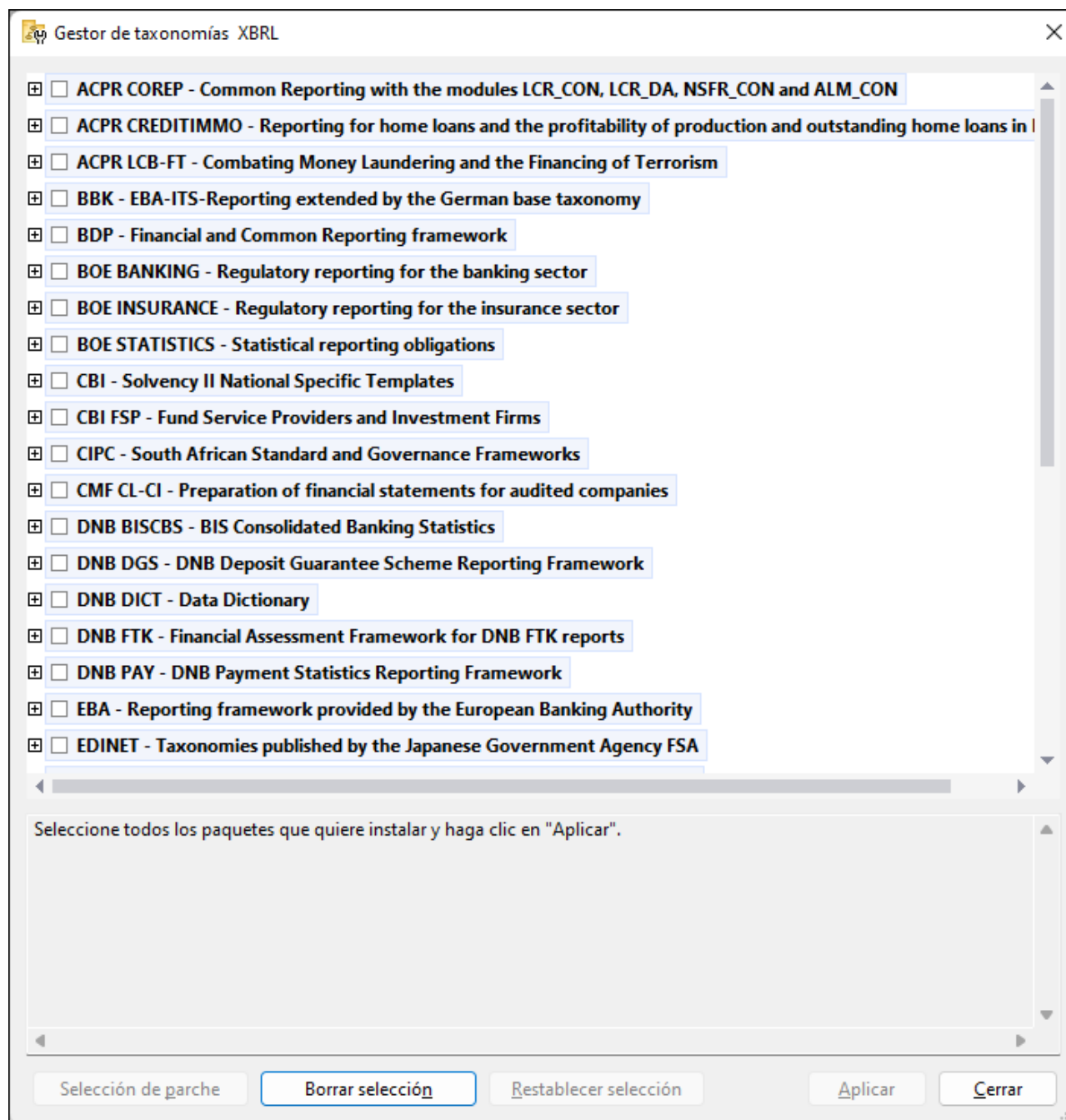
La biblioteca que utiliza la versión actual de Altova MapForce tiene algunas limitaciones. Por ese motivo Windows seleccionará automáticamente el certificado necesario del almacén de certificados cuando ejecute la asignación. La asignación se ejecutará correctamente si, tras filtrar el almacén de certificados **Usuario actual/Personal**, el servidor encuentra un certificado adecuado. Recuerde que de la autenticación HTTPS (y de la operación de selección del certificado) se ocupa Windows y no Altova MapForce ni MapForce Server. En algunos casos, si existen varios certificados en el almacén de certificados **Usuario actual/Personal**, puede

que el sistema operativo seleccione automáticamente un certificado inadecuado. Esto puede dar lugar a un error en la ejecución de la asignación. Esto puede evitarse limitando el número de certificados disponibles en el almacén de certificados **Usuario actual/Personal**.

7 Gestor de taxonomías

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 **MapForce Server**.

- En Windows, el Gestor de taxonomías tiene una interfaz gráfica del usuario (*imagen siguiente*) a la que también puede acceder desde la línea de comandos. (Las aplicaciones de escritorio de Altova sólo están disponibles para Windows; *consulte la lista siguiente*.)
- En Linux y macOS el Gestor de taxonomías sólo está disponible en la línea de comandos. (Las aplicaciones de escritorio de Altova están disponibles para Windows, Linux y macOS; *consulte la lista siguiente*.)



Aplicaciones de Altova compatibles con XBRL

Aplicaciones de escritorio (sólo para Windows)	Aplicaciones de servidor (Windows, Linux, macOS)
Complementos XBRL de Altova para Excel (EBA, ESEF, Solvency II, WIP)	MapForce Server (ediciones Standard y Advanced)

MapForce Enterprise Edition	RaptorXML+XBRL Server
StyleVision Enterprise Edition	StyleVision Server
XMLSpy Enterprise Edition	

Instalación y desinstalación del Gestor de taxonomías

El Gestor de taxonomías se instala automáticamente al instalar cualquiera de las aplicaciones de Altova compatibles con XBRL o el Altova Mission Kit Enterprise Edition (véase *la tabla de más arriba*). También se elimina automáticamente si desinstala todas las aplicaciones de Altova compatibles con XBRL o el Altova Mission Kit Enterprise Edition del equipo.

Características de Gestor de taxonomías

El Gestor de taxonomías 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 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 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.
- Todas las taxonomías principales están incluidas en el Gestor de taxonomías y se actualizan de forma periódica a la versión más reciente. De esta forma puede administrar todas las taxonomías desde un punto común y tenerlas siempre listas para las aplicaciones de Altova que las usan.
- Los cambios que se realizan en el Gestor de taxonomías afectan a todos los productos de Altova que estén instalados en ese equipo.

Taxonomías XBRL personales

Si necesita trabajar con taxonomías XBRL que no estén incluidas en el Gestor de taxonomías, puede agregarlas a un paquete personal al que el MapForce Server podrá hacer referencia más adelante. Puede habilitar las taxonomías XBRL así:

- *Desde las aplicaciones de escritorio de Altova:* 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. Para más información consulte la descripción de este comando en la documentación del producto correspondiente.
- *Desde las aplicaciones servidor de Altova:* desde una línea de comandos compatible con XBRL proporcione las opciones `-taxonomy-package 0 ---taxonomy-package-config-file`. Por ejemplo, estas opciones son compatibles con los comandos de validación XBRL como `valxbr1` o `valxbr1taxonomy` en RaptorXML+XBRL Server, o el comando `run` en MapForce Server.

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 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](#). El sitio web prepara un archivo de tipo `.altova_taxonomies` que puede descargar y que contiene la información sobre las taxonomías seleccionadas. Al hacer doble clic en este archivo o pasarlo al **Gestor de taxonomías** desde la línea de comandos como argumento del comando `install`¹⁵⁸, el Gestor de taxonomías instala las taxonomías que contiene.

Memoria caché local: seguimiento de taxonomías

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 está en:

<i>Windows</i>	C:\ProgramData\Altova\pkgs\.cache
<i>Linux</i>	/var/opt/Altova/pkgs/.cache
<i>macOS</i>	/var/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, la memoria caché se actualiza:

- Al ejecutar el Gestor de taxonomías.
- Al ejecutar MapForce Server por primera vez en un mismo día natural.
- Si MapForce 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 `update`¹⁶¹ desde la línea de comandos.

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

No modifique la memoria caché manualmente.

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 a su estado original, ejecute el comando `reset`¹⁵⁹ desde la línea de comandos y después ejecute el comando `initialize`¹⁵⁷. (También puede ejecutar el comando `reset` con la opción `--i`.)

Proxy de HTTP

Puede usar un proxy de HTTP para las conexiones del Gestor de taxonomías. Se utilizará la configuración proxy del sistema.

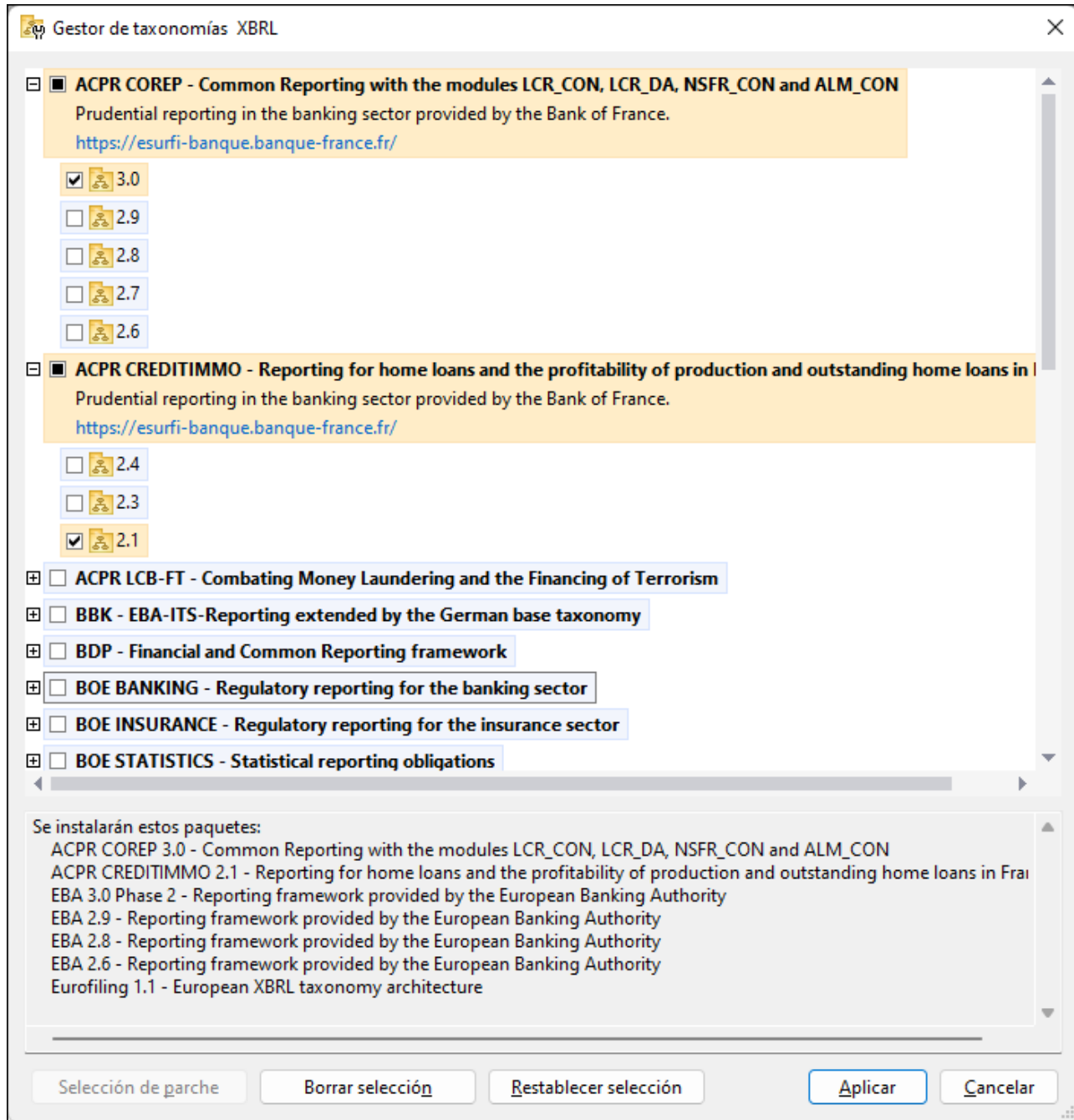
7.1 Ejecutar el gestor de taxonomías

Interfaz gráfica del usuario

Hay varias formas de acceder a la IGU del Gestor de taxonomías:

- *Durante la instalación de MapForce Server:* al final del proceso de instalación, seleccione la casilla Invocar al Gestor de taxonomías para acceder directamente al gestor de taxonomías XBRL. Con él puede instalar taxonomías durante el proceso de instalación de su aplicación de Altova.
- Mediante el archivo `.altova_taxonomies` que descargó del [Centro de descargas de taxonomías XBRL de Altova](#): haga doble clic en el archivo para ejecutar Gestor de taxonomías, que instalará las taxonomías que haya seleccionado.

Cuando se abra la IGU del Gestor de taxonomías (*imagen siguiente*) podrá ver en ella las taxonomías que ya se han instalado. Si quiere instalar más sólo tiene que seleccionarlas, y al contrario si quiere desinstalar alguna. Una vez haya terminado, puede aplicar los cambios. Las taxonomías que se vayan a instalar o desinstalar aparecerán resaltadas y un mensaje le avisará de los cambios que está a punto de hacer en la ventana *Mensajes*, en la parte inferior de la ventana del **Gestor de taxonomías** (*imagen siguiente*).



Interfaz de la línea de comandos

Para ejecutar el Gestor de taxonomías desde una interfaz de la línea de comandos debe usar su archivo ejecutable, `taxonomymanager.exe`.

Puede encontrar este archivo:

- *En Windows:* C:\ProgramData\Altova\SharedBetweenVersions
- *En Linux o macOS (sólo para aplicaciones de servidor):* %INSTALLDIR%/bin, donde %INSTALLDIR% es el directorio de instalación del programa.

Puede usar cualquiera de los comandos de la referencia de la línea de comandos, a continuación.

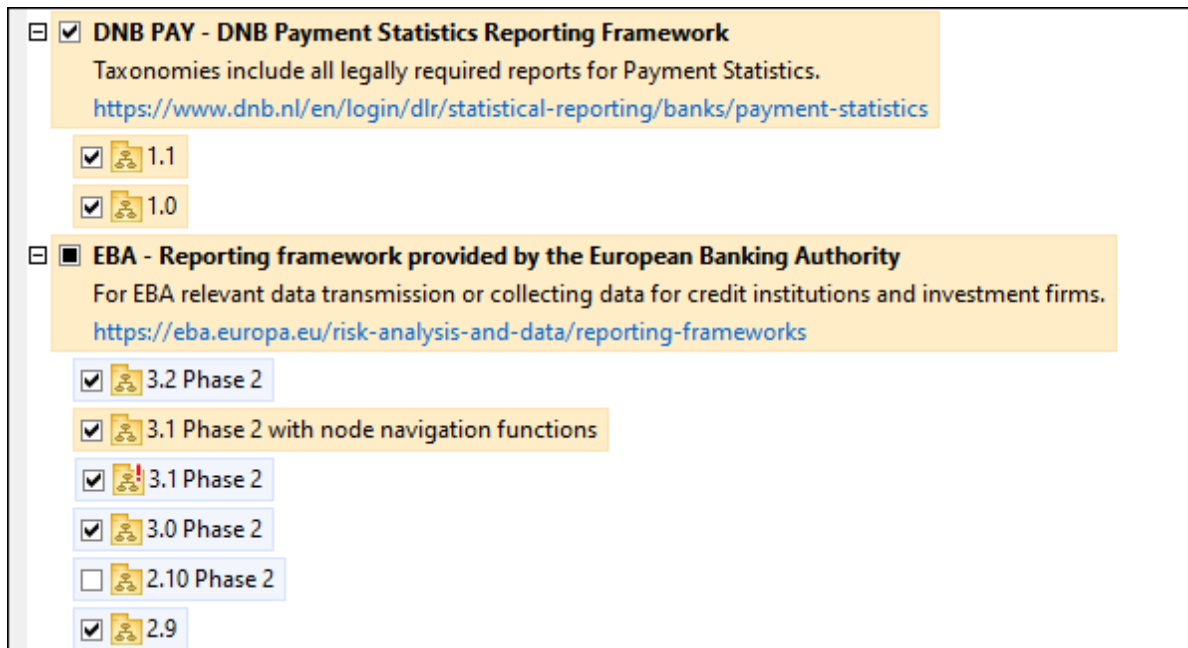
Para mostrar al ayuda de los comandos ejecute:


- *En Windows:* `taxonomymanager.exe --help`
- *En Linux o macOS (sólo para aplicaciones de servidor):* `sudo ./taxonomymanager --help`

7.2 Categorías de estado

El Gestor de taxonomías diferencia las taxonomías que administra entre:

- *Taxonomías instaladas:* estas aparecen en la IGU con sus casillas marcadas (*en la imagen siguiente las versiones marcadas de la taxonomía EBA son las que están instaladas*). Si se seleccionan todas las versiones de una taxonomía, en la casilla de la taxonomía aparece una marca de verificación. Si hay al menos una taxonomía sin seleccionar, en la casilla de la taxonomía aparece un cuadrado negro. Para **desinstalar** una taxonomía debe desmarcar la casilla correspondiente.
- *Taxonomías disponibles no instaladas:* Estas aparecen en la IGU con las casillas correspondientes sin seleccionar. Para **instalar** taxonomías, marque la casilla correspondiente.



- *Taxonomías que pueden actualizarse:* son las que han sido revisadas por sus emisores. Aparecen indicadas en la IGU con el icono  (*imagen anterior*). Puede aplicar **parches** a la taxonomía seleccionada con la revisión que esté disponible.

Notas importantes

- En la imagen anterior se han marcado tanto las taxonomías DNB como las EBA. Las que tienen un fondo azul ya están instaladas. Las que tienen el fondo amarillo no están instaladas pero se han seleccionado para instalarlas. Observe que (i) la taxonomía EBA 2.10 Phase 2 no está instalada ni se ha seleccionado para instalarla, (ii) la taxonomía EBA 3.1 Phase 2 se ha instalado pero su emisor ha generado un parche que no se ha instalado aún.
- Al ejecutar el Gestor de taxonomías desde la línea de comandos puede usar el comando `list`¹⁵⁸ con distintas opciones para ver distintas categorías de taxonomías:

<code>taxonomymanager.exe list</code>	Muestra todas las taxonomías instaladas y disponibles; también indica qué taxonomías se pueden actualizar
---------------------------------------	---

<code>taxonomymanager.exe list -i</code>	Muestra sólo las taxonomías instaladas; también indica qué taxonomías se pueden actualizar
<code>taxonomymanager.exe list -u</code>	Muestra qué taxonomías se pueden actualizar




Nota: En Linux y macOS use `sudo ./taxonomymanager list`

7.3 Aplicar parches o instalar una taxonomía

Aplicar un parche a una taxonomía instalada

A veces los emisores de las taxonomías XBRL generan parches. Cuando el Gestor de taxonomías detecta que hay parches disponibles, estos aparecen en las listas del Gestor de taxonomías, desde donde puede instalarlos.

En la IGU

Los parches se indican con el icono . (Consulte también el apartado anterior sobre las [categorías de estado](#) ¹⁵¹.) Si hay parches disponibles se habilita el botón **Seleccionar parches**. Haga clic en él para seleccionar y preparar los parches. En la IGU, el icono de las taxonomías correspondientes cambia de  a  y el cuadro de diálogo le informa de qué parches se van a aplicar. Las listas del panel principal y las del panel Mensajes están ordenadas alfabéticamente, por lo que puede ver y revisar las taxonomías antes de aplicar los parches. Una vez esté listo para instalar los parches seleccionados, haga clic en **Aplicar**.

En la línea de comandos

Para aplicar un parche desde la línea de comandos:

1. Ejecute el comando `list -u` ¹⁵⁸. Aparece una lista con las taxonomías para las que hay parches disponibles.
2. Ejecute el comando `upgrade` ¹⁶¹ para instalar todos los parches.

Instalar una taxonomía disponible

Para instalar taxonomías puede usar la IGU del Gestor de taxonomías o enviar las instrucciones al Gestor de taxonomías desde la línea de comandos.

Nota: Si la taxonomía actual tiene dependencias en otras taxonomías, también se instalan (o desinstalan, según el caso) las taxonomías dependientes.

En la IGU

Para instalar taxonomías con la IGU del Gestor de taxonomías, seleccione las taxonomías que quiere instalar y haga clic en **Aplicar**.

También puede seleccionar las taxonomías que quiere instalar en el [sitio web de Altova](#) y generar desde allí un archivo `.altova_taxonomies`. Al hacer doble clic en este archivo se abre el Gestor de taxonomías con las taxonomías que indicó preseleccionadas. Sólo tiene que hacer clic en **Aplicar**.

En la línea de comandos

Para instalar taxonomías desde la línea de comandos ejecute el comando `install` ¹⁵⁸:

```
taxonomymanager.exe install [opciones] FILTER+
```

donde **FILTER** es la taxonomía (o las taxonomías) que quiere instalar o un archivo `.altova_taxonomies`. Para hacer referencia a una taxonomía se usa un identificador con el formato `<nombre>-<versión>` que aparece junto a cada taxonomía que muestra el comando `list` ¹⁵⁸. Puede introducir tantas taxonomías como quiera. Para más detalles consulte la descripción del comando `install` ¹⁵⁸.

Nota: En Linux o macOS, use el comando `sudo ./taxonomymanager`.

Instalar una taxonomía requerida

Si ejecuta un comando en MapForce Server y MapForce Server descubre que una de las taxonomías que necesita para ejecutar el comando falta o está incompleta, el Gestor de taxonomías incluirá información sobre ese componente de taxonomía que falta. Entonces puede aplicar el parche indicado y/o instalar la taxonomía que falta.

Siempre puede ver todas las taxonomías instaladas previamente ejecutando el Gestor de taxonomías desde **Herramientas | Gestor de taxonomías**.

7.4 Desinstalar o restaurar taxonomías

Desinstalar una taxonomía

Para desinstalar taxonomías puede usar la IGU del Gestor de taxonomías o enviar las instrucciones al Gestor de taxonomías desde la línea de comandos.

Nota: Si la taxonomía actual tiene dependencias en otras taxonomías, también se instalan (o desinstalan, según el caso) las taxonomías dependientes.

En la IGU

Para desinstalar taxonomías con la IGU del Gestor de taxonomías, seleccione las taxonomías que quiere desinstalar y haga clic en **Aplicar**. Las taxonomías seleccionadas y sus dependencias se desinstalarán.

Para desinstalar todas las taxonomías haga clic en **Deseleccionar todas** y haga clic en **Aplicar**.

En la línea de comandos

Para desinstalar taxonomías desde la línea de comandos ejecute el comando `uninstall`:

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

donde **FILTER** es la taxonomía (o las taxonomías) que quiere desinstalar o un archivo `.altova_taxonomies`. Para hacer referencia a una taxonomía se usa un identificador con el formato `<nombre>-<versión>` que aparece junto a cada taxonomía que muestra el comando [list](#)¹⁵⁸. Puede introducir tantas taxonomías como quiera. Para más detalles consulte la descripción del comando [uninstall](#)¹⁶⁰.

Nota: En Linux o macOS, use el comando `sudo ./taxonomymanager`.

Restaurar el Gestor de taxonomías

Puede restaurar el Gestor de taxonomías.

- En la IGU, haga clic en **Restaurar selección**. Esto reinicia la IGU para mostrar qué taxonomías están actualmente instaladas. Se cancelarán todas las opciones que el usuario haya seleccionado o deseleccionado en la sesión actual.
- En la línea de comandos, use el comando [reset](#)¹⁵⁹. Esto eliminará todas las taxonomías instaladas y el directorio de caché.

Una vez haya ejecutado este comando, asegúrese de que ejecuta también el comando [initialize](#)¹⁵⁷ para recrear el directorio caché. También puede ejecutar el comando [reset](#)¹⁵⁹ con la opción `-i`.

Recuerde que [reset -i](#)¹⁵⁹ restaura la instalación original del producto, por lo que es recomendable ejecutar el comando [update](#)¹⁶¹ después de restaurar el gestor. Puede ejecutar el comando [reset](#)¹⁵⁹ con las opciones `-i 0 -u`.

7.5 Interfaz de la línea de comandos (ILC)

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

<i>Windows</i>	C:\ProgramData\Altova\SharedBetweenVersions\TaxonomyManager.exe
<i>Linux</i>	/opt/Altova/MapForceServer2024/bin/taxonomymanager
<i>macOS</i>	/usr/local/Altova/MapForceServer2024/bin/taxonomymanager

Nota: En los sistemas Linux y macOS una vez haya cambiado el directorio al que contiene el ejecutable, puede llamar al ejecutable con `sudo ./taxonomymanager`. 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.

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.

7.5.1 help

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

Sintaxis

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

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

Tenga en cuenta que:

- Puede invocar la ayuda tecleando un comando seguido por `--h` or `--help`, por ejemplo: `<exec> list -h`
- Puede invocar la ayuda general (no para el comando) tecleando `--h` o `--help` directamente después del ejecutable, por ejemplo: `<exec> -h list`

Ejemplo

Este comando muestra la ayuda del comando `list`:

```
taxonomymanager help list
```

7.5.2 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

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

- El argumento **Taxonomy** es el nombre de una taxonomía o parte del nombre de una taxonomía. (Para ver el ID de un paquete de taxonomías y la información relativa a su estado de instalación use el comando [list](#) ¹⁵⁸.)
- Use `<exec> info -h` para ver la ayuda sobre este comando en la línea de comandos.

Ejemplo

Este comando muestra información detallada sobre las taxonomías `eba-2.10` y `us-gaap-2020`:

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

7.5.3 initialize

Este comando inicializa el entorno del Gestor de taxonomías y crea un directorio caché donde se guardan todas las taxonomías localmente. El Gestor de taxonomías 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 sólo es necesario ejecutarlo después de haber ejecutado el comando [reset](#) ¹⁵⁹.

Sintaxis

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

Opciones

Estas son las opciones del comando `initialize`:

<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 <code>false</code> .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es <code>false</code> .

Ejemplo

Este comando inicializa el Gestor de taxonomías:

```
taxonomymanager initialize
```

7.5.4 install

Este comando instala una o más taxonomías.

Sintaxis

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

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

El argumento de `Taxonomy` puede ser:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: `eba-2.10`. Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list`¹⁵⁸. También puede usar el nombre de la taxonomía abreviado, si este es único, por ejemplo `eba`. Si usa una abreviación del nombre se desinstalan todas las taxonomías que contengan esa abreviación.
2. La ruta de acceso a un archivo `.altova_taxonomies` descargado desde el sitio web de Altova. Para más información sobre estos archivos consulte la [Introducción al Gestor de taxonomías: Funcionamiento](#)¹⁴⁴.

Opciones

Estas son las opciones del comando `install`:

<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 <code>false</code> .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es <code>false</code> .

Ejemplo

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

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

7.5.5 list

Use este comando para ver las taxonomías del Gestor de taxonomías; 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 | ls [options] Taxonomy?
```

Si no se indica ningún argumento `Taxonomy` la lista incluye todas las taxonomías. De lo contrario la lista incluye las taxonomías indicadas en las opciones (véase el ejemplo de más abajo). Recuerde que puede usar el argumento `Taxonomy` tantas veces como quiera.

Opciones

Estas son las opciones del comando `list`:

<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 <code>false</code> .
<code>--upgradeable, --u</code>	Muestra solamente las taxonomías para las que hay disponible una versión más reciente (parches). El valor predeterminado es <code>false</code> .

Ejemplos

- Para ver todas las taxonomías disponibles ejecute: `taxonomymanager list`
- Para ver solamente las taxonomías instaladas ejecute: `taxonomymanager list -i`
- Para ver todas las taxonomías cuyos nombres contienen "eba" o "us-gaap" ejecute: `taxonomymanager list eba us-gaap`

7.5.6 reset

Este comando elimina todas las taxonomías instaladas, así como el directorio de caché. Este comando elimina todas las taxonomías instaladas y su información. Una vez haya ejecutado este comando, asegúrese de que ejecuta el comando [initialize](#)¹⁵⁷ 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](#)¹⁶¹ después de una restauración. También puede ejecutar el comando `reset` con las opciones `-i` y `-u`.

Sintaxis

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

Opciones

Estas son las opciones del comando `reset`:

<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 después de una restauración. El valor predeterminado es <code>false</code> .
<code>--silent, --s</code>	Muestra solamente los mensajes de error. El valor predeterminado es <code>false</code> .
<code>--update, --u</code>	Inicializa y actualiza el entorno del Gestor de taxonomías XBRL después de una restauración. El valor predeterminado es <code>false</code> .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es <code>false</code> .

Ejemplos

- Para restaurar el Gestor de taxonomías, ejecute: `taxonomymanager reset`
- Para restaurar el Gestor de taxonomías e inicializarlo, ejecute: `taxonomymanager reset -i`
- Para restaurar el Gestor de taxonomías, inicializarlo y actualizar la lista de taxonomías, ejecute: `taxonomymanager reset -i -u`

7.5.7 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 [opciones] Taxonomy+
```

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

El argumento de `Taxonomy` puede ser:

1. Un identificador de taxonomía en el formato `<name>-<version>`, por ejemplo: `eba-2.10`). Para ver todos los identificadores de taxonomías y sus versiones ejecute el comando `list -i`¹⁵⁸. También puede usar el nombre de la taxonomía abreviado, si este es único, por ejemplo `eba`. Si usa una abreviación del nombre se desinstalan todas las taxonomías que contengan esa abreviación.
2. La ruta de acceso a un archivo `.altova_taxonomies` descargado desde el sitio web de Altova. Para más información sobre estos archivos consulte la [Introducción al Gestor de taxonomías: Funcionamiento](#)¹⁴⁴ ..

Opciones

Estas son las opciones del comando `uninstall`:

<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 <code>false</code> .

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

Ejemplo

Este comando desinstala las taxonomías `eba-2.10` y `us-gaap-2020.0` y sus dependencias:

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

Este comando desinstala la taxonomía `eba-2.10` pero no las taxonomías a las que hace referencia:

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

7.5.8 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](#)¹⁵⁹ e [initialize](#)¹⁵⁷.

Sintaxis

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

Opciones

Estas son las opciones del comando `update`:

<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 <code>false</code> .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es <code>false</code> .

Ejemplo

Este comando actualiza la lista de taxonomías en la memoria caché local:

```
taxonomymanager update
```

7.5.9 upgrade

Este comando actualiza todas las taxonomías aptas para 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 slo si hay taxonomías por actualizar. Puede identificar cuáles lo son con el comando [list -u](#)¹⁵⁸.

Nota: El comando `upgrade` eliminaría una taxonomía obsoleta si no hay ninguna versión disponible.

Sintaxis

<exec> upgrade [opciones]

Opciones

Estas son las opciones del comando `upgrade`:

<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 <code>false</code> .
<code>--verbose, --v</code>	Muestra información suplementaria durante la ejecución. El valor predeterminado es <code>false</code> .

8 Archivos de catálogo

Si usa el mecanismo de catalogación de Altova encontrará los archivos relevantes en la carpeta `etc` de la carpeta de la aplicación MapForce Server. Para leer información más detallada consulte [la sección Catálogos del manual de Altova MapForce](#).

Puede crear el archivo `CustomCatalog.xml` a partir del archivo de plantilla `CustomCatalog_template.xml`. Asegúrese de que cambia el nombre del archivo de plantilla a `CustomCatalog.xml`, ya que es este el que se usa en el mecanismo de catalogación (y no la plantilla).

Tenga en cuenta que:

- Durante una instalación nueva de la misma versión principal (misma u otra subversión), el archivo de plantilla se sustituye por uno nuevo, pero `CustomCatalog.xml` no cambia.
- Sin embargo, si instala una versión principal nueva sobre una versión principal anterior, la carpeta de la instalación anterior se borra junto con su archivo `CustomCatalog.xml`. Por tanto, si quiere seguir usando el `CustomCatalog.xml` de la versión anterior debe guardarlo en una ubicación segura. Después de instalar la versión principal nueva puede copiar el archivo `CustomCatalog.xml` guardado en la carpeta `etc` de la versión principal nueva y editarlo allí como necesite.

Índice

A

Actualizar MapForce Server en Windows, 27

ADO, 32

ADO.NET, 32

Altova ServiceController, 12

API de MapForce Server,

para Java, 80

Asignar licencias a MapForce Server en Linux, 19

Asignar licencias a MapForce Server en macOS, 24

Asignar licencias a MapForce Server en Windows, 12

Asignar una licencia a MapForce Server en Linux, 20

Asignar una licencia a MapForce Server en macOS, 25

Asignar una licencia a MapForce Server en Windows, 14

Asignar una licencia para MapForce Server,

asignar licencias en Linux, 20

asignar licencias en macOS, 25

asignar licencias en Windows, 14

assignlicense,

as CLI command, 50

C

Catálogos XML,

configurar, 163

funcionamiento, 163

Certificados digitales,

de confianza en Linux, 128

de confianza en Mac, 129

de confianza en Windows, 130

en asignaciones de MapForce, 125

exportar de Windows, 132

gestión en Windows, 131

transferir a Linux, 138

transferir a OS X / macOS, 140

CLI commands,

assignlicense, 50

exportresourcestrings, 51

help, 53

licenseserver, 54

run, 55

setdeflang, 59

verifylicense, 60, 61

Configuración,

en Linux, 16

D

Desinstalación, 8

Desinstalar, 8

E

exportresourcestrings,

as CLI command, 51

F

FlowForce Server, 29, 32

G

Gestor de taxonomías,

actualizar una taxonomía, 153

aplicar un parche a una taxonomía, 153

desinstalar una taxonomía, 155

estado de las taxonomías en el, 151

funcionamiento, 148

help (comando ILC), 156

info (comando ILC), 157

initialize (comando ILC), 157

instalar una taxonomía, 153

install (comando ILC), 158

introducción, 144

introducción a la ILC, 156

list (comando ILC), 158

lista de taxonomías por estado, 151

reset (comando ILC), 159

restaurar, 155

uninstall (comando ILC), 160

Gestor de taxonomías,

- update (comando ILC), 161
- upgrade (comando ILC), 161

Global Resources,

- definition, 38
- using in mappings, 38

H**help,**

- as CLI command, 53

HTTPS,

- llamar a servicios web por, 125

I**Iniciar LicenseServer en Linux, 19****Iniciar LicenseServer en macOS, 24****Iniciar LicenseServer en Windows, 12****Iniciar MapForce Server en Linux, 19****Iniciar MapForce Server en macOS, 24****Iniciar MapForce Server en Windows, 12****Instalación en Linux, 16****Instalación en macOS, 22****Instalación en Windows, 8****Instalación en Windows Server Core, 9****Instalación y configuración,**

- en macOS, 22
- en Windows, 8

Instalar LicenseServer en Linux, 18**Instalar LicenseServer en macOS, 23****Instalar LicenseServer en Windows, 11****Interfaz de la línea de comandos,**

- cómo usar, 47

J**Java, 32**

- agregar MapForce Server a CLASSPATH, 80

Join optimization,

- how it works, 40

L**licenseserver,**

- as CLI command, 54

Linux,

- asignar una licencia a MapForce Server en, 19
- certificados de servidor de confianza en, 128
- ejecutar asignaciones con llamadas a servicios web por HTTPS, 138
- Instalación en, 16
- transferir certificados clientes a, 138

M**Mac,**

- certificados de servidor de confianza en, 129

macOS,

- asignar una licencia a MapForce Server en, 24
- Instalación en, 22

MapForce Server,

- migrar a un equipo nuevo, 28

Mappings,

- compiling to execution files, 29
- deploying to FlowForce Server, 29
- preparing for execution, 32
- running at the command line, 55

Migrar MapForce Server a un equipo nuevo, 28**O****ODBC, 32****OS X / macOS,**

- ejecutar asignaciones con llamadas a servicios web por HTTPS, 140
- transferir certificados clientes a, 140

R**Registrar MapForce Server con LicenseServer en Linux, 19**

Registrar MapForce Server con LicenseServer en macOS, 24

Registrar MapForce Server con LicenseServer en Windows, 13

run,

as CLI command, 55

S

Servicios web,

llamar por HTTPS, 125

setdeflang,

as CLI command, 59

V

verifylicense,

as CLI command, 60

version,

as CLI command, 61

Versiones de LicenseServer, 11, 18, 23

W

Windows,

actualizar MapForce Server en, 27

asignar una licencia a MapForce Server en, 12

certificados de servidor de confianza en, 130

ejecutar asignaciones con llamadas a servicios web por HTTPS, 142

Instalación en, 8