

IBM WebSphere Commerce



# Guía de implementación de servicios Web

*Versión 5.5*



IBM WebSphere Commerce



# Guía de implementación de servicios Web

*Versión 5.5*

**Nota**

Antes de utilizar esta información y el producto al que da soporte, asegúrese de leer la información general del apartado "Avisos" en la página 57.

**Primera edición, primera revisión (septiembre de 2003)**

Esta edición se aplica a IBM WebSphere Commerce, Versión 5.5 y a todos los releases y modificaciones posteriores de los productos listados hasta que se indique lo contrario en nuevas ediciones. Asegúrese de utilizar la edición correcta para el nivel del producto.

Esta edición se aplica a IBM WebSphere Commerce Studio, Business Developer Edition, Versión 5.5 y IBM WebSphere Commerce Studio, Professional Developer Edition, Versión 5.5, y a todos los releases y modificaciones posteriores de los productos listados hasta que se indique lo contrario en nuevas ediciones. Asegúrese de utilizar la edición correcta para el nivel del producto.

Puede solicitar las publicaciones a través de su representante de IBM o a la sucursal de IBM que atiende a su localidad.

IBM agradece sus comentarios. Puede enviar sus comentarios utilizando el formulario de comentarios y sugerencias en línea de IBM WebSphere Commerce, disponible en:

<http://www.ibm.com/software/commerce/rcf.html>

Cuando se envía información a IBM, se otorga a IBM un derecho no exclusivo para utilizar o distribuir la información de la forma que considere apropiada, sin incurrir por ello en ninguna obligación para con el remitente.

© Copyright International Business Machines Corporation 2002, 2003. Reservados todos los derechos.

# Contenido

<b>Antes de empezar</b> . . . . .	<b>v</b>
Conocimientos necesarios . . . . .	vi
Convenios utilizados en esta guía . . . . .	vi
Vías de acceso por omisión . . . . .	vi
Resumen de cambios . . . . .	viii
<b>Capítulo 1. Introducción</b> . . . . .	<b>1</b>
Terminología . . . . .	1
Visión general . . . . .	2
Por qué utilizar servicios Web . . . . .	2
Arquitectura de los servicios Web . . . . .	3
Prerrequisitos . . . . .	4
Referencias . . . . .	5
<b>Capítulo 2. Servicios Web para WebSphere Commerce</b> . . . . .	<b>7</b>
Infraestructura de mensajería de WebSphere Commerce . . . . .	7
Descripción de la implementación de servicios Web para WebSphere Commerce . . . . .	7
WebSphere Commerce como proveedor de servicios . . . . .	7
Proceso de la petición . . . . .	8
Respuesta . . . . .	8
Habilitación de WebSphere Commerce como proveedor de servicios . . . . .	9
WebSphere Commerce como peticionario de servicios . . . . .	10
<b>Capítulo 3. Habilitación de WebSphere Commerce como proveedor de servicios</b> . . . . .	<b>11</b>
Infraestructura para habilitar servicios Web . . . . .	11
Arquitectura . . . . .	11
Consideraciones de seguridad . . . . .	13
Servicios Web y control de acceso . . . . .	13
Seguridad del nivel de transporte . . . . .	14
Generación de código de cliente para acceder a servicios Web . . . . .	14
Servicios Web de ejemplo que pueden invocar los clientes Web externos . . . . .	15
Importación del código de ejemplo . . . . .	15
Servicio OrderCreate (Business Edition) . . . . .	16
Habilitación del servicio Web OrderCreate de ejemplo. . . . .	16
Prueba del ejemplo . . . . .	18
Seguridad para el servicio Web OrderCreate . . . . .	18
Creación de usuarios y contratos . . . . .	19
Inserción de datos en las tablas de base de datos . . . . .	20
Nuevas correlaciones de mensajes . . . . .	23
Configuración de correlacionadores de mensajes . . . . .	24
Servicio OrderStatus . . . . .	24
Habilitación del servicio Web OrderStatus de ejemplo. . . . .	24
Control de acceso para el servicio OrderStatus . . . . .	28
Roles de usuario para el servicio Web OrderStatus . . . . .	29
La definición WSDL de OrderStatus . . . . .	29
Nuevas correlaciones de mensajes . . . . .	29
Configuración de correlacionadores de mensajes . . . . .	31
Inhabilitación de los servicios Web OrderCreate y OrderStatus . . . . .	31
<b>Capítulo 4. Habilitación de WebSphere Commerce como peticionario de servicios</b> . . . . .	<b>35</b>
Arquitectura . . . . .	35
Localización de una implementación de servicio . . . . .	36
Invocación de una implementación de servicio . . . . .	37
Servicio Web de ejemplo que WebSphere Commerce puede invocar en un sistema externo . . . . .	38
Servicio OrderFulfillment . . . . .	38
Definiciones WSDL de interfaz . . . . .	38
Mandato de tarea . . . . .	38
Objetos de negocio . . . . .	41
Prueba del ejemplo . . . . .	42
Inhabilitación del servicio Web OrderFulfillment . . . . .	42
<b>Apéndice A. Archivos de definición WSDL de ejemplo</b> . . . . .	<b>45</b>
<b>Apéndice B. Utilización de IBM WebSphere Studio Application Developer</b> . . . . .	<b>47</b>
Importación de una WSDL en WebSphere Studio Application Developer. . . . .	47
Generación del código de cliente y los tipos de datos . . . . .	48
Prueba del servicio Web . . . . .	49
Utilización del Supervisor TCP para rastrear los mensajes SOAP . . . . .	49
Creación de una instancia de servidor de Supervisor TCP . . . . .	50
Configuración de la instancia de servidor de Supervisor TCP . . . . .	50
Rastreo de los mensajes SOAP utilizando el Supervisor TCP/IP . . . . .	51
Personalización del código de cliente . . . . .	54
<b>Avisos</b> . . . . .	<b>57</b>
Marcas registradas . . . . .	58



---

## Antes de empezar

Esta *Guía de implementación de servicios Web* está pensada para aquellos que deseen exponer los procesos de negocio de WebSphere Commerce Versión 5.5, Business Edition y Professional Edition, como servicios Web. También explica la forma en que WebSphere Commerce puede acceder a los servicios Web alojados en sistemas externos. Esta guía sirve de ayuda a los desarrolladores que necesitan saber cómo habilitar e implementar servicios Web para WebSphere Commerce. También pueden consultar esta guía otros usuarios, como los administradores del sistema, que deseen configurar servicios Web y utilizar la funcionalidad que proporciona WebSphere Commerce. Es necesario tener conocimientos previos del modelo de programación de WebSphere Commerce. Además, también son necesarios conocimientos sobre cómo invocar e implementar servicios Web y los protocolos relacionados.

Esta guía proporciona información sobre cómo habilitar WebSphere Commerce para que funcione con servicios Web y clientes de servicios Web. Ofrece una visión general de los servicios Web y su arquitectura. Describe brevemente la infraestructura de mensajería en WebSphere Commerce, el soporte de ejecución de WebSphere Commerce para habilitar servicios Web, así como referencias relacionadas. Esta guía también describe los servicios Web de ejemplo suministrados, junto con sus objetos de negocio y mandatos de tarea asociados.

Esta guía está dividida en las siguientes secciones:

**Capítulo 1. Introducción** Una breve visión general de los servicios Web, así como una definición de los términos utilizados en esta guía, prerequisites y referencias relacionadas.

**Capítulo 2. Servicios Web para WebSphere Commerce** Una breve visión general de la infraestructura de mensajería de WebSphere Commerce y la arquitectura de servicios Web para WebSphere Commerce. Describe la metodología a seguir para exponer los procesos de negocio de WebSphere Commerce como servicios Web, y la metodología que habilita el acceso de WebSphere Commerce a los servicios Web alojados en sistemas externos.

**Capítulo 3. Habilitación de WebSphere Commerce como proveedor de servicios Web** Describe cómo habilitar un proceso de negocio definido por WebSphere Commerce como un servicio Web utilizando una definición de ejemplo. Se describen los nuevos elementos de programa incorporados a WebSphere Commerce y los detalles de implementación de los servicios Web de ejemplo OrderCreate y OrderStatus.

**Capítulo 4. Habilitación de WebSphere Commerce como peticionario de servicios Web** Describe cómo puede habilitar WebSphere Commerce para invocar un servicio definido por un sistema externo, utilizando una definición de ejemplo. Se describen los nuevos elementos de programa incorporados a WebSphere Commerce y los detalles de implementación del servicio Web de ejemplo OrderFulfillment.

**Apéndice A. Archivos de definición WSDL de ejemplo** Lista los archivos de definición WSDL para los servicios Web de ejemplo.

**Apéndice B. Utilización de IBM WebSphere Application Developer para crear código de cliente de servicio Web** Describe cómo utilizar IBM WebSphere Studio para generar el código de cliente de servicio Web.

---

## Conocimientos necesarios

Para poder utilizar la información de esta guía, debe tener conocimientos de los siguientes temas:

- Tecnologías de servicios Web, como SOAP, WSDL y UDDI
- XML
- Java
- Tecnología JavaServer Pages
- WebSphere Studio Application Developer (también denominado WebSphere Studio en este documento)
- Modelo de programación de WebSphere Commerce

## Convenios utilizados en esta guía

Esta guía utiliza los siguientes convenios:

<b>Negrita</b>	indica mandatos o controles de interfaz gráfica de usuario (GUI), como nombres de campos, botones u opciones de menú.
monoespaciado	indica ejemplos de texto que se escriben exactamente tal como se muestran, así como para nombres de archivos y vías de acceso de directorios.
<i>Cursiva</i>	se utiliza para dar mayor énfasis y para variables para las que el usuario debe especificar sus propios valores.
 AIX	indica información específica de WebSphere Commerce para AIX.
 400	indica información específica de WebSphere Commerce para IBM @server iSeries (antes denominado AS/400).
 Linux	indica información específica de WebSphere Commerce para xSeries Linux, información específica de WebSphere Commerce para @server zSeries y S/390 Linux, información específica de WebSphere Commerce para @server iSeries Linux e información específica de WebSphere Commerce para @server pSeries Linux.
 Solaris	indica información específica de WebSphere Commerce para el software Solaris Operating Environment.
 Windows	indica información específica de WebSphere Commerce para Windows.
 DB2	indica información específica de DB2 Universal Database.
 Oracle	indica información específica de la base de datos Oracle®.
 Business	indica información específica de WebSphere Commerce Business Edition.
 Studio	hace referencia a WebSphere Commerce Studio, Versión 5.5 (Professional o Business Developer Editions).

## Vías de acceso por omisión

Esta guía utiliza las siguientes vías de instalación por omisión:

*dir\_instal\_WC* Indica la vía de instalación para WebSphere Commerce. Cuando vea esta variable, sustituya la vía de instalación por la de su instalación de WebSphere Commerce. Por ejemplo, sustituya

▶ Windows	C:\IBM\WebSphere\CommerceServer55
▶ AIX	/usr/WebSphere/CommerceServer55
▶ Linux	/opt/WebSphere/CommerceServer55
▶ Solaris	/opt/WebSphere/CommerceServer55
▶ 400	/QIBM/ProdData/CommerceServer55

*dir\_instal\_WAS* Indica la vía de instalación para WebSphere Application Server. Cuando vea esta variable, sustituya la vía de instalación por la de su instalación de WebSphere Application Server. Por ejemplo, sustituya

▶ Windows	C:\IBM\WebSphere\AppServer
▶ AIX	/usr/WebSphere/AppServer
▶ Linux	/opt/WebSphere/AppServer
▶ Solaris	/opt/WebSphere/AppServer
▶ 400	/QIBM/ProdData/WebAS5/Base

*dir\_instal\_WCStudio* Indica la vía de instalación para WebSphere Commerce Studio. Cuando vea esta variable, sustituya la vía de instalación por la de su instalación de WebSphere Commerce Studio. Por ejemplo, sustituya

▶ Studio	C:\IBM\WebSphere\CommerceStudio55
----------	-----------------------------------

*dir\_espacio\_trabajo\_WCStudio* es el directorio para el espacio de trabajo de WebSphere Commerce Studio. Por ejemplo, sustituya

▶ DB2	C:\WebSphere\workspace_db2
▶ Oracle	C:\WebSphere\workspace_oracle

*dir\_usuario\_WAS* Indica el directorio para todos los datos que utiliza WebSphere Application Server que pueden modificarse o que el usuario debe configurar. Cuando vea esta variable, sustituya,

▶ 400	/QIBM/UserData/WebAS5/Base/ <i>nombre_instancia_WAS</i>
-------	---

donde *nombre\_instancia\_WAS* es la variable que representa el nombre del WebSphere Application Server al que está asociado su instancia de WebSphere Commerce.

---

## Resumen de cambios

- + La versión más reciente de este documento se encuentra, en formato de archivo PDF, en la página de la biblioteca técnica del sitio Web de WebSphere Commerce:
- + <http://www.ibm.com/software/commerce/library/>
- + Para conocer los cambios de última hora efectuados en el producto, consulte el archivo README actual del producto, que también se encuentra disponible en el sitio Web de WebSphere Commerce.
- + Esta publicación utiliza el siguiente convenio para los caracteres de revisión:
  - El carácter "+" (más) indica actualizaciones que se han efectuado en la revisión actual de este documento.
- + Las actualizaciones en la revisión actual de esta guía son específicas de Linux.

---

# Capítulo 1. Introducción

Este capítulo ofrece una visión general de los servicios Web. Define los términos utilizados en esta guía, lista los prerrequisitos y proporciona referencias a otras fuentes de información relacionada.

---

## Terminología

En esta guía se utilizan los siguientes términos:

### **Proveedor de servicios**

Un proveedor de servicios es el propietario del servicio Web. El proveedor de servicios crea un servicio Web, publica el servicio en un registro de servicios o directamente en los clientes de servicios, y aloja la plataforma que proporciona acceso al servicio.

### **Peticionario de servicios**

Un petionario de servicios es la aplicación que inicia una interacción con un servicio Web. Un ejemplo de un petionario de servicios sería el clásico navegador de Internet. Otro ejemplo sería una aplicación que utiliza los servicios Web para recopilar información necesaria para sus procesos. El petionario de servicios enlaza con el servicio utilizando la información publicada e invoca el servicio.

### **SOAP (Simple Object Access Protocol)**

SOAP define un modelo para utilizar mensajes simples petición-respuesta que están escritos en XML como protocolo básico para comunicaciones electrónicas. Constituye un mecanismo de transporte independiente de la plataforma y el protocolo subyacentes. Facilita las operaciones de publicación, búsqueda, enlace e invocación.

### **UDDI (Universal Description, Discovery, and Integration)**

El depósito de UDDI almacena las descripciones de los servicios Web. UDDI es una especificación que define un registro de servicios de los servicios Web disponibles, y funciona como una colección de páginas amarillas electrónicas globales. UDDI permite que una empresa pueda publicar en el registro una descripción de los bienes o servicios de los que dispone, anunciándose como proveedor de servicios.

### **Servicio Web**

Un servicio Web es un componente de software, basado en una interfaz publicada, que describe un conjunto de operaciones que son accesibles desde la red utilizando mensajería XML estándar.

### **WSDL (Web Services Description Language)**

WSDL proporciona un método para describir las interfaces específicas de servicios Web e Interfaces de programas de aplicación (API).

### **Plantilla de correlación XML**

Las plantillas de correlación XML (o plantillas de correlación) se utilizan para procesar los mensajes XML de entrada y convertirlos en los parámetros de mandatos de WebSphere Commerce. El archivo de plantilla de correlación de WebSphere Commerce define cómo correlacionar los elementos de un mensaje XML de entrada con los parámetros del mandato de destino.

Para obtener más información acerca de la plantilla de correlación, consulte la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

### **Correlacionador de mensajes XML**

El correlacionador de mensajes XML de WebSphere Commerce es el responsable de convertir los mensajes XML de entrada al objeto Java que el mandato de destino espera. Un correlacionador de mensajes puede estar registrado en el archivo de configuración *nombre\_instancia.xml* y se identifica de forma exclusiva ante un ID de correlacionador de mensajes. El correlacionador de mensajes XML utiliza las correlaciones definidas en el archivo de plantilla de correlación para convertir el mensaje XML al parámetro de mandato de objeto Java correspondiente.

**Nota:** *nombre\_instancia* es el nombre de la instancia de WebSphere Commerce.

Para obtener más información acerca del correlacionador de mensajes, consulte la información sobre el correlacionador de mensajes XML de la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

---

## **Visión general**

Los servicios Web son un nuevo tipo de aplicaciones Web. Son aplicaciones autocontenidas, autodescriptivas y modulares que pueden publicarse, localizarse e invocarse en toda la Web. Los servicios Web realizan funciones que pueden invocarse y comprenden desde una simple petición hasta complicados procesos de negocio. Una vez desplegado y registrado un servicio Web, otras aplicaciones pueden descubrir e invocar el servicio desplegado. Las bases de los servicios Web son estándares como SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) y el registro UDDI (Universal Description, Discovery, and Integration).

WebSphere Commerce Versión 5.5 incluye código y documentación que le permite habilitar las funciones de negocio de WebSphere Commerce como servicios Web. Puede permitir que WebSphere Commerce sea el proveedor de servicios habilitando sus funciones de negocio como servicios Web a los que puedan tener acceso sistemas externos. También puede hacer que WebSphere Commerce sea el peticionario de los servicios habilitándolo para que invoque los servicios Web alojados en sistemas externos.

### **Por qué utilizar servicios Web**

Los servicios Web permiten que las aplicaciones se integren con mayor rapidez, facilidad y con un coste inferior del experimentado hasta ahora. La integración se produce a un nivel más alto en la pila de protocolos, basándose más en la semántica del servicio que en la semántica del protocolo de red, lo cual permite una integración mas permisiva de las funciones de negocio. Estas características son ideales para conectar funciones de negocio en la red, tanto entre varias empresas como dentro de una sola empresa. Proporcionan un modelo de programación unificado, de manera que la integración de aplicaciones, tanto dentro como fuera de la empresa, puede realizarse con un enfoque común, sacando el máximo partido a una infraestructura común. La integración y aplicación de servicios Web puede realizarse de forma incremental, utilizando los idiomas y plataformas existentes y adoptando las aplicaciones existentes.

Un servicio Web es una interfaz que describe un conjunto de operaciones a las que puede accederse desde la red utilizando mensajería XML estándar. Un servicio Web se describe utilizando una notación XML formal, estándar, denominada descripción de servicio. La descripción de servicio incluye todos los detalles necesarios para interactuar con el servicio, incluyendo los formatos de los mensajes (que detallan las operaciones), la ubicación y los protocolos de transporte. La interfaz oculta los detalles de implementación del servicio, lo que permite utilizarlo independientemente de la plataforma de hardware o software sobre la que se implementa; también es independiente del lenguaje de programación en el que está escrito.

WebSphere Commerce está diseñado para dar soporte a servicios Web, tanto como proveedor de servicios Web como peticionario (cliente) de otros servicios Web. Por ejemplo, los clientes de servicios Web externos pueden conectarse a WebSphere Commerce y utilizar los servicios Web que se proporcionan (tal como se muestra en la Figura 1). Asimismo, WebSphere Commerce puede conectarse a otros proveedores de servicios Web y solicitar información (tal como se muestra en la Figura 2).

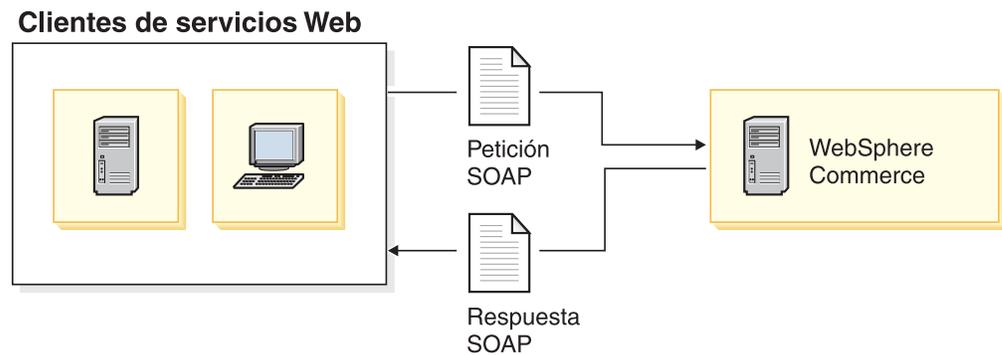


Figura 1. WebSphere Commerce como proveedor de servicios

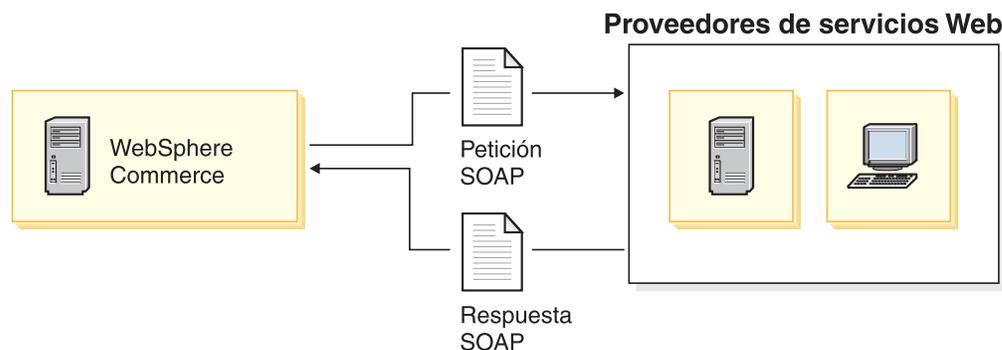


Figura 2. WebSphere Commerce como peticionario de servicios

## Arquitectura de los servicios Web

La arquitectura de los servicios describe tres roles, proveedor de servicios, peticionario de servicios e intermediario de servicios, y tres operaciones básicas, publicación, búsqueda y enlace. Un componente de red puede desempeñar uno de estos roles o todos ellos. Los proveedores de servicios publican los servicios Web

en un intermediario de servicios. Los peticionarios de servicios buscan los servicios Web que necesitan utilizando un intermediario de servicios y enlazando con ellos.

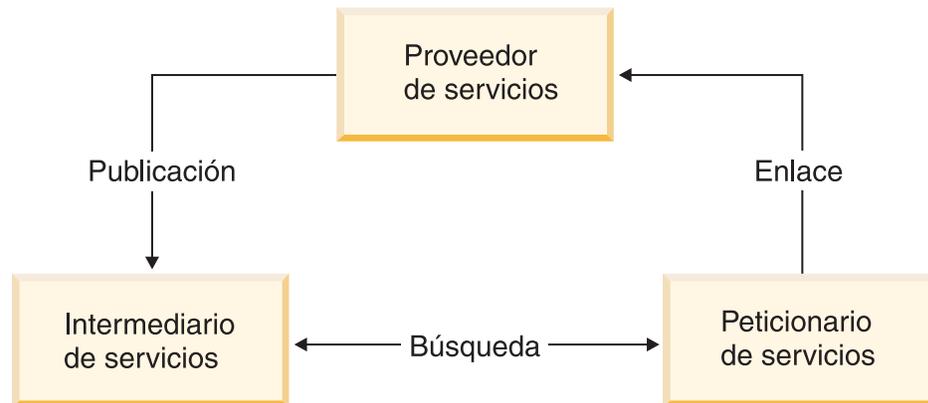


Figura 3. Arquitectura de los servicios Web

En un escenario típico, un proveedor de servicios aloja un módulo de servicio accesible desde la red, que es una implementación de un servicio Web. El proveedor de servicios define una descripción del servicio Web y la publica en un peticionario de servicios o en un registro de servicios. El peticionario de servicios utiliza una operación de búsqueda para recuperar la descripción del servicio de forma local o desde el registro de servicios alojado en el intermediario de servicios. El peticionario de servicios utiliza la descripción del servicio para enlazar con el proveedor de servicios e invocar, o interactuar, con la implementación del servicio Web.

Esta solución proporciona soporte en tiempo de ejecución para habilitar procesos de negocio que están definidos en WebSphere Commerce como servicios Web. Para obtener ejemplos sobre cómo utilizar mandatos de WebSphere Commerce existentes como servicios Web, consulte el Capítulo 3, “Habilitación de WebSphere Commerce como proveedor de servicios”, en la página 11. Para obtener ejemplos sobre cómo puede invocar WebSphere Commerce servicios Web definidos por sistemas externos, consulte el Capítulo 4, “Habilitación de WebSphere Commerce como peticionario de servicios”, en la página 35.

---

## Prerrequisitos

Antes de utilizar la información que se proporciona en esta guía, asegúrese de haber instalado IBM WebSphere Commerce Versión 5.5. Cuando instale WebSphere Commerce Versión 5.5, debe seleccionar la opción de instalación **Personalizada** para instalar los servicios Web de ejemplo que se proporcionan. Se le solicitará que seleccione los componentes que desea instalar. Seleccione **Archivos de ejemplo de WebSphere Commerce**. Para obtener más información, consulte la publicación *WebSphere Commerce, Guía de instalación*.

Alternativamente, si está utilizando WebSphere Commerce Studio para habilitar los servicios Web de ejemplo, asegúrese de haber completado la instalación de WebSphere Commerce Studio Versión 5.5. Para obtener más información, consulte la publicación *WebSphere Commerce Studio, Guía de instalación*.

Si desea modificar los servicios Web de ejemplo, asegúrese de estar utilizando WebSphere Commerce Studio como entorno de desarrollo.

---

## Referencias

Para obtener más información acerca de los servicios Web y sus tecnologías relacionadas, puede consultar los siguientes sitios Web:

- Para obtener información sobre SOAP de Apache, una implementación de la inclusión de SOAP en W3C, consulte <http://ws.apache.org/soap/>
- Para obtener información detallada sobre la especificación SOAP, consulte <http://www.w3.org/TR/SOAP/>
- Para obtener información sobre UDDI, consulte <http://www.uddi.org/>
- Para obtener información sobre WSDL, consulte <http://www.uddi.org/specification.html>
- Para ampliar sus conocimientos sobre servicios Web en IBM Alphaworks, consulte <http://www.alphaworks.ibm.com/webservices>
- Para ampliar sus conocimientos sobre servicios Web desde la perspectiva de los desarrolladores, consulte <http://www.ibm.com/developerworks/webservices>
- Para obtener información completa sobre el software IBM WebSphere Commerce Business Edition, consulte [http://www.ibm.com/software/webservers/commerce/wc\\_be/](http://www.ibm.com/software/webservers/commerce/wc_be/)
- Para obtener información completa sobre el software IBM WebSphere Commerce Professional Edition, consulte [http://www.ibm.com/software/webservers/commerce/wc\\_pe/](http://www.ibm.com/software/webservers/commerce/wc_pe/)
- Para obtener información de soporte de business partners para desarrollar soluciones de servicios Web de WebSphere de calidad, consulte <http://www.ibm.com/websphere/wow>
- Para obtener información sobre IBM WebSphere Studio, consulte [www.ibm.com/software/webservers/commerce/commercestudio/](http://www.ibm.com/software/webservers/commerce/commercestudio/)

**Nota:** Las direcciones Web aquí mencionadas podrían cambiarse en cualquier momento sin previo aviso. IBM no es responsable de la autenticidad o exactitud de la información existente en sitios Web que no son de IBM.



---

## Capítulo 2. Servicios Web para WebSphere Commerce

Este capítulo explica brevemente la infraestructura de mensajería de WebSphere Commerce y le ayuda a comprender mejor la arquitectura de servicios Web para WebSphere Commerce. También describe la metodología a seguir para exponer los procesos de negocio de WebSphere Commerce como servicios Web y cómo puede acceder WebSphere Commerce a los servicios Web alojados en sistemas externos.

---

### Infraestructura de mensajería de WebSphere Commerce

La arquitectura de ejecución de WebSphere Commerce incluye escuchas de protocolo, adaptadores, controladores, mandatos y vistas. La infraestructura de mensajería de WebSphere Commerce utiliza estos componentes.

Un escucha de protocolo es un componente de ejecución de WebSphere Commerce que recibe peticiones de entrada de los transportes y las distribuye a los adaptadores correspondientes, en función del protocolo que utilizan.

En WebSphere Commerce Versión 5.5, el servlet de peticiones `RequestServlet` tiene la función de escucha de protocolo que recibe los mensajes de SOAP versión 1.1. Cuando `RequestServlet` recibe una petición XML, pasa dicha petición al gestor de adaptadores. A continuación, el gestor de adaptadores consulta los tipos de adaptador disponibles para determinar qué adaptador puede procesar la petición. Una vez se ha determinado un adaptador específico, la petición se pasa a dicho adaptador.

El adaptador de programa se invoca cuando se reciben mensajes XML a través del protocolo HTTP. Permite que los sistemas externos se comuniquen con WebSphere Commerce pasando las peticiones XML a través del protocolo HTTP. El adaptador de programa proporciona a los sistemas externos, como sistemas de compras, sistemas de suministros y otros, una manera común para comunicarse con WebSphere Commerce a través del protocolo HTTP. De esta manera, WebSphere Commerce se integra de forma transparente con los sistemas externos.

Para obtener más información sobre el flujo de una petición, consulte la publicación *WebSphere Commerce, Guías de programación y aprendizaje*.

---

### Descripción de la implementación de servicios Web para WebSphere Commerce

El modelo de programación de WebSphere Commerce consta de mandatos URL, mandatos de controlador, mandatos de tarea, mandatos de vista, un adaptador de programa, un controlador Web y otros elementos. Para obtener detalles sobre el modelo de programación de WebSphere Commerce, consulte la publicación *WebSphere Commerce, Guías de programación y aprendizaje*.

Las secciones siguientes explican cómo encaja WebSphere Commerce en los roles de proveedor de servicios y peticionario de servicios.

### WebSphere Commerce como proveedor de servicios

Cuando habilita los procesos de negocio de WebSphere Commerce como servicios Web a los que puede accederse desde sistemas externos, WebSphere Commerce se

convierte en un proveedor de servicios. WebSphere Commerce Versión 5.5 incluye mejoras en el sistema de mensajería para dar soporte a mensajes y respuestas SOAP. Además, los ejemplos suministrados muestran cómo puede exponer la lógica de negocio de WebSphere Commerce existente como servicios Web.

Un cliente pueden invocar un mandato de URL a través de la red utilizando los protocolos HTTP o HTTP seguro. WebSphere Commerce también puede recibir mensajes XML a través de los protocolos HTTP y HTTPS, o a través del middleware de mensajería. Esta capacidad se ha ampliado aún más para que WebSphere Commerce pueda recibir mensajes SOAP a través de HTTP; esta funcionalidad se utiliza para habilitar servicios Web. RequestServlet es el punto de entrada para todas las peticiones de URL en el sistema WebSphere Commerce. Cuando RequestServlet recibe un mensaje XML, identifica el tipo de contenido como XML e invoca el adaptador de programa.

### **Proceso de la petición**

El adaptador de programa proporciona soporte para programas remotos que invocan mandatos de WebSphere Commerce. Recibe las peticiones y utiliza un correlacionador de mensajes para convertir la petición en la entrada adecuada para un mandato. Una vez realizada la conversión, el adaptador de programa ejecuta el mandato.

En WebSphere Commerce Versión 5.5, se ha añadido un nuevo correlacionador de mensajes y un archivo de plantilla de correlación a la arquitectura de ejecución de WebSphere Commerce, lo cual facilita el análisis y el proceso de los mensajes XML SOAP.

Este nuevo archivo de plantilla de correlación contiene las correlaciones entre los mensajes XML SOAP de entrada y los parámetros del mandato. El nuevo correlacionador de mensajes examina este archivo de plantilla de correlación y analiza el mensaje XML SOAP convirtiéndolo en entrada adecuada para un mandato; a continuación, pasa de nuevo la entrada al adaptador de programa. Entonces, el adaptador de programa invoca el controlador Web que, a su vez, ejecuta el mandato especificado por el correlacionador de mensajes.

### **Respuesta**

En WebSphere Commerce, cuando el mandato de controlador se ejecuta satisfactoriamente devuelve el nombre de una vista que debe ejecutarse. El controlador Web invoca el mandato de vista para la vista correspondiente y compone el mensaje que va a devolverse. Hay varias maneras de formar una vista de respuesta. Entre ellas, redirigir a un URL distinto, reenviar a una plantilla JSP o escribir un documento HTML para la respuesta.

Para poder generar mensajes para distintas clases de dispositivos cliente, WebSphere Commerce da soporte a diferentes formatos de dispositivo. El controlador Web selecciona un nombre de vista correspondiente al formato de dispositivo del mensaje de entrada. En WebSphere Commerce Versión 5.5, se ha incorporado un nuevo formato de dispositivo, SOAPHTTP, para el mensaje XML SOAP. Puede definirse un nuevo conjunto de plantillas JSP para los mismos nombres de vistas, pero con un nuevo formato de dispositivo. Estas plantillas JSP componen la respuesta SOAP que debe devolverse al cliente SOAP. Esto sigue el patrón de controlador de la vista del modelo y facilita la separación de la vista de la lógica de negocio.

Se proporciona una nueva implementación de mandato de vista para manejar las respuestas SOAP. La interfaz para esta implementación del mandato de vista se

denomina `com.ibm.commerce.me.soap.commands.SOAPViewCommand` y el nombre de la clase de implementación es `com.ibm.commerce.me.soap.commands.HttpSOAPViewCommandImpl`. Por tanto, siempre que se registre una nueva plantilla JSP en la tabla VIEWREG para componer el mensaje de respuesta SOAP, debe utilizarse este nombre de interfaz y nombre de clase.

La nueva implementación del mandato de vista SOAP puede manejar peticiones de redirección HTTP y también de reenvío HTTP. Normalmente, las redirecciones HTTP las maneja una redirección de URL. Puesto que un cliente SOAP no puede comprender una redirección de URL, este mandato de vista SOAP implementa también la redirección HTTP como un reenvío HTTP.

## Habilitación de WebSphere Commerce como proveedor de servicios

A continuación se ofrece una visión general de los pasos que le permiten hacer que un proceso de negocio de WebSphere Commerce esté disponible como un servicio Web:

1. Identifique la lógica de negocio que desea exponer como servicio Web.
2. Identifique el mandato de URL que implementa esta lógica de negocio. Si no está disponible, cree un nuevo mandato de URL. Puede escribir nuevos mandatos para llamar a una colección de mandatos ya existentes.
3. Identifique los parámetros obligatorios y opcionales que este mandato de URL necesita.
4. Establezca la definición WSDL para el servicio Web.
5. Correlacione la petición SOAP de entrada con los parámetros del mandato, utilizando para ello el archivo de plantilla de correlación. Asegúrese de que la plantilla contiene correlaciones para todos los elementos presentes en la definición WSDL. Para ello, deberá insertar datos para todos los elementos de la petición SOAP cuando envíe la petición desde el cliente. El archivo de plantilla de correlación define cómo se correlaciona cada elemento del mensaje XML SOAP con un parámetro de mandato utilizando correlaciones de XPATH. Puede interceptar la petición SOAP utilizando un programa de utilidad como Supervisor TCP y, a continuación, utilizar el mensaje interceptado para definir la correlación.

**Nota:** Supervisor TCP está disponible con WebSphere Studio Application Developer 5.0. Consulte la ayuda de WebSphere Studio para obtener más información. Con la distribución de AXIS de Apache se suministra una versión autónoma del Supervisor TCP. Para más detalles, consulte <http://ws.apache.org/axis/>.

6. Escriba una plantilla JSP para componer una respuesta.
7. Despliegue el servicio. Esto incluye el despliegue de los archivos y recursos relacionados. En caso de un nuevo mandato, debe desplegarlo y registrarlo en el registro de mandatos de WebSphere.

La lógica de negocio en WebSphere Commerce se implementa utilizando mandatos de controlador. Un mandato de controlador es un mandato que interactúa directamente con el controlador Web. Un mandato de URL es un mandato de controlador que puede invocarse utilizando un URL. Como resultado, puede invocar un mandato de URL entrando un URL en un navegador Web. Antes de implementar un servicio Web, identifique el mandato de URL que implementa la

lógica de negocio que desea exponer como servicio Web. Por ejemplo, el mandato `OrderStatus` le permite actualizar el estado de un pedido existente en WebSphere Commerce.

Para obtener más información sobre los mandatos de URL, consulte la sección Referencias de la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

## WebSphere Commerce como peticionario de servicios

Cuando utiliza WebSphere Commerce para iniciar un servicio Web alojado en un sistema externo, WebSphere Commerce se convierte en un peticionario de servicios. WebSphere Commerce Versión 5.5 incluye ejemplos que muestran cómo puede integrarse la lógica de negocio de WebSphere Commerce con otros sistemas realizando llamadas de servicios Web. WebSphere Commerce envía las peticiones de servicio a través de los proxy del servicio Web, que envían y reciben peticiones y respuestas basadas en SOAP. La lógica de negocio funcionará basándose en las interfaces de enlace del servicio Web identificado. Los ejemplos proporcionados soportan la invocación de un servicio Web basado en estas interfaces.

Las implementaciones puede publicarse en un registro de servicios o bien almacenarse en el sistema de archivos local. La invocación de un servicio Web desde WebSphere Commerce incluye las siguientes actividades:

1. Identificar la definición de interfaz de servicio Web publicada, que se adapta a los escenarios de negocio existentes a los WebSphere Commerce da soporte.
2. Descargar la definición WSDL de la implementación del servicio a su máquina local de WebSphere Commerce.
3. Registrar el archivo de definición del servicio Web en el archivo de configuración *nombre\_instancia.xml* (donde *nombre\_instancia* es el nombre de la instancia de WebSphere Commerce Server).
4. Crear los proxy de cliente que WebSphere Commerce puede utilizar. Puede generarlos utilizando herramientas como IBM WebSphere Studio. Cuando genere el proxy de cliente para la definición de interfaz de servicio Web, se generarán también los tipos de datos necesarios para los mensajes de entrada y los mensajes de salida. Consulte la sección “Generación de código de cliente para acceder a servicios Web” en la página 14 de este documento para obtener más información.

También puede utilizar otras herramientas para crear el proxy de cliente, por ejemplo, Web Services Toolkit. Consulte <http://www.alphaworks.ibm.com/webservices>

5. Escribir un mandato de tarea que implemente la lógica para que:
  - a. Recupere los puntos de acceso.
  - b. Invoque el proxy para el servicio Web.
  - c. Procese los resultados que devuelve el servicio Web.
6. Actualizar el registro de mandatos de WebSphere Commerce para registrar el mandato de tarea.

---

## Capítulo 3. Habilitación de WebSphere Commerce como proveedor de servicios

Este capítulo describe cómo habilitar un proceso de negocio definido por WebSphere Commerce como un servicio Web. En este escenario, WebSphere Commerce actúa como proveedor de servicios para sistemas externos. Se proporcionan ejemplos que muestran cómo habilitar los procesos de negocio OrderCreate y OrderStatus como servicios Web.

---

### Infraestructura para habilitar servicios Web

Los siguientes elementos son los ejecutables de WebSphere Commerce que dan soporte a una petición de servicio Web de entrada:

- Correlacionador de mensajes SOAP
- Plantilla de correlación de servicios Web
- Mandatos de vista de respuesta SOAP

Los usuarios externos pueden invocar los servicios de WebSphere Commerce utilizando un programa cliente. Estos usuarios pueden conectarse con cualquier sistema que sea capaz de enviar y recibir mensajes SOAP. WebSphere Commerce versión 5.5 da soporte a los mensajes SOAP a través del protocolo de transportes HTTP.

Los servicios de WebSphere Commerce deben publicarse como definiciones WSDL. Los clientes de servicios Web pueden descargar las definiciones WSDL, generar un cliente y llamar al servicio Web de WebSphere Commerce utilizando SOAP a través de HTTP.

### Arquitectura

El flujo de mensajes en esta arquitectura es síncrono. Esto significa que el sistema externo espera la respuesta del sistema WebSphere Commerce antes de continuar procesando. El sistema de mensajería de entrada se utiliza para permitir que los clientes externos invoquen los procesos de negocio de WebSphere Commerce que están publicados como servicios Web.

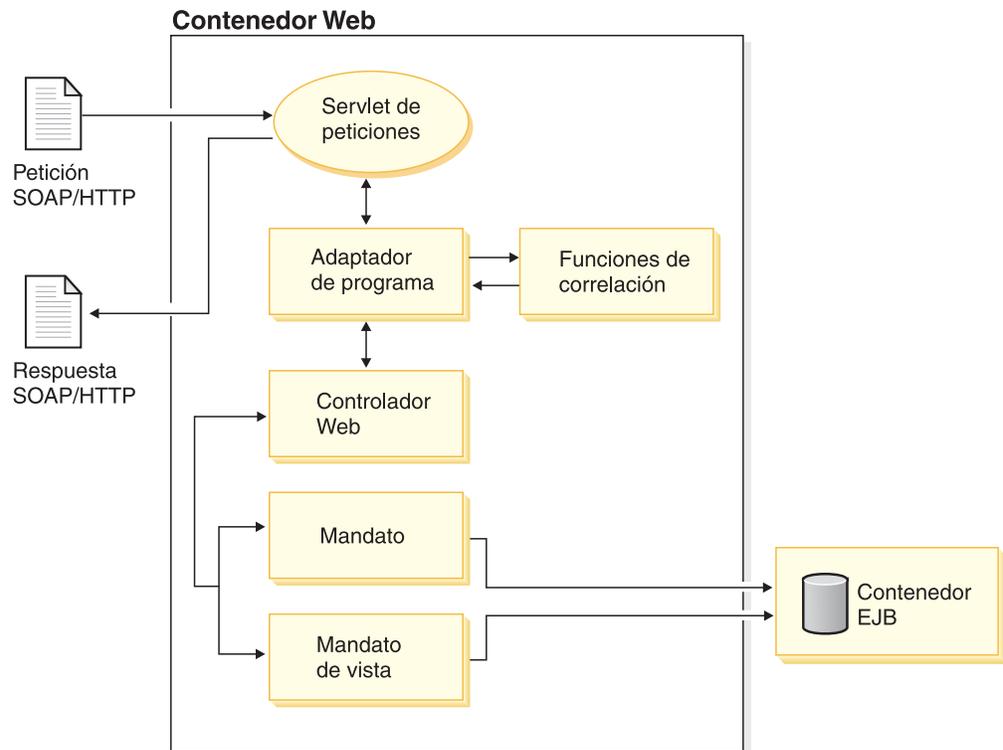


Figura 4. Arquitectura para habilitar WebSphere Commerce como proveedor de servicios

La lista siguiente describe la interacción que se produce cuando un sistema externo invoca un proceso de negocio de WebSphere Commerce utilizando un servicio Web:

1. El sistema externo envía la petición SOAP correctamente formada a WebSphere Commerce.
2. El servlet de peticiones (RequestServlet) recibe la petición y la delega al adaptador de programa.
3. El adaptador de programa invoca el correlacionador de mensajes SOAP registrado.
4. El correlacionador de mensajes SOAP analiza los mensajes SOAP y devuelve las parejas nombre-valor al adaptador de programa utilizando un archivo de plantilla, el cual contiene la información de correlación entre los parámetros de la petición de servicio Web y los parámetros del mandato.
5. El adaptador de programa llama al controlador Web y le pasa el nombre del mandato y las parejas nombre-valor.
6. El controlador Web ejecuta el mandato. En el ejemplo suministrado, se invocan los mandatos BatchOrderRequest y OrderStatusCmd.
7. Después de ejecutar el mandato, se devuelve el nombre de la vista al controlador Web.
8. El controlador Web ejecuta el mandato de vista registrado para formar la respuesta SOAP y la devuelve al adaptador de programa.
9. El adaptador de programa HTTP devuelve el mensaje a RequestServlet.
10. RequestServlet envía la respuesta al cliente externo que está en espera.

Se han añadido los siguientes ejemplos para dar soporte al servicio Web que proporciona el proveedor de servicios:

- Se proporciona el archivo `webservice_template.xml` para correlacionar XPath con las propiedades del mandato.
- Se proporcionan las siguientes plantillas JSP de ejemplo para componer la respuesta SOAP y los mensajes de error:
  - `SOAP_OrderStatusResponse.jsp`
  - `SOAP_PurchaseOrderResponse.jsp`
  - `SOAP_GenericApplicationError.jsp`
  - `SOAP_GenericSystemError.jsp`

---

## Consideraciones de seguridad

Las secciones siguientes tratan sobre las consideraciones de seguridad que afectan a la habilitación de los servicios Web.

### Servicios Web y control de acceso

El control de acceso es el proceso de restringir el acceso a los recursos protegidos en el sistema WebSphere Commerce, limitándolo a personas individuales y organizaciones. Para facilitar la gestión de las bases de datos y garantizar la seguridad, WebSphere Commerce dispone de un mecanismo de control de acceso que restringe el acceso a los recursos para usuarios específicos. El control de acceso puede definirse como las directrices de seguridad que:

1. Permiten o niegan a un usuario de un sistema el acceso a los recursos gestionados por un sistema.
2. Especifican qué acciones puede realizar el usuario en cada recurso.

El modelo de control de acceso para WebSphere Commerce está basado en la aplicación de políticas de control de acceso. Para ampliar la información sobre el control de acceso y sus políticas, consulte la publicación *WebSphere Commerce, Guía de seguridad*.

Cuando WebSphere Commerce actúa como proveedor de servicios, permite que los clientes de servicios Web externos invoquen procesos de negocio que están definidos mediante mandatos de controlador. En consecuencia, los servicios Web heredan las políticas de control de acceso que son aplicables al mandato invocado por los clientes de servicios Web. Por lo tanto, debe definirse la información de credenciales para cada definición WSDL que se utilice con WebSphere Commerce. Además de pasar las credenciales necesarias para autenticar al usuario, el cliente de servicio Web debe asegurarse de que los usuarios tiene la autorización necesaria para ejecutar los mandatos.

El formato del esquema XML en la definición WSDL para las credenciales debe parecerse al siguiente:

```
<complexType
name="com.ibm.commerce.webservice.datatype.Credentials">
<all>
<element name="password" nillable="true" type="string"/>
<element name="logonId" nillable="true" type="string"/>
</all>
</complexType>
```

La información de credenciales debe correlacionarse con los siguientes campos en la definición de plantilla disponible en el archivo de plantilla de correlación:

- El nombre del campo para la conexión del usuario es 'logonId' y FieldInfo es 'CONTROL'.
- El nombre del campo para la contraseña es 'logonPassword' y FieldInfo es 'CONTROL'.

## Seguridad del nivel de transporte

Para habilitar la seguridad del nivel de transporte, los mensajes SOAP pueden enviarse a través de HTTPS. Es altamente recomendable que el proveedor de servicios utilice HTTPS para enviar los mensajes SOAP; de lo contrario, cualquier sujeto que interceptara los mensajes podría ver las credenciales que se están pasando.

**Nota:** Para habilitar HTTPS, el cliente SOAP debe utilizar el URL con HTTPS como nombre de protocolo. Antes de invocar la llamada desde el cliente, establezca las propiedades para habilitar SSL (Secure Socket Layer). Si está utilizando IBM Developer Kit, Java Technology Edition, Versión 1.2 o posterior, para el cliente de servicio Web, el código para habilitar HTTPS será parecido al siguiente:

```
System.setProperty("java.protocol.handler.pkgs", "com.ibm.net.ssl.internal.www.protocol");
```

```
java.security.Security.addProvider
(new com.ibm.jsse.JSSEProvider());
```

El código anterior se utiliza si emplea WebSphere Commerce Studio para desarrollar el cliente de servicio Web.

Si está utilizando J2SE SDK (Java 2 Standard Edition Software Development Kit) de Sun Microsystems, el código de cliente para invocar el servicio Web a través de HTTPS será parecido al siguiente:

```
System.setProperty("java.protocol.handler.pkgs", "com.sun.net.ssl.internal.www.protocol");
```

```
java.security.Security.addProvider
(new com.sun.net.ssl.internal.ssl.Provider());
```

Si recibe un error del tipo 'error al establecer comunicación', 'CA desconocida' o 'conexión rechazada por excepción del servidor' ('handshake failed', 'unknown CA', 'connection refused by server exception'), en el lado del cliente cuando intentaba invocar un servicio de entrada definido por WebSphere Commerce, significa que el cliente no reconoce el certificado que presenta el servidor. Si es este el caso y confía en el servidor, puede añadir el certificado del servidor a su lista de certificados de confianza.

---

## Generación de código de cliente para acceder a servicios Web

Esta sección describe cómo generar el código de cliente.

Los servicios Web de ejemplo se publican utilizando los archivos de definición WSDL y están situados en el siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\messaging`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
-    `dir_instal_WC/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`

Puede utilizar la definición WSDL para desarrollar aplicaciones cliente como peticionarios de servicios Web. Puede generar el código de cliente y los tipos de datos necesarios para acceder a los servicios Web que implementa WebSphere Commerce a partir de la definición WSDL del servicio Web, utilizando la herramienta adecuada.

IBM WebSphere Studio Application Developer (también denominado WebSphere Studio en este documento) es una de las herramientas que puede utilizar para generar el código de cliente y de proxy a partir de la definición WSDL. Para obtener más información, consulte el "Apéndice B. Utilización de IBM WebSphere Studio Application Developer" en la página 47.

Otra herramienta que puede utilizar para generar el proxy de cliente es Web Services Toolkit. Consulte <http://www.alphaworks.ibm.com/webservices>.

---

## Servicios Web de ejemplo que pueden invocar los clientes Web externos

Esta sección describe cómo puede habilitar los siguientes servicios Web de ejemplo:

1.  Servicio OrderCreate
2. Servicio OrderStatus

Puede habilitar los servicios Web de ejemplo en una máquina de WebSphere Commerce o en una máquina de desarrollo instalada con WebSphere Commerce Studio. Si está habilitando los servicios Web de ejemplo en una máquina de WebSphere Commerce Studio, siga las instrucciones de esta sección y consulte las vías de acceso a los directorios señalados con el símbolo  .

### Importación del código de ejemplo

Si desea editar o modificar los servicios Web de ejemplo, debe importar el código del ejemplo en WebSphere Commerce Studio. Para importar este código, haga lo siguiente:

1. Abra el entorno de desarrollo de WebSphere Commerce desde **Inicio > Programas > IBM WebSphere Commerce Studio > Entorno de desarrollo de WebSphere Commerce**.
2. Vaya a la perspectiva J2EE y seleccione la vista Navegador J2EE.
3. Expanda el proyecto **WebSphereCommerceServerExtensionsLogic**. Con el botón derecho del ratón, pulse sobre el directorio **src** y seleccione **Importar**. Se abrirá el asistente de importación.
4. Seleccione **Sistema de archivos** y pulse **Siguiente**.
5. Pulse **Examinar** y localice el siguiente directorio:  
`dir_instal_WCStudio\Commerce\samples\webservices\com`
6. Pulse **Seleccionar todo** y, a continuación, **Finalizar**.
7. Con el botón derecho del ratón, pulse sobre el proyecto **WebSphereCommerceServerExtensionsLogic** y seleccione **Propiedades**.
8. Pulse **Vía de construcción Java** y vaya a la pestaña **Bibliotecas**.
9. Pulse **Añadir variable**. Seleccione las variables SOAPJAR y XERCES y pulse **Aceptar**.
10. Pulse **Aceptar**.

11. Con el botón derecho del ratón, pulse sobre el proyecto **WebSphereCommerceServerExtensionsLogic** y seleccione **Reconstruir proyecto**.

Ha finalizado la importación del código fuente del ejemplo en WebSphere Commerce Studio. Para obtener más información sobre cómo iniciar el servidor WebSphereCommerceServer en el Entorno de prueba de WebSphere, consulte la publicación *WebSphere Commerce Studio, Guía de instalación*

## Servicio OrderCreate (Business Edition)

El servicio Web OrderCreate es aplicable a WebSphere Commerce Business Edition y WebSphere Commerce Business Developer Edition.

Esta sección muestra cómo habilitar la lógica de negocio en WebSphere Commerce como un servicio Web que dé soporte a la recepción de pedidos de aplicaciones externas, como por ejemplo aplicaciones de compras. El sistema de compras puede crear pedidos en WebSphere Commerce utilizando el mandato BatchOrderRequest de WebSphere Commerce. La implementación de este servicio Web OrderCreate de ejemplo describe cómo crear un servicio Web que reciba pedidos de una organización compradora. Para utilizar el servicio Web OrderCreate de ejemplo, necesita tener una tienda publicada dentro del sistema WebSphere Commerce.

Habitualmente, el vendedor debe conocer a la organización compradora antes de crear un pedido en WebSphere Commerce. De la misma manera, la organización compradora debe registrarse en WebSphere Commerce antes de que el comprador pueda llamar al servicio Web OrderCreate.

### Habilitación del servicio Web OrderCreate de ejemplo

Para habilitar el servicio Web OrderCreate de ejemplo, haga lo siguiente:

1. Identifique la lógica de negocio que desea exponer como servicio Web. En este caso, permitir que una organización compradora cree una orden de compra utilizando una aplicación similar a un sistema de compras.
2. Identifique el mandato que implementa esta lógica de negocio. En este caso, el mandato BatchOrderRequest.
3. Identifique los parámetros obligatorios y opcionales que el mandato BatchOrderRequest necesita. Hay disponible una descripción del mandato BatchOrderRequest en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce. Para obtener una lista completa de parámetros, consulte el archivo `webservice_template.xml` que se proporciona en el siguiente directorio:
  -  `dir_instal_WC\samples\webservices\xml\messaging`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
  -    `dir_instal_WC/samples/webservices/xml/messaging`
  -  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`
4. Cree la definición WSDL para el servicio Web OrderCreate. Hay un archivo de definición OrderCreate.wsdl de ejemplo en el siguiente directorio:
  -  `dir_instal_WC\samples\webservices\xml\messaging`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`

-    `dir_instal_WC/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`

Puede modificar esta definición WSDL para cumplir los requisitos de la definición para el servicio que se proporciona.

La definición WSDL publicada para el servicio Web OrderCreate contiene el punto de acceso, la plantilla de enlace y el esquema de tipo de datos necesarios para invocar este servicio Web.

5. Correlacione la petición SOAP de entrada con los parámetros del mandato, utilizando para ello el archivo de plantilla de correlación. Hay una plantilla de correlación de ejemplo en el siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\messaging\webservice_template.xml`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging\webservice_template.xml`
-    `dir_instal_WC/samples/webservices/xml/messaging/webservice_template.xml`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/webservice_template.xml`

Puede personalizar este archivo. Si ha modificado la definición WSDL de ejemplo, es posible que tenga que modificar esta plantilla de correlación para que refleje los cambios realizados en la definición WSDL.

6. Para habilitar esta plantilla, que incluye el servicio Web OrderCreate de ejemplo, haga lo siguiente:

- a. Copie el archivo `webservice_template.xml` en el siguiente directorio:

-  `dir_instal_WC\xml\messaging`
-  `dir_instal_WCStudio\Commerce\xml\messaging`
-    `dir_instal_WC/xml/messaging`
-  `/QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml/messaging`

- b. Si el archivo `webservice_template.xml` ya existe y no contiene ninguna información, sobrescríbalo. Si el archivo contiene información, copie la información de `TemplateDocument` correspondiente al servicio `OrderCreate` en el archivo ya existente.

7. Habilite el correlacionador de mensajes para "SOAPINTEGRATION". Para hacerlo, abra el archivo de configuración `nombre_instancia.xml` situado en el siguiente directorio:

-  `dir_instal_WC\instances\nombre_instancia\xml`
-  `dir_instal_WCStudio\Commerce\instances\nombre_instancia\xml`
-    `dir_instal_WC/instances/nombre_instancia/xml`
-  `/QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml`

Localice el correlacionador de mensajes cuyo nombre es "WCS.SOAPINTEGRATION". Si el valor del atributo "enable" es "false", establézcalo en "true".

8. Escriba las plantillas JSP para componer una respuesta. En este escenario, copie las siguientes plantillas JSP de ejemplo:
  - SOAP\_PurchaseOrderResponse.jsp - componen el mensaje de respuesta SOAP para la petición OrderCreate.
  - SOAP\_GenericApplicationError.jsp - componen el mensaje de error SOAP en caso de que se produzca una error de la aplicación.
  - SOAP\_GenericSystemError.jsp - componen la respuesta de error SOAP en caso de que se produzca un error del sistema.

Realice la copia del siguiente directorio:

-  `dir_instal_WC\samples\webservices\wcstores`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\wcstores`
-    `dir_instal_WC/samples/webservices/wcstores`
-  `dir_instal_WC/samples/webservices/wcstores`

en el siguiente directorio:

-  `dir_instal_WAS\installedApps\nombre_casilla\WC_nombre_instancia.ear\Stores.war`  
donde *nombre\_casilla* es el nombre de la máquina donde ha instalado WebSphere Application Server.
  -  `dir_espacio_trabajo_WCStudio\Stores\Web Content`
-    `dir_instal_WAS/installedApps/nombre_casilla\WC_nombre_instancia.ear/Stores.war`
-  `dir_usuario_WAS/installedApps/nombre_casilla\WC_nombre_instancia.ear/Stores.war`

9. Cree una organización compradora y un contrato entre las organizaciones compradora y proveedora. Para obtener más información, consulte el apartado "Creación de usuarios y contratos" en la página 19 de este documento.
10. Inserte datos en las tablas de base de datos tal como se describe en el apartado "Inserción de datos en las tablas de base de datos" en la página 20 de este documento.
11. Reinicie la instancia de Commerce Server para que los cambios entren en vigor si está utilizando WebSphere Commerce. Si está utilizando una máquina de WebSphere Commerce Studio, reinicie el servidor WebSphereCommerceServer en el Entorno de prueba de WebSphere.

### Prueba del ejemplo

Puede generar el cliente de servicio Web desde la definición WSDL. El cliente puede invocar este servicio Web enviando un mensaje PurchaseOrderRequest. Si el pedido se formaliza satisfactoriamente, el mensaje de respuesta contiene un código de estado (200), un mensaje de estado "OK", texto de estado y el ID del pedido creado en WebSphere Commerce. Si se produce un error, recibirá el mensaje de error correspondiente.

### Seguridad para el servicio Web OrderCreate

La lógica de negocio que se expone como un servicio Web en este ejemplo requiere la autenticación de la organización compradora peticionaria. En el modelo de

control de acceso de WebSphere Commerce, la autenticación de los mandatos y sus servicios Web asociados puede llevarse a cabo proporcionando el ID de usuario y la contraseña de WebSphere Commerce del peticionario. Para ello, es necesario que el usuario esté registrado como usuario de WebSphere Commerce.

La autenticación de un usuario que desea utilizar el servicio Web OrderCreate de ejemplo requiere que la organización compradora a la que pertenece el usuario esté registrada en WebSphere Commerce. Las credenciales que se pasan deben verificarse antes de procesar el servicio OrderCreate.

Las credenciales para el servicio Web OrderCreate incluyen el código de organización y el tipo de código de organización para el proveedor y el comprador en el sistema WebSphere Commerce. El cliente debe enviar el mismo valor de código de organización y tipo de código de organización que el registrado en la tabla ORGCODE.

A continuación se proporciona el esquema de definición WSDL para las credenciales, donde `identity` es el código de organización, `domain` es el tipo de código de organización y el secreto compartido (`sharedSecret`) una contraseña compartida entre las organizaciones compradora y proveedora:

```
<complexType
name="com.ibm.commerce.webservice.datatype.ordercreate.
Credential">
<all>
<element name="identity" nillable="true" type="string"/>
<element name="domain" nillable="true" type="string"/>
<element name="sharedSecret" nillable="true" type="string"/>
</all>
```

**Nota:** El secreto compartido sólo es aplicable a las credenciales del remitente.

### Control de acceso

Cuando WebSphere Commerce recibe el mensaje OrderCreate, invoca el mandato BatchOrderRequest para procesar la petición de entrada. Para utilizar el servicio Web OrderCreate, debe existir un contrato entre el proveedor que está utilizando WebSphere Commerce y el comprador que actúa como cliente de servicio Web.

### Creación de usuarios y contratos

Si no hay ninguna organización compradora, siga los pasos siguientes antes de llamar al servicio Web OrderCreate. Siguiendo estos pasos, cree una nueva organización compradora, asígnele el rol de Responsable de compras y cree un contrato entre la organización vendedora y la organización compradora:

1. Cree una nueva organización compradora en el negocio utilizando la Consola de administración. Para obtener más detalles, consulte la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.
2. Asigne los roles siguientes a esta organización compradora:
  - a. Responsable de compras
  - b. Administrador de responsables de compras
  - c. Director de compras
3. Cree un usuario para esta organización compradora desde la Consola de administración. El ID de usuario de este usuario debe ser el ID de usuario de grupo de la organización compradora y la contraseña debe ser el secreto compartido de la organización compradora. EL ID de usuario de grupo y el

secreto compartido forman parte de las credenciales de la organización compradora mencionadas en el apartado “Consideraciones de seguridad” en la página 13.

4. Asigne los roles siguientes al usuario que ha creado en el paso 3:
  - a. Responsable de compras
  - b. Administrador de responsables de compras
5. Cree una cuenta de negocio para esta organización compradora utilizando WebSphere Commerce Accelerator. Para obtener instrucciones sobre cómo crear una cuenta de negocio, consulte la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce. Cuando cree una cuenta de negocio, tenga en cuenta lo siguiente:
  - No seleccione el recuadro **Permitir a los clientes comprar bajo el contrato por omisión de la tienda** en la página Cliente.
  - No seleccione el recuadro **El número de orden de compra puede especificarse en el momento de hacer el pedido** en la página Orden de compra.
  - Habilite el **método de pago de línea de crédito** y especifique el **número de cuenta de línea de crédito** en la página Línea de crédito.
6. Cree un contrato para la cuenta de negocio que ha creado. Para obtener instrucciones sobre cómo crear un contrato para la cuenta de negocio, consulte la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce. Cuando cree el contrato, tenga en cuenta lo siguiente:
  - Pulse la pestaña **General**. Entre el Nombre del contrato, la Descripción y seleccione la duración que desea que tenga el contrato.
  - Pulse la pestaña **Cliente**. Seleccione la organización compradora que ha creado anteriormente.
  - Pulse la pestaña **Pagos**. Asegúrese de que el uso de **Línea de crédito** esté seleccionado.
7. Pulse **Someter** para activar el contrato.

Los pasos anteriores crean una cuenta para la organización compradora en WebSphere Commerce, y un contrato entre la organización proveedora (WebSphere Commerce) y la organización compradora. Se asignan los permisos de acceso necesarios a la organización compradora.

## Inserción de datos en las tablas de base de datos

Para ejecutar el mandato `BatchOrderRequest`, debe insertar datos en determinadas tablas ejecutando un script de base de datos.

Para actualizar la base de datos, haga lo siguiente:

1. Vaya al directorio siguiente:
  -  `dir_instal_WC\samples\webservices\bin`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\bin`
  -     `dir_instal_WC/samples/webservices/bin`

2. Localice el conjunto de archivos script correctos para su configuración, tal como se describe en la tabla siguiente:

Sistema operativo	Tipo de base de datos	Nombres de archivo script
<span>Windows</span> <span>Studio</span>	<span>DB2</span>	Webservices_DBUpdate.db2.bat Webservices_UndoDBupdate.db2.bat
	<span>Oracle</span>	Webservices_DBUpdate.oracle.bat Webservices_UndoDBupdate.oracle.bat
<span>AIX</span> <span>Solaris</span>	<span>DB2</span>	Webservices_DBUpdate.db2.sh Webservices_UndoDBupdate.db2.sh
	<span>Oracle</span>	Webservices_DBUpdate.oracle.sh Webservices_UndoDBupdate.oracle.sh
<span>400</span> <span>Linux</span>	<span>DB2</span>	Webservices_DBUpdate.db2.sh Webservices_UndoDBupdate.db2.sh

+  
+  
+  
+  
+

3. Copie los archivos script adecuados en el siguiente directorio:

- Windows *dir\_instal\_\samples\webservices\bin*
- Studio *dir\_instal\_WCStudio\Commerce\samples\webservices\bin*
- AIX Linux Solaris *dir\_instal\_WC/samples/webservices/bin*
- 400 */QIBM/ProdData/CommerceServer55/samples/webservices/bin*

4. DB2 Copie los scripts *wcs.webservices.enable.sql* y *wcs.webservices.disable.sql* del siguiente directorio:

- Windows *dir\_instal\_WC\samples\webservices\schema\db2*
- Studio *dir\_instal\_WCStudio\Commerce\samples\webservices\schema\db2*
- AIX Linux Solaris *dir\_instal\_WC/samples/webservices/schema/db2*
- 400 */QIBM/ProdData/CommerceServer55/samples/webservices/schema/db2*

en el siguiente directorio:

- Windows *dir\_instal\_WC\schema\db2*
- Studio *dir\_instal\_WCStudio\Commerce\schema\db2*
- AIX Linux Solaris *dir\_instal\_WC/schema/db2*
- 400 */QIBM/UserData/CommerceServer55/*  
*WebSphere\_Commerce\_nombre\_instancia/schema*

Oracle Copie los scripts *wcs.webservices.enable.sql* y *wcs.webservices.disable.sql* del siguiente directorio:

- Windows *dir\_instal\_WC\samples\webservices\schema\oracle*
- Studio *dir\_instal\_WCStudio\Commerce\samples\webservices\*  
*schema\oracle*

-   `dir_instal_WC/samples/webservices/schema.oracle`

en el siguiente directorio:

-  `dir_instal_WC\schema\oracle`
-  `dir_instal_WCStudio\Commerce\schema\oracle`
-   `dir_instal_WC/schema/oracle`

5. Modifique los siguientes valores por omisión, utilizados en los script SQL `wcs.webservices.enable.sql` y `wcs.webservices.disable.sql` que ha copiado en el paso 4, para adaptarse a los detalles de la organización compradora que invoca el servicio Web OrderCreate:
  - `WebServicesUserId` - el código de organización de la organización compradora. Este código, junto con el tipo de código de organización, identifica de forma exclusiva la organización compradora.
  - `Webservices@ibm.com` - el tipo de código de organización de la organización compradora.
  - `aurora` - el código de organización de la organización proveedora. Normalmente, se trata del número D-U-N-S de la organización o de cualquier otro identificador exclusivo estándar de la organización. (El número D-U-N-S es un número de nueve dígitos que se utiliza como estándar global en la industria para la identificación y seguimiento de negocios).
  - `DUNS` - el tipo de código de organización de la organización proveedora. Use "DUNS" si utiliza el número DUNS del código de organización.
6. Una vez actualizados los scripts SQL, ejecute el archivo script de mandatos adecuado:

 **DB2** Si está utilizando una base de datos DB2:

-   Desde un ventana de mandatos de DB2, ejecute `Webservices_DBUpdate.db2.bat`
-    Desde una ventana de mandatos, ejecute `Webservices_DBUpdate.db2.sh`
-  Inicie una sesión QSH. Ejecute el mandato `/QIBM/ProdData/CommerceServer55/samples/webservices/bin/Webservices_DBUpdate.db2.sh`

 **Oracle** Si está utilizando una base de datos Oracle:

-   Desde una ventana de mandatos, ejecute `Webservices_DBUpdate.oracle.bat`
-   Desde una ventana de mandatos, ejecute `Webservices_DBUpdate.oracle.sh`

Verifique el archivo de anotaciones `dir_instal_WC\logs\Webservices_DBUpdate.log` y asegúrese de que la ejecución finaliza satisfactoriamente.

**Nota:** Si desea habilitar el servicio Web OrderCreate únicamente, deberá comentar las líneas del script SQL que registran el mandato de tarea `ExtOrderProcessServiceCmdImpl` y `WSDLFinderCmdImpl` para el servicio Web OrderFulfillment.

## Nuevas correlaciones de mensajes

Para habilitar el servicio Web OrderCreate, se proporciona un nuevo conjunto de correlaciones de ejemplo con el archivo `webservice_template.xml`. A continuación se muestra el documento plantilla para la correlación del mensaje:

```
<!-- correlación para mensaje orderCreate de entrada -->
<TemplateDocument>
  <DocumentType version='1.0'>ns1:orderCreate</DocumentType>
  <StartElement>ns1:orderCreate</StartElement>
  <TemplateTagName>BatchOrderCreate10Map</TemplateTagName>
  <CommandMapping>
    <Command CommandName='BatchOrderRequest'>
      <Constant Field='protocolName'>SOAP</Constant>
      <Constant Field='protocolVersion'>1.1</Constant>
    </Command>
  </CommandMapping>
</TemplateDocument>
```

La correspondiente correlación del mensaje de este servicio se denomina `BatchOrderCreate10Map` y está disponible en el archivo de plantilla `webservice_template.xml`, en el siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\messaging`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
-    `dir_instal_WC/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`

La correlación del mensaje que se proporciona corresponde a la definición WSDL del siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\messaging\OrderCreate.wsdl`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging\OrderCreate.wsdl`
-    `dir_instal_WC/samples/webservices/xml/messaging/OrderCreate.wsdl`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging/OrderCreate.wsdl`

Funciona con cualquier mensaje de petición SOAP que envíe un cliente de servicio Web y que se genere a partir de la definición WSDL proporcionada.

La definición de una correlación del mensaje requiere conocer la información de XPATH de la petición SOAP. Puede obtener la información de XPATH de la definición WSDL. Alternativamente, quizá sea más adecuado examinar el mensaje SOAP. La sección siguiente describe cómo acceder al mensaje SOAP y definir una plantilla de correlación.

Para definir una correlación del mensaje similar a la que corresponde al mensaje `OrderCreate` (`BatchOrderCreate10Map`) de la petición SOAP correspondiente, haga lo siguiente:

1. Utilice la definición WSDL para generar el proxy de cliente. Consulte los pasos descritos en el “Apéndice B. Utilización de IBM WebSphere Studio Application Developer” en la página 47.

2. Utilice un programa de utilidad de intercepción de nivel de transporte para ver qué aspecto tienen los documentos de la petición SOAP y la respuesta SOAP. Por ejemplo, puede utilizar el Supervisor TCP disponible en WebSphere Studio Application Developer 5.0, o cualquier otro programa de utilidad de encapsulación HTTP que pueda anotar el mensaje. Permita que el proxy de cliente apunte al interceptor estableciendo el punto de acceso. Ver la petición SOAP le ayudará a escribir las correlaciones de la petición y a escribir el archivo JSP para generar la respuesta.
3. Defina el elemento `TemplateDocument` del archivo de definición de plantilla. Este contiene los detalles del mandato que se ejecutará cuando se reciba el mensaje XML SOAP. Para obtener más información, consulte la sección titulada "Elemento `TemplateDocument` de un archivo de definición de plantilla" en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.
4. Utilice la petición SOAP obtenida en el paso 2 para definir el elemento `TemplateTag` que contiene una lista de definiciones de códigos. Asegúrese de que `TemplateTag` contiene correlaciones para todos los elementos de la definición WSDL. Para ello, deberá insertar datos para todos los elementos de la petición SOAP cuando envíe la petición desde el cliente. Cada definición de código correlaciona un elemento o atributo del mensaje XML SOAP de entrada con un parámetro del mandato. Consulte la sección titulada "Elemento `TemplateTag` de un archivo de definición de plantilla" en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

### **Configuración de correlacionadores de mensajes**

El correlacionador de mensajes SOAP utiliza el archivo de definición de plantilla de servicio Web `webservice_template.xml` para correlacionar la petición SOAP de entrada con las propiedades del mandato. Como resultado, puede configurar el correlacionador de mensajes SOAP modificando esta plantilla.

Si tiene una definición WSDL distinta, o si desea cambiar la definición del servicio, deberá modificar la correlación en la plantilla. Las correlaciones constan de las definiciones de XPATH del mensaje XML correlacionadas con los parámetros del mandato. En este caso, debe cambiar la estructura de XPATH para adaptarse al esquema del tipo de datos de acuerdo con la nueva definición del servicio.

## **Servicio OrderStatus**

Esta sección muestra cómo habilitar una lógica de negocio en WebSphere Commerce que actualice el estado de un pedido que se ha cursado en WebSphere Commerce como un servicio Web. Utilizando el servicio Web `OrderStatus`, puede invocar el mandato `OrderStatus` de WebSphere Commerce mediante una llamada de servicio Web. El mandato de controlador `OrderStatus` permite que un sistema externo actualice información relacionada con el estado de un pedido existente en WebSphere Commerce. Para utilizar el servicio Web `OrderStatus` de ejemplo debe tener un pedido existente, lo cual requiere que haya una tienda publicada dentro del sistema WebSphere Commerce.

El ejemplo que se proporciona muestra cómo habilitar el mandato `OrderStatus` de WebSphere Commerce para peticiones de servicio Web.

### **Habilitación del servicio Web OrderStatus de ejemplo**

A continuación se ofrece una visión general de los pasos que implementan el servicio Web `OrderStatus` de ejemplo:

1. Identifique la lógica de negocio que desea exponer como servicio Web. En este caso, permitir que una aplicación que despacha un pedido actualice el estado del pedido en WebSphere Commerce.

2. Identifique el mandato que implementa esta lógica de negocio. En este caso, se trata del mandato `OrderStatusService` de ejemplo. Amplía el mandato `OrderStatus` existente.
3. Identifique los parámetros obligatorios y opcionales que este mandato necesita. Esta información está disponible en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.
4. Cree la definición WSDL para el servicio Web `OrderStatus`. El archivo de definición `OrderStatus.wsdl` de ejemplo está en el siguiente directorio:
  -  `dir_instal_WC\samples\webservices\xml\messaging`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
  -    `dir_instal_WC/samples/webservices/xml/messaging`
  -  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`
5. Si ha habilitado el servicio `OrderCreate`, no ejecute este paso. En caso contrario, correlacione la petición SOAP de entrada con los parámetros del mandato, utilizando para ello el archivo de plantilla de correlación. La plantilla de correlación de ejemplo está en el siguiente directorio:
  -  `dir_instal_WC\samples\webservices\xml\messaging\webservice_template.xml`
  -  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging\webservice_template.xml`
  -    `dir_instal_WC/samples/webservices/xml/messaging\webservice_template.xml`
  -  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging\webservice_template.xml`
  - a. Copie el archivo `webservice_template.xml` en el siguiente directorio:
    -  `dir_instal_WC\xml\messaging`
    -  `dir_instal_WCStudio\Commerce\xml\messaging`
    -    `dir_instal_WC/xml/messaging`
    -  `/QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml/messaging`
  - b. Si el archivo `webservice_template.xml` ya existe y no contiene ninguna información, sobrescríbalo. Si el archivo contiene información, copie la información de `TemplateDocument` correspondiente al servicio `OrderStatus` en el archivo ya existente.
6. Si ha habilitado el servicio `OrderCreate`, no ejecute este paso. En caso contrario, habilite el correlacionador de mensajes para "SOAPINTEGRATION". Para hacerlo, abra el archivo de configuración `nombre_instancia.xml` situado en el siguiente directorio:
  -  `dir_instal_WC\instances\nombre_instancia\xml`
  -  `dir_instal_WCStudio\Commerce\instances\nombre_instancia\xml`
  -    `dir_instal_WC/instances/nombre_instancia/xml`
  -  `/QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml`

Localice el correlacionador de mensajes cuyo nombre es "WCS.SOAPINTEGRATION". Si el valor del atributo "enable" es "false", establézcalo en "true".

7. Escriba las plantillas JSP para componer una respuesta. El archivo JSP de ejemplo SOAP\_OrderStatusResponse.jsp está disponible para crear una respuesta.

a. Copie el archivo SOAP\_OrderStatusResponse.jsp del siguiente directorio:

-  *dir\_instal\_WC*\samples\webservices\wcstores
-  *dir\_instal\_WCStudio*\Commerce\samples\webservices\wcstores
-    *dir\_instal\_WC*/samples/webservices/wcstores
-  *dir\_instal\_WC*/samples/webservices/wcstores

en el siguiente directorio:

-  *dir\_instal\_WAS*\installedApps\*nombre\_casilla*\WC\_*nombre\_instancia*.ear\Stores.war
-  *dir\_instal\_WCStudio*\Commerce\IBM WebSphere Test Environment\hosts\default\_host\default\_app\web
-    *dir\_instal\_WAS*/installedApps/*nombre\_casilla*/WC\_*nombre\_instancia*.ear/Stores.war
-  *dir\_usuario\_WAS*/installedApps/*nombre\_casilla*/WC\_*nombre\_instancia*.ear/Stores.war

Esta plantilla JSP crea el mensaje de respuesta SOAP cuando el estado del pedido se actualiza satisfactoriamente en WebSphere Commerce.

b. Si ha habilitado el servicio OrderCreate, no es necesario que ejecute este paso. En caso contrario, copie los archivos SOAP\_GenericApplicationError.jsp y SOAP\_GenericSystemError.jsp del siguiente directorio:

-  *dir\_instal\_WC*\samples\webservices\wcstores
-  *dir\_instal\_WCStudio*\Commerce\samples\webservices\wcstores
-    *dir\_instal\_WC*/samples/webservices/wcstores
-  /QIBM/ProdData/CommerceServer55/samples/webservices/wcstores

en el siguiente directorio:

-  *dir\_instal\_WAS*\installedApps\*nombre\_casilla*\WC\_*nombre\_instancia*.ear\Stores.war
-  *dir\_espacio\_trabajo\_WCStudio*\Stores\Web Content
-    *dir\_instal\_WAS*/installedApps/*nombre\_casilla*/WC\_*nombre\_instancia*.ear/Stores.war
-  *dir\_usuario\_WAS*/installedApps/*nombre\_casilla*/WC\_*nombre\_instancia*.ear/Stores.war

Estas plantilla JSP crean los mensajes de error SOAP cuando se produce un error de una aplicación o un error del sistema.

8. Si ha habilitado el servicio OrderCreate, no es necesario que ejecute este paso. En caso contrario, haga lo siguiente:

a. Localice el conjunto de archivos script correctos para su configuración, tal como se describe en la tabla siguiente:

Sistema operativo	Tipo de base de datos	Nombres de archivo script
Windows Studio	DB2	Webservices_DBUpdate.db2.bat Webservices_UndoDBUpdate.db2.bat
	Oracle	Webservices_DBUpdate.oracle.bat Webservices_UndoDBUpdate.oracle.bat
AIX Solaris	DB2	Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh
	Oracle	Webservices_DBUpdate.oracle.sh Webservices_UndoDBUpdate.oracle.sh
400 Linux	DB2	Webservices_DBUpdate.db2.sh Webservices_UndoDBUpdate.db2.sh

+  
+  
+  
+  
+

b. Copie los archivos script adecuados en el siguiente directorio:

- Windows `dir_instal_WC\samples\webservices\bin`
- Studio `dir_instal_WCStudio\Commerce\samples\webservices\bin`
- AIX Linux Solaris `dir_instal_WC/samples/webservices/bin`
- 400 `/QIBM/ProdData/CommerceServer55/samples/webservices/bin`

c. DB2 Copie los scripts `wcs.webservices.enable.sql` y `wcs.webservices.disable.sql` del siguiente directorio:

- Windows `dir_instal_WC\samples\webservices\schema\db2`
- Studio `dir_instal_WCStudio\Commerce\samples\webservices\schema\db2`
- AIX Linux Solaris `dir_instal_WC/samples/webservices/schema/db2`
- 400 `/QIBM/ProdData/CommerceServer55/samples/webservices/schema/db2`

en el siguiente directorio:

- Windows `dir_instal_WC\schema\db2`
- Studio `dir_instal_WCStudio\Commerce\schema\db2`
- AIX Linux Solaris `dir_instal_WC/schema/db2`
- 400 `/QIBM/ProdData/CommerceServer55/schema/db2`

► **Oracle** Copie los scripts `wcs.webservices.enable.sql` y `wcs.webservices.disable.sql` del siguiente directorio:

- ► **Windows** `dir_instal_WC\samples\webservices\schema\oracle`
  - ► **Studio** `dir_instal_WCStudio\Commerce\samples\webservices\schema\oracle`
- ► **AIX** ► **Solaris** `dir_instal_WC/samples/webservices/schema/oracle`

en el siguiente directorio:

- ► **Windows** `dir_instal_WC\schema\oracle`
    - ► **Studio** `dir_instal_WCStudio\Commerce\schema\oracle`
  - ► **AIX** ► **Solaris** `dir_instal_WC/schema/oracle`
- d. Actualice los archivos script SQL que ha copiado en el paso 7c comentando las sentencias del script que no estén relacionadas con el servicio Web OrderStatus.
- e. Ejecute los archivos script que ha copiado en el paso 7b.
9. Reinicie la instancia de Commerce Server para que los cambios entren en vigor si está utilizando una máquina de WebSphere Commerce. Si está utilizando una máquina de WebSphere Commerce Studio, reinicie el servidor WebSphereCommerceServer en el Entorno de prueba de WebSphere.

### Control de acceso para el servicio OrderStatus

Para el servicio Web OrderStatus se utiliza el modelo de control de acceso de WebSphere Commerce. Este modelo requiere pasar un ID de usuario y una contraseña a la petición de servicio Web, de manera que WebSphere Commerce pueda autenticar al peticionario y verificar que tiene la autorización necesaria para realizar la acción requerida. El ID de usuario y la contraseña están definidos en la definición WSDL de la siguiente manera:

El formato del esquema XML para las credenciales debe tener el aspecto siguiente:

```
<complexType
name="com.ibm.commerce.webservice.datatype.Credentials">
<all>
<element name="password" nillable="true" type="string"/>
<element name="logonId" nillable="true" type="string"/>
</all>
</complexType>
```

La información de credenciales debe correlacionarse con los siguientes campos en la definición de plantilla:

- El nombre del campo para la conexión del usuario es 'logonId' y FieldInfo es 'CONTROL'.
- El nombre del campo para la contraseña es 'logonPassword' y FieldInfo es 'CONTROL'.

Por ejemplo, para el servicio OrderStatus, el LogonID y la contraseña se correlacionan con el mandato OrderStatus utilizando la siguiente sección de la plantilla de correlación:

```
<TemplateTag name='OrderStatus20Map'>
<!--Entorno de ejecución -->
<Tag XPath='updateOrderStatusMessage/controlArea/credentials/logonId'
```

```

Field='logonId' FieldInfo='CONTROL' />
<Tag XPath='updateOrderStatusMessage/controlArea/credentials/password'
Field='logonPassword' FieldInfo='CONTROL' />

```

Este ejemplo proporciona un método típico que se ajusta a los requisitos de control de acceso de los servicios Web definidos para WebSphere Commerce.

## Roles de usuario para el servicio Web OrderStatus

El servicio OrderStatus utiliza los mandatos de controlador OrderStatusServiceCmd y OrderStatusCmd. Por omisión, estos mandatos de controlador se añaden como recursos al grupo de recursos SellersCmdResourceGroup, y como acciones al grupo de acciones BackendOrderStatusCreate. Como resultado, únicamente los usuarios a los que se ha asignado el rol de 'vendedor' pueden ejecutar los mandatos OrderStatusCmd y OrderStatusServiceCmd. Esto significa que el usuario cuyas credenciales se pasan como parte del mensaje OrderStatus debe tener el rol de 'vendedor'. Por omisión, cualquier usuario que tenga autorización de administrador de WebSphere Commerce puede ejecutar cualquiera de estos mandatos. En consecuencia, esto sólo es aplicable cuando se desea habilitar el servicio OrderStatus para un usuario que no tiene autorización de administrador.

## La definición WDSL de OrderStatus

El servicio Web OrderStatus se basa en la definición WSDL de OrderStatus disponible en el siguiente directorio:

-  *dir\_instal\_WC*\samples\webservices\xml\messaging
-  *dir\_instal\_WCStudio*\Commerce\samples\webservices\xml\messaging
-  *dir\_instal\_WC*/samples/webservices/xml/messaging
-   *dir\_instal\_WC*/samples/webservices/xml/messaging
-  /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging

## Nuevas correlaciones de mensajes

Para habilitar el servicio Web OrderStatus, se proporciona un nuevo conjunto de correlaciones en el archivo de plantilla de correlación webservice\_template.xml. A continuación se muestra el documento plantilla para la correlación del mensaje:

```

<!-- correlación para mensaje update_WCS_OrderStatus de entrada -->
<TemplateDocument>
  <DocumentType version='1.0'>ns1:update_WCS_OrderStatus
</DocumentType>
  <StartElement>ns1:update_WCS_OrderStatus</StartElement>
  <TemplateName>OrderStatus20Map</TemplateName>
  <CommandMapping>
    <Command CommandName='OrderStatusService'>
      Condition='OrderStatusCommand'>
        <Constant Field='redirecturl'>
          SOAP_OrderStatusResponse
        </Constant>
        <Constant Field='SOAP_OrderStatusResponse'>
          SOAP_OrderStatusResponse.jsp
        </Constant>
      </Command>
    </CommandMapping>
  </TemplateDocument>

```

La correspondiente correlación del mensaje de este servicio se denomina OrderStatus20Map y está disponible en el archivo de plantilla webservice\_template.xml, en el siguiente directorio:

-  *dir\_instal\_WC*\samples\webservices\xml\messaging

-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
-    `dir_instal_WC/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/samples/`  
`webservices/xml/messaging`

La correlación del mensaje que se proporciona corresponde a la definición WSDL descrita en el siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\`  
`messaging\Orderstatus.wsdl`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\`  
`messaging\Orderstatus.wsdl`
-    `dir_instal_WC/samples/webservices/xml/`  
`messaging/Orderstatus.wsdl`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/`  
`xml/messaging/OrderStatus.wsdl`

La correlación del mensaje funciona con cualquier mensaje que envíe un cliente de servicio Web y que se genere a partir de la definición WSDL proporcionada.

La definición de una correlación del mensaje requiere conocer la información de XPATH de la petición SOAP. Puede obtener la información de XPATH de la definición WSDL. Alternativamente, quizá sea más adecuado examinar el mensaje SOAP. La sección siguiente describe cómo acceder al mensaje SOAP y definir una plantilla de correlación.

Para definir una correlación del mensaje similar a la que corresponde al mensaje OrderStatus (OrderStatus20Map) de la petición SOAP correspondiente, haga lo siguiente:

1. Utilice el archivo de definición WSDL para generar el proxy de cliente. Consulte los pasos descritos en el “Apéndice B. Utilización de IBM WebSphere Studio Application Developer” en la página 47 para obtener más información.
2. Utilice un programa de utilidad de intercepción de nivel de transporte para ver qué aspecto tienen los documentos de la petición SOAP y la respuesta SOAP. Por ejemplo, puede utilizar el Supervisor TCP disponible en la distribución AXIS de Apache, o cualquier otro programa de utilidad de encapsulación HTTP que pueda anotar el mensaje. Permita que el proxy de cliente apunte al interceptor estableciendo el punto de acceso. Esto le ayudará a escribir las correlaciones de la petición y a escribir el archivo JSP para generar la respuesta.
3. Defina el elemento `TemplateDocument` del archivo de definición de plantilla. Este contiene los detalles del mandato que se ejecutará cuando se reciba un mensaje XML. Para obtener más información, consulte la sección titulada “Elemento `TemplateDocument` de un archivo de definición de plantilla” en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.
4. Defina el elemento `TemplateTag` que contiene una lista de definiciones de códigos. Asegúrese de que `TemplateTag` contiene correlaciones para todos los elementos de la definición WSDL. Para ello, deberá insertar datos para todos los elementos de la petición SOAP cuando envíe la petición desde el cliente. Cada definición de código correlaciona un elemento o atributo del mensaje XML de entrada con un parámetro del mandato. Consulte la sección titulada “Elemento `TemplateTag` de un archivo de definición de plantilla” en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

## Configuración de correlacionadores de mensajes

El correlacionador de mensajes SOAP utiliza el archivo de definición de plantilla de servicio Web (`webservice_template.xml`) para correlacionar la petición SOAP de entrada con las propiedades del mandato. Como resultado, puede configurar el correlacionador de mensajes SOAP modificando esta plantilla. Para obtener más información sobre los archivos de definición de plantilla para mensajes de entrada, consulte la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

Si tiene una definición WSDL distinta, o si desea cambiar la definición del servicio, deberá modificar la correlación en la plantilla. Las correlaciones constan de las definiciones de XPATH del mensaje XML correlacionadas con los parámetros del mandato. En este caso, debe cambiar la estructura de XPATH para adaptarse al esquema del tipo de datos de acuerdo con la nueva definición del servicio.

---

## Inhabilitación de los servicios Web OrderCreate y OrderStatus

Esta sección explica cómo inhabilitar los servicios Web OrderCreate y OrderStatus que proporciona WebSphere Commerce.

1. Suprima los siguientes archivos JSP:

- `SOAP_PurchaseOrderResponse.jsp`
- `SOAP_GenericApplicationError.jsp`
- `SOAP_GenericSystemError.jsp`
- `SOAP_OrderStatusResponse.jsp`

del siguiente directorio:

- `Windows dir_instal_WAS\installedApps\nombre_casilla\WC_nombre_instancia.ear\Stores.war`
- `Studio dir_espacio_trabajo_WCStudio\Stores\Web Content`
- `AIX Linux Solaris dir_instal_WAS/installedApps/nombre_casilla/WC_nombre_instancia.ear/Stores.war`
- `400 dir_usuario_WAS/installedApps/nombre_casilla/WC_nombre_instancia.ear/Stores.war`

2. Sustituya el archivo `webservice_template.xml` del siguiente directorio:

- `Windows dir_instal_WC\xml\messaging`
- `Studio dir_instal_WCStudio\Commerce\xml\messaging`
- `AIX Linux Solaris dir_instal_WC/xml/messaging`
- `400 /QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml/messaging`

por una plantilla vacía que contenga un nodo `ECTemplate` vacío, tal como se muestra en el ejemplo siguiente:

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE ECTemplate SYSTEM 'ec_template.dtd'>
<ECTemplate>
</ECTemplate>
```

3. Localice los scripts `wcs.webservices.disable.sql` en el siguiente directorio:

`DB2` Si está utilizando una base de datos DB2:

- `Windows dir_instal_WC\schema\db2`

-  `dir_instal_WCStudio\Commerce\schemadb2`
-    `dir_instal_WC/schema/db2`
-  `QIBM/UserData/CommerceServer55/instances/  
nombre_instancia/schema`

 Si está utilizando una base de datos Oracle:

-  `dir_instal_WC\schemoracle`
-  `dir_instal_WCStudio\Commerce\schemoracle`
-   `dir_instal_WC/schema/oracle`

- Con un editor de texto, abra el script `wcs.webservices.disable.sql` mencionado en el paso 3 y asegúrese de que los valores siguientes sean los mismos que los del script `wcs.webservices.enable.sql`:
  - `id_entidad_tienda` - el identificador exclusivo de su tienda.
  - `OrgCode1` - la clave primaria de la entrada de la tabla ORGCODE correspondiente a la organización proveedora.
  - `OrgCode2` - la clave primaria de la entrada de la tabla ORGCODE correspondiente a la organización compradora.
  - `MemberId` - la clave primaria del grupo de miembros al que pertenece la organización compradora.
- Después de completar los cambios mencionados en el paso 4, ejecute el archivo script de mandatos adecuado:

 Si está utilizando una base de datos DB2:

-   Desde un ventana de mandatos de DB2, ejecute `WebServices_UndoDBUpdate.db2.bat`
-    Desde una ventana de mandatos, ejecute `WebServices_UndoDBUpdate.db2.sh`
-  Inicie una sesión QSH. Ejecute el mandato `/QIBM/ProdData/CommerceServer55/samples/webservices/bin/  
WebServices_UndoDBUpdate.db2.sh`

 Si está utilizando una base de datos Oracle:

-   Desde un ventana de mandatos de DB2, ejecute `WebServices_UndoDBUpdate.oracle.bat`
-   Desde una ventana de mandatos, ejecute `WebServices_UndoDBUpdate.oracle.sh`

Verifique el archivo de anotaciones y asegúrese de que la ejecución finaliza satisfactoriamente. El nombre y la ubicación del archivo de anotaciones se muestra después de que finalice la ejecución del archivo script.

**Nota:** Si sólo desea inhabilitar los servicios Web OrderCreate y OrderStatus, o si no ha habilitado el servicio Web OrderFulfillment, comente las siguientes sentencias SQL de los archivos script listados en el paso 4:

- `delete from cmdreg where interfacename='com.ibm.commerce.webservice.utils.WSDLFinderCmd' and classname='com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl';`
- `delete from cmdreg WHERE storeent_id='id_entidad_tienda' AND interfacename='com.ibm.commerce.order.commands.ExtOrderProcessCmd';`

donde *id\_entidad\_tienda* es el identificador exclusivo de su tienda.

6.  Suprima el código de ejemplo que ha importado en el apartado “Importación del código de ejemplo” en la página 15 y vuelva a construir el proyecto `WebSphereCommerceServerExtensionsLogic`.
7. Reinicie la instancia de Commerce Server para que los cambios entren en vigor si está utilizando una máquina de WebSphere Commerce. Si está utilizando una máquina de WebSphere Commerce Studio, reinicie el servidor `WebSphereCommerceServer` en el Entorno de prueba de WebSphere.



---

## Capítulo 4. Habilitación de WebSphere Commerce como peticionario de servicios

Esta sección describe cómo puede habilitar WebSphere Commerce para invocar un servicio Web definido por un sistema externo. Describe el ejemplo que se proporciona, el cual habilita WebSphere Commerce para buscar e invocar servicios Web. Además, se ofrecen detalles de la implementación del servidor Web OrderFulfillment de ejemplo.

---

### Arquitectura

El flujo de mensajes en esta arquitectura es síncrono. Esto significa que WebSphere Commerce espera una respuesta del sistema externo antes de continuar procesando. El mandato de tarea se implementa para que interactúe con un sistema externo mediante la llamada a un servicio Web.

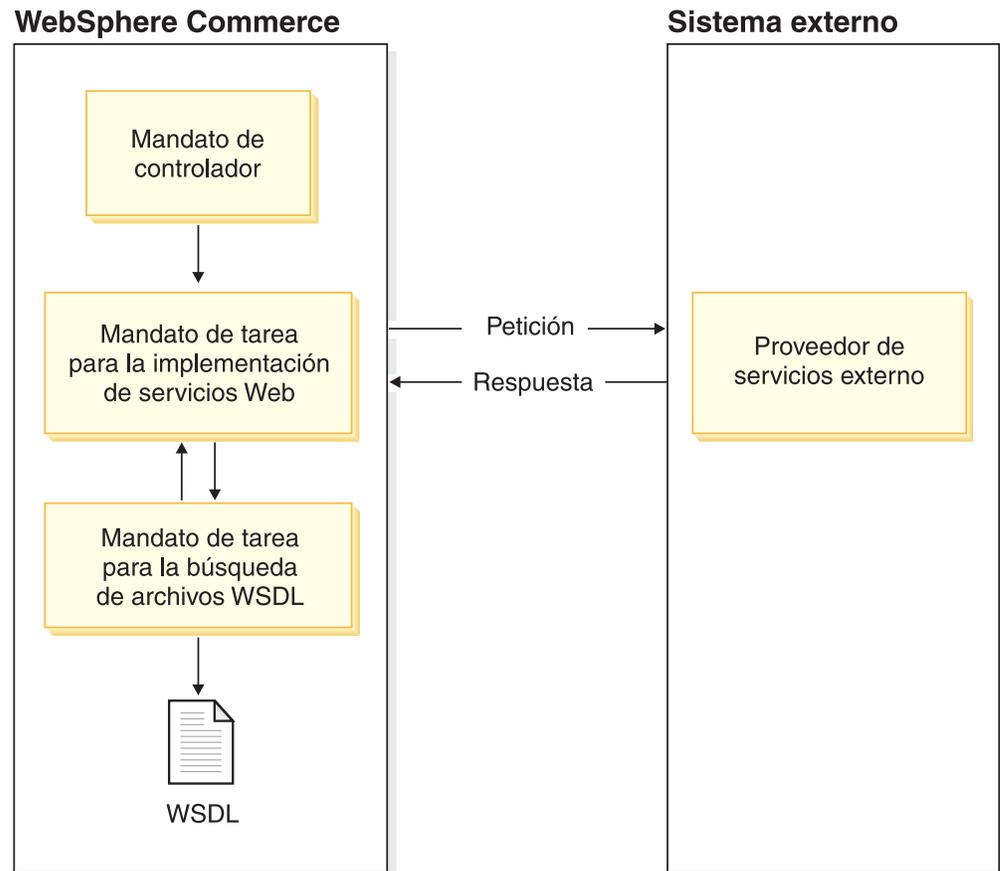


Figura 5. Arquitectura que habilita WebSphere Commerce como peticionario de servicios

El modelo de programación de WebSphere Commerce le permite personalizar el comportamiento de un proceso de negocio ya existente alterando los mandatos. Un mandato de controlador invoca muchos mandatos de tarea para completar la ejecución de un proceso de negocio. Una manera de modificar el comportamiento de un mandato de controlador es escribir una nueva implementación de un mandato de tarea que el mandato de controlador invoca.

Para averiguar qué mandatos de tarea se invocan como parte de un mandato de controlador, consulte la sección titulada "Referencia cruzada de mandatos, tareas y tablas" en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce. La lista siguiente describe la interacción que se produce cuando un proceso de negocio de WebSphere Commerce invoca un servicio Web alojado en un sistema externo:

1. El mandato de controlador invoca el mandato de tarea registrado. En este caso, la tarea se implementa para que interactúe con un sistema externo mediante la llamada a un servicio Web. El mandato de tarea contiene el código de integración con un sistema externo.
2. El mandato de tarea recupera el punto de acceso del servicio Web especificado.
3. El mandato de tarea invoca un proxy de servicio Web especificando el punto de acceso del servicio y sus parámetros de entrada.
4. El proxy llama al servicio Web, recibe la respuesta y la devuelve al mandato de tarea.

## Localización de una implementación de servicio

Para poder habilitar WebSphere Commerce para invocar un servicio Web definido por un sistema externo, primero debe localizar un proveedor de servicios que aloje la implementación del tipo de servicio requerido. Para invocar un servicio Web, puede hacer que el mandato de tarea apunte al proveedor de servicios que ha identificado. Para mayor flexibilidad, se incluye un mandato de tarea para localizar un proveedor de servicios, definido en el mandato de tarea `WSDLFinderCmd`. En el siguiente directorio se suministra una implementación de ejemplo para este mandato de tarea:

-  `dir_instal_WC\samples\webservices\com\ibm\commerce\webservice\utils`
-  `dir_instal_WCStudio\Commerce\samples\webservices\com\ibm\commerce\webservice\utils`
-    `dir_instal_WC/samples/webservices/com/ibm/commerce/webservice/utils`
-  `/QIBM/ProdData/CommerceServer55/samples/webservices/com/ibm/commerce/webservice/utils`

Una manera de buscar una implementación de servicio es examinar un registro de servicios. No obstante, el ejemplo proporcionado con WebSphere Commerce Versión 5.5 utiliza el sistema de archivos local para recuperar la definición WSDL del servicio Web. `WSDLFinderCmdImpl` busca los archivos de definiciones WSDL disponibles localmente en el archivo de configuración `nombre_instancia.xml`. Para registrar una definición de servicio WSDL, edite el archivo de configuración `nombre_instancia.xml` situado en el siguiente directorio:

-  `dir_instal_WC\instances\nombre_instancia.xml`
-  `dir_instal_WCStudio\Commerce\instances\nombre_instancia.xml`
-    `dir_instal_WC/instances/nombre_instancia.xml`
-  `/QIBM/UserData/CommerceServer55/instances/nombre_instancia.xml`

y haga lo siguiente:

1. Localice el elemento `SOAPWSDLFinderInfo`. Este elemento tiene atributos para registrar la vía de acceso de la definición WSDL y el archivo WSDL. El siguiente extracto del archivo `nombre_instancia.xml` muestra estos parámetros:

```

<SOAPWSDLFinderInfo
LocalWSDLFiles="OrderFulfillment.wsdl"
LocalWSDLPath="dir_instal_WC/xml/messaging"
UDDIQueryURL=""
UDDIPublishURL=""
WSDLFinderImplClassName=" "
display="false" />

```

2. Para el valor de LocalWSDLFiles, incluya la lista de definiciones WSDL separadas por comas disponibles en el sistema de archivos local, y para el valor de LocalWSDLPath, especifique el directorio donde residen estos archivos.
3. Para los valores de UDDIQueryURL y UDDIPublishURL, especifique una serie vacía puesto que este ejemplo utiliza el sistema de archivos local para buscar una definición de servicio.
4. Para el valor de WSDLFinderImplClassName, especifique una serie vacía.

Para buscar dinámicamente una empresa asociada e invocar un servicio que dicha empresa ha implementado y publicado en el registro de servicios, debe crear y registrar una nueva implementación del mandato de tarea de ejemplo WSDLFinderCmdImpl que se proporciona con WebSphere Commerce Versión 5.5. En su implementación personalizada, incluya el código para buscar la definición WSDL. Aquí puede utilizar los campos UDDIQueryURL y UDDIPublishURL para registrar un registro UDDI. Para obtener más información sobre cómo alterar los mandatos de tarea, consulte la publicación *WebSphere Commerce, Guías de programación y aprendizaje*.

## Invocación de una implementación de servicio

Para averiguar qué mandatos de tarea se invocan como parte de un mandato de controlador, consulte la sección titulada "Referencia cruzada de mandatos, tareas y tablas" en la Ayuda en línea a la producción y el desarrollo de WebSphere Commerce.

La tabla siguiente muestra las tareas que se ejecutan como parte del mandato OrderProcessCmd:

Nombre del mandato de tarea	Descripción
CheckOrderCmd	El mandato de tarea CheckOrderCmd comprueba si se ha procesado un pedido cuyo estado es "sometido".
DoPaymentCmd	El mandato de controlador OrderProcessCmd llama al mandato de tarea DoPaymentCmd para procesar el pago de un pedido.
ExtOrderProcessCmd	Realiza cualquier proceso adicional necesario justo antes de la finalización del mandato OrderProcess.
UpdateInventoryCmd	El mandato de tarea UpdateInventory le permite actualizar los elementos del inventario.

En esta tabla queda claro que si desea personalizar el comportamiento del mandato OrderProcess para enviar una petición de despacho de pedido a un centro de despacho de pedidos externo, puede alterar el mandato de tarea ExtOrderProcessCmd. La nueva implementación de ExtOrderProcessCmd puede buscar una empresa asociada que haya implementado el servicio OrderFulfillment e invocar dicho servicio.

---

## Servicio Web de ejemplo que WebSphere Commerce puede invocar en un sistema externo

Esta sección describe cómo puede habilitar el servicio Web OrderFulfillment de ejemplo.

Puede habilitar el servicio OrderFulfillment de ejemplo en una máquina de WebSphere Commerce o en una máquina de desarrollo instalada con WebSphere Commerce Studio. Si está habilitando este ejemplo en una máquina de WebSphere Commerce Studio, siga las instrucciones que se proporcionan en esta sección y consulte las vías de acceso a directorios con el símbolo "  Studio ".

### Servicio OrderFulfillment

Esta sección explica cómo puede habilitar WebSphere Commerce para llamar a un servicio Web definido por un sistema externo. El servicio OrderFulfillment se utiliza para mostrar cómo funciona WebSphere Commerce como peticionario de servicios. Para utilizar el servicio Web OrderFulfillment de ejemplo, necesita tener una tienda publicada dentro del sistema WebSphere Commerce.

Si está habilitando este servicio Web de ejemplo en el entorno de WebSphere Commerce, también será necesario un entorno de WebSphere Commerce Studio.

#### Definiciones WSDL de interfaz

La lógica de negocio del servicio OrderFulfillment está diseñada para funcionar con una interfaz de servicio predefinida, que es el servicio OrderFulfillment. El archivo de definición WSDL de interfaz para el servicio OrderFulfillment está situado en el sistema de archivos local para su consulta y para realizar el enlace estático por omisión con un servicio conocido.

#### Mandato de tarea

El mandato ExtOrderProcessServiceCmdImpl que se invoca como parte de OrderProcessCmd implementa la interfaz ExtOrderProcessServiceCmd para interactuar con un sistema externo. Inicia una petición con el servicio Web a través del proxy. Este mandato de tarea procesa un pedido sometido enviando una petición OrderFulfillment al centro de despacho de pedidos correspondiente.

Estos son los pasos a seguir para habilitar el servicio OrderFulfillment:

1. Si ya ha habilitado WebSphere Commerce como proveedor de servicios, tal como se describe en el Capítulo 3, "Inserción de datos en las tablas de base de datos" en la página 20, no ejecute este paso. Sin embargo, si anteriormente ha comentado la sección que incluye los mandatos OrderFulfillment, debe ejecutar este paso.

Para registrar ExtOrderProcessServiceCmdImpl y WSDLFinderCmdImpl en el registro de mandatos de WebSphere Commerce, ejecute las sentencias SQL relacionadas con estos mandatos. El archivo script SQL está disponible en el siguiente directorio:

 DB2 Si está utilizando una base de datos DB2:

-  Windows `dir_instal_WC\samples\webservices\schema\db2\wcs.webservices.enable.sql`
-  Studio `dir_instal_WCStudio\Commerce\samples\webservices\schema\db2\wcs.webservices.enable.sql`
-  AIX  Linux  Solaris `dir_instal_WC/samples/webservices/schema/db2/wcs.webservices.enable.sql`

- `400 /QIBM/ProdData/CommerceServer55/samples/webservices/schema/db2/wcs.webservices.enable.sql`

► **Oracle** Si está utilizando una base de datos Oracle:

- ► **Windows** `dir_instal_WC\samples\webservices\schema\oracle\wcs.webservices.enable.sql`
- ► **Studio** `dir_instal_WCStudio\Commerce\samples\webservices\schema\oracle\wcs.webservices.enable.sql`
- ► **AIX** ► **Solaris** `dir_instal_WC/samples/webservices/schema/oracle/wcs.webservices.enable.sql`

A continuación se muestran las sentencias SQL del archivo script anterior que debe ejecutar:

```
insert into cmdreg (id_entidad_tienda, interfacename, description,
  classname, properties, lastupdate, target) values(0, 'com.ibm.commerce.
webservice.utils.WSDLFinderCmd', 'Wsd1 finder class for webservices',
com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl',
null, null, 'Local');
```

```
insert into cmdreg (id_entidad_tienda, interfacename, description,
  classname, properties, lastupdate, target) values(storeent_id,
'com.ibm.commerce.order.commands.ExtOrderProcessCmd', '
webservices implementation of ExtOrderProcessCmd',
'com.ibm.commerce.webservice.order.commands.ExtOrderProcessServiceCmdImpl',
null,null,'Local');
```

donde *id\_entidad\_tienda* es el identificador exclusivo de su tienda.

2. Para habilitar la búsqueda del servicio OrderFulfillment, haga lo siguiente:
  - a. Localice el elemento SOAPWSDLFinderInfo en el archivo de configuración *nombre\_instancia.xml*.
  - b. Asegúrese de que los valores de LocalWSDLFiles y LocalWSDLPath sean correctos y reflejen el directorio donde se encuentran los archivos de definición WSDL. El elemento SOAPWSDLFinderInfo debe tener el siguiente aspecto:

```
<SOAPWSDLFinderInfo
  LocalWSDLFiles="OrderFulfillment.wsdl"
  LocalWSDLPath="dir_instal_WC/xml/messaging"
  WSDLFinderImplClassName=""
  display="false" />
```

- c. Copie el archivo WSDL OrderFulfillment.wsdl de ejemplo situado en el siguiente directorio:

- ► **Windows** `dir_instal_WC\samples\webservices\xml\messaging`
- ► **Studio** `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
- ► **AIX** ► **Linux** ► **Solaris** `dir_instal_WC/samples/webservices/xml/messaging`
- `400 /QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`

en el siguiente directorio:

-  `dir_instal_WC\xml\messaging`
  -  `dir_instal_WCStudio\Commerce\xml\messaging`
  -    `dir_instal_WC/xml/messaging`
  -  `/QIBM/UserData/CommerceServer55/  
instances/nombre_instancia/xml/messaging`
3. Identifique un proveedor de servicios para los servicios definidos en el archivo `OrderFulfillment.wsdl`. Actualice el archivo `OrderFulfillment.wsdl` con la información del proveedor de servicios.
    - a. Abra el archivo `OrderFulfillment.wsdl` que ha copiado en el paso 2c.
    - b. Localice el nodo de definición de servicio (la sección que empieza con "service name"). Es parecida a la siguiente:
 

```
<service name=" POCreateService ">
<port binding="tns: POCreateBinding " name=" POCreatePort>
<soap:address location="http://localhost:8080/WebservicesForWCBEWeb/
servlet/rpcrouter"/>
</port>
</service>
```
    - c. Cambie el URL resaltado existente por el URL del servicio Web de destino.
    - d. Guarde y cierre el archivo.
  4. Si ha habilitado el servicio Web de ejemplo `OrderCreate` o `OrderStatus`, sátese los pasos 4 y 5. De lo contrario, siga con este paso. Para desplegar el código de ejemplo proporcionado para el servicio Web `OrderFulfillment` debe tener una máquina de desarrollo con `WebSphere Commerce Studio` instalado y configurado. Los pasos de la lista siguiente que requieren `WebSphere Studio Application Developer` están pensados para ser ejecutados en esta máquina de desarrollo.
  5. Si desea editar o modificar los servicios Web de ejemplo, debe importar el código del ejemplo en `WebSphere Commerce Studio`. Para importar este código, haga lo siguiente:
    - a. Abra el entorno de desarrollo de `WebSphere Commerce` desde **Inicio > Programas > IBM WebSphere Commerce Studio > Entorno de desarrollo de WebSphere Commerce**.
    - b. Vaya a la perspectiva `J2EE` y seleccione la vista `Navegador J2EE`.
    - c. Expanda el proyecto `WebSphereCommerceServerExtensionsLogic`. Con el botón derecho del ratón, pulse sobre el directorio `src` y seleccione **Importar**. Se abrirá el asistente de importación.
    - d. Seleccione **Sistema de archivos** y pulse **Siguiente**.
    - e. Pulse **Examinar** y localice el siguiente directorio:
 

```
dir_instal_WCStudio\Commerce\samples\webservices\com
```
    - f. Pulse **Seleccionar todo** y, a continuación, **Finalizar**.
    - g. Con el botón derecho del ratón, pulse sobre el proyecto `WebSphereCommerceServerExtensionsLogic` y seleccione **Propiedades**.
    - h. Pulse **Vía de construcción Java** y vaya a la pestaña **Bibliotecas**.
    - i. Pulse **Añadir variable**. Seleccione las variables `SOAPJAR` y `XERCES` y pulse **Aceptar**.
    - j. Pulse **Aceptar**.
    - k. Con el botón derecho del ratón, pulse sobre el proyecto `WebSphereCommerceServerExtensionsLogic` y seleccione **Reconstruir proyecto**.

6. Si está habilitando el servicio Web de ejemplo en un servidor WebSphere Commerce, debe desplegar este código en su servidor WebSphere Commerce de destino. Consulte la publicación *WebSphere Commerce, Guías de programación y aprendizaje* para obtener más información acerca de cómo desplegar mandatos nuevos.
7. Reinicie la instancia de Commerce Server para que los cambios entren en vigor si está utilizando una máquina de WebSphere Commerce. Si está utilizando una máquina de WebSphere Commerce Studio, reinicie el servidor WebSphereCommerceServer en el Entorno de prueba de WebSphere.

### Objetos de negocio

El mandato de tarea ExtOrderProcessServiceCmdImpl inserta datos en un conjunto de objetos de negocio que el proxy de servicio Web utiliza para construir el mensaje de petición XML SOAP. Este mensaje de petición refleja los elementos de la definición WSDL de OrderFulfillment. Con el servicio OrderFulfillment se proporciona un conjunto de objetos de negocio de ejemplo que se genera a partir de la definición WSDL de OrderFulfillment. Cuando WebSphere Commerce invoca servicios Web externos, puede generar el proxy de cliente y los objetos de negocio para dichos servicios utilizando IBM WebSphere Studio, tal como se describe en el "Apéndice B. Utilización de IBM WebSphere Studio Application Developer" en la página 47, o utilizando otras herramientas.

La tabla siguiente lista los paquetes y los objetos de negocio que utiliza el servicio OrderFulfillment:

Nombre del paquete	Nombre del objeto	Descripción
com.ibm.commerce.webservice.datatype	Address	Este paquete consta de todos los objetos que se utilizan en los distintos mensajes de servicio Web.
	ContactInfo	
	ContactPersonName	
	ControlArea	
	Credentials	
	Email	
	InvoiceInfo	
	ProductDimension	
	ProductMeasurement	
	ProductWeight	
	RequisitionerID	
	Telephone	
	UserData	
	UserDataField	

Nombre del paquete	Nombre del objeto	Descripción
com.ibm.commerce. webservice.datatype. orderfulfillment	BillToInfo	Este paquete incluye todos los objetos que contienen datos específicos para el mensaje XML OrderFulfillment.
	BuyOrgAccountingDetail	
	BuyOrgInfo	Report_NC_PurchaseOrder es el objeto wrapper padre que se envía al proxy y OrderFulfillmentStatus el objeto recibido como respuesta.
	DataArea	
	DateTimeReference	
	ItemShippingSchedule	
	ItemUnitPrice	
	MerchantInfo	
	MonetaryAmount	
	OrderFulfillmentStatus	
	PCardInfo	
	Report_NC_PurchaseOrder	
	ReportPO	
	ReportPOHeader	
	ReportPOItem	
	RequisitionerInfo	
	ServiceAllowanceCharge	
	ShipDateReference	
	ShippingCarrierInfo	
	ShipToInfo	
TaxInfo		
TotalPriceInfo		

### Prueba del ejemplo

Cuando se formaliza un pedido en una tienda publicada de WebSphere Commerce, se invoca el servicio Web OrderFulfillment. El servicio Web conecta con el proveedor de servicios externo para formalizar un pedido en el centro de despacho de pedidos. Si el proveedor de servicios devuelve el ID del pedido formalizado, se actualiza el elemento OrderID en la columna ORMORDER de la tabla ORDERS en la base de datos de WebSphere Commerce.

## Inhabilitación del servicio Web OrderFulfillment

Esta sección explica cómo inhabilitar el servicio Web OrderFulfillment de ejemplo que proporciona WebSphere Commerce.

1. Suprime el archivo JAR webservices.jar que desplegó para habilitar el servicio Web OrderFulfillment.
2. Suprime el archivo OrderFulfillment.wsdl del siguiente directorio:
  -  `dir_instal_WC\xml\messaging`

- Studio `dir_instal_WCStudio\Commerce\xml\messaging`
- AIX Linux Solaris `dir_instal_WC/xml/messaging`
- 400 `/QIBM/UserData/CommerceServer55/instances/nombre_instancia/xml/messaging`

3. Elimine las referencias al archivo `OrderFulfillment.wsdl` del archivo de configuración `nombre_instancia.xml`.
4. Si no ha ejecutado el script SQL para inhabilitar los servicios `Web OrderCreate` y `OrderStatus` o si ha comentado las siguientes sentencias SQL al inhabilitar los servicios `Web OrderCreate` y `OrderStatus`, ejecute las siguientes sentencias SQL:

```
delete from cmdreg where interfacename='com.ibm.commerce.webservice.utils.WSDLFinderCmd' and classname='com.ibm.commerce.webservice.utils.WSDLFinderCmdImpl';
```

```
delete from cmdreg where storeent_id= id_entidad_tienda and interfacename='com.ibm.commerce.order.commands.ExtOrderProcessCmd';
```

donde `id_entidad_tienda` es el identificador exclusivo de su tienda.

Las sentencias SQL anteriores están disponibles en el siguiente archivo script:

DB2 Si está utilizando una base de datos DB2:

- Windows Studio Desde un ventana de mandatos de DB2, ejecute `Webservices_UndoDBUpdate.db2.bat`
- AIX Linux Solaris Desde una ventana de mandatos, ejecute `Webservices_UndoDBUpdate.db2.sh`
- 400 Inicie una sesión QSH. Ejecute el mandato `Webservices_UndoDBUpdate.db2.sh`.

Oracle Si está utilizando una base de datos Oracle:

- Windows Studio Desde un ventana de mandatos de DB2, ejecute `Webservices_UndoDBUpdate.oracle.bat`
- AIX Solaris Desde una ventana de mandatos, ejecute `Webservices_UndoDBUpdate.oracle.sh`

Verifique el archivo de anotaciones y asegúrese de que la ejecución finaliza satisfactoriamente. El nombre y la ubicación del archivo de anotaciones se muestra después de que finalice la ejecución del archivo script.

5. Studio Suprima el código de ejemplo que ha importado en el apartado "Importación del código de ejemplo" en la página 15 y vuelva a construir el proyecto `WebSphereCommerceServerExtensionsLogic`.
6. Reinicie la instancia de `Commerce Server` para que los cambios entren en vigor si está utilizando una máquina de `WebSphere Commerce`. Si está utilizando una máquina de `WebSphere Commerce Studio`, reinicie el servidor `WebSphereCommerceServer` en el Entorno de prueba de `WebSphere`.



---

## Apéndice A. Archivos de definición WSDL de ejemplo

Las definiciones WSDL para los servicios Web de ejemplo que se proporcionan con WebSphere Commerce Versión 5.5 están disponibles en el siguiente directorio:

-  `dir_instal_WC\samples\webservices\xml\messaging`
-  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
-    `dir_instal_WC/samples/webservices/xml/messaging`
-  `/QIBM/ProdData/CommerceServer55/  
samples/webservices/xml/messaging`

Nombre del servicio Web	Nombre del archivo
OrderFulfillment	OrderFulfillment.wsdl
OrderCreate	OrderCreate.wsdl
OrderStatus	OrderStatus.wsdl



---

## Apéndice B. Utilización de IBM WebSphere Studio Application Developer

Para generar el código del servicio Web pueden utilizarse diversas herramientas. El código de ejemplo que se proporciona con esta implementación de los servicios Web se genera utilizando IBM WebSphere Studio Application Developer. La información de esta sección se ha probado en IBM WebSphere Studio Application Developer, Versión 5.0. Si no tiene WebSphere Commerce, o si no está utilizando WebSphere Commerce Studio, pero desea desarrollar un cliente de servicio Web para acceder a los servicios Web que proporciona WebSphere Commerce, puede utilizar cualquier herramienta que sea capaz de generar el código de cliente de servicio Web.

Este apéndice explica cómo utilizar WebSphere Studio para generar un cliente de prueba para los servicios que proporciona WebSphere Commerce. Para ello, debe importar los archivos de definición WSDL en WebSphere Studio, generar un cliente de prueba, ejecutar el cliente de prueba, crear y configurar una instancia de servidor de Supervisor TCP y utilizar el Supervisor TCP para rastrear los mensajes SOAP.

También puede utilizar Web Services Toolkit (WSTK) para generar el código de cliente y los tipos de datos. Para obtener más información sobre Web Services Toolkit, consulte <http://www.alphaworks.ibm.com/tech/webservicestoolkit>. WSTK presenta la novedosa tecnología de servicios Web y le ayuda a comprender las últimas especificaciones de los servicios Web. Sin embargo, para un entorno de desarrollo a nivel de producto de los servicios Web, es recomendable utilizar IBM WebSphere Studio Application Developer.

Alternativamente, puede escribir el código de cliente y los tipos de datos manualmente utilizando la implementación SOAP de Apache como cliente. Para obtener más detalles, consulte la documentación SOAP de Apache.

En esta sección se presupone que tiene conocimientos de WebSphere Studio Application Developer. Si no está familiarizado con el desarrollo de servicios Web y aplicaciones de empresa utilizando WebSphere Studio, le recomendamos que lea primero la documentación de WebSphere Studio o las guías de aprendizaje que se proporcionan en el sitio de desarrolladores de IBM (<http://www.ibm.com/developerworks/webservices>).

---

### Importación de una WSDL en WebSphere Studio Application Developer

Antes de importar una WSDL en WebSphere Studio, asegúrese de que se haya creado una aplicación Web, así como una aplicación de empresa a la que pertenezca esta aplicación Web. Si no existe una aplicación Web, debe crearla antes de continuar con estas instrucciones. Si tiene configurado el entorno de desarrollo de WebSphere Commerce, no cree una aplicación de empresa nueva ni una aplicación Web. En su lugar, utilice el proyecto `WebSphereCommerceServerExtensionsLogic`.

A continuación se ofrece una visión general del procedimiento a seguir para generar el código de cliente y los tipos de datos utilizando IBM WebSphere Studio:

1. Cree una carpeta nueva, por ejemplo wsdl, en el proyecto Web donde desea importar las definiciones WSDL. Si no tiene un proyecto Web, créelo.
2. Pulse con el botón derecho del ratón sobre la carpeta wsdl.
  - a. Seleccione **Importar** > **Sistema de archivos** y pulse **Siguiente**.
  - b. Pulse **Examinar** y vaya a la carpeta donde están almacenados los archivos de definición WSDL.
  - c. Pulse **Aceptar**. Encontrará las definiciones WSDL en el siguiente directorio:
    -  `dir_instal_WC\samples\webservices\xml\messaging`
    -  `dir_instal_WCStudio\Commerce\samples\webservices\xml\messaging`
    -    `dir_instal_WC/samples/webservices/xml/messaging`
    -  `/QIBM/ProdData/CommerceServer55/samples/webservices/xml/messaging`
3. Desde en panel derecho, seleccione los archivo WSDL que desea importar en WebSphere Studio y pulse **Finalizar**.

---

## Generación del código de cliente y los tipos de datos

Después de importar las definiciones WSDL en WebSphere Studio, puede generar el código de proxy de cliente y un cliente de prueba de ejemplo para probar el servicio Web. Un proxy de cliente es una clase Java que puede utilizarse para acceder a un servicio Web desde un cliente Java. El cliente de prueba de ejemplo proporciona una JSP de ejemplo que habilita la prueba del servicio Web desde WebSphere Studio. Haga lo siguiente para generar el proxy de cliente y el cliente de prueba de ejemplo:

**Nota:** Puede utilizar el siguiente conjunto de instrucciones para generar el código de cliente para acceder a un servicio Web, alojado en un sistema externo, desde WebSphere Commerce. La única diferencia es que lo primero que debe hacer es obtener la definición de la implementación del servicio externo del proveedor de servicios e importar la definición WSDL en WebSphere Studio.

1. Vaya a la perspectiva Web y seleccione el proyecto Web en el que desea guardar el código de cliente generado. Si no tiene ningún proyecto Web, créelo.
2. Seleccione y pulse con el botón derecho del ratón sobre el archivo de definición WSDL para generar el código de cliente. En el menú, seleccione **Nuevo** > **Otro**. Se mostrará el asistente Nuevo.
3. Seleccione **Servicio Web** en el panel izquierdo y **Cliente de servicio Web** en el panel derecho. Pulse **Siguiente**.
4. En el diálogo de cliente de servicio Web seleccione **Proxy Java** y pulse **Siguiente**.
5. En el diálogo Selección de archivo WSDL de servicio Web asegúrese de que esté seleccionado el archivo WSDL que haya elegido y pulse **Siguiente**.
6. Anote el nombre de la clase en el diálogo Generación de un proxy de enlace para el servicio Web y pulse **Siguiente**.
7. Seleccione **Probar el proxy generado**. Asegúrese de que el valor del recurso de prueba esté establecido en los archivos JSP de ejemplo de servicio Web. Pulse **Finalizar**.
8. Vaya a la perspectiva Web y expanda la carpeta de ejemplos. Se mostrarán los clientes JSP de prueba.

---

## Prueba del servicio Web

Esta sección explica cómo probar los servicios Web que proporciona WebSphere Commerce, utilizando para ello el cliente de prueba de ejemplo que ha generado en la sección precedente. Haga lo siguiente para probar el servicio Web:

1. Asegúrese de que exista al menos un servidor de Entorno de prueba de WebSphere. Si no tienen ningún servidor, créelo de la siguiente manera:
  - a. Vaya a la perspectiva del servidor.
  - b. Seleccione **Archivo > Nuevo > Servidor y configuración de servidor**.
  - c. Entre el nombre del servidor y seleccione el tipo como **WebSphere versión 5.0 > Entorno de prueba**.
2. Con el botón derecho del ratón, pulse sobre `TestClient.jsp` y seleccione **Ejecutar en servidor**. Aparecerá el cliente de prueba en una ventana de navegador.
3. Seleccione el método **getEndPoint()** en el marco izquierdo y pulse **Invocar**. En el marco de resultados, asegúrese de que el punto final devuelto sea el URL del servidor al que desea conectarse.
4. Si el punto final devuelto no coincide con el servidor donde se está ejecutando WebSphere Commerce, establezca el URL de manera que apunte al servidor de WebSphere Commerce:
  - a. Seleccione el método **setEndPoint()** en el marco izquierdo.
  - b. Actualice el URL del servidor de WebSphere Commerce en el marco derecho y pulse **Invocar**.
5. Seleccione el método de negocio del proxy de cliente. Para el proxy de `OrderCreate` el método de negocio es `orderCreate`.
  - a. Escriba los detalles de los parámetros que han de pasarse en el formulario, que se muestra en el marco derecho.
  - b. Escriba los valores para todos los campos obligatorios.
  - c. Pulse **Invocar**. Se compondrá un mensaje SOAP y se enviará al servidor identificado por el punto final.

Puede ver el resultado de la invocación del servicio Web en el marco Resultado.

---

## Utilización del Supervisor TCP para rastrear los mensajes SOAP

Esta sección describe cómo crear y configurar una instancia de servidor de supervisor TCP/IP y cómo utilizar el Supervisor TCP/IP para rastrear los mensajes SOAP. Esto le ayudará a depurar un servicio Web o a definir una plantilla de correlación para un servicio Web que esté desarrollando.

El Supervisor TCP/IP es un sencillo servidor que supervisa todas las peticiones y respuestas entre un navegador Web y un servidor de aplicaciones. Puede capturar y mostrar todo el tráfico existente entre el cliente y el servidor de aplicaciones.

En nuestro caso, debe crear una nueva instancia de servidor de Supervisor TCP/IP y hacer lo siguiente:

- Configurar el servidor de Supervisor TCP/IP para que reenvíe las peticiones al servidor de WebSphere Commerce.
- Configurar el cliente de prueba para que apunte al Supervisor TCP/IP, de manera que se puedan interceptar las peticiones y respuestas SOAP.

## Creación de una instancia de servidor de Supervisor TCP

Para crear una instancia de servidor de Supervisor TCP/IP denominada TCPMon, haga lo siguiente:

1. En el menú principal, seleccione **Archivo > Nuevo > Otro**.
2. Seleccione **Servidor** en el panel izquierdo, **Servidor y configuración de servidor** en el panel derecho y pulse **Siguiente**.
3. En la ventana Crear un servidor nuevo y una configuración de servidor nueva, haga lo siguiente:
  - a. Escriba un **Nombre del servidor**, por ejemplo, TCPMon.
  - b. Seleccione el **Tipo de servidor**, Servidor de supervisión TCP/IP.
  - c. Pulse **Siguiente**.
4. En la ventana Crear un servidor nuevo y una configuración de servidor nueva, haga lo siguiente:
  - a. Escriba el nombre de sistema principal del servidor de WebSphere Commerce en el campo **Sistema principal remoto**.
  - b. En el campo **Puerto remoto**, escriba el número de puerto en el que escucha el servidor de WebSphere Commerce. Puede utilizar localhost como nombre de sistema principal si WebSphere Commerce está ejecutándose en la misma máquina que el cliente.
  - c. Pulse **Finalizar**.

## Configuración de la instancia de servidor de Supervisor TCP

Configure el Supervisor TCP de manera que el puerto de escucha del servidor de WebSphere Commerce sea el puerto remoto del Supervisor TCP. Haga lo siguiente:

1. Vaya a la perspectiva del servidor.
2. Efectúe una doble pulsación del ratón sobre la instancia del Supervisor TCP en la vista de configuración del servidor. La configuración del servidor de Supervisor TCP se abrirá en la vista del editor, tal como se muestra en la siguiente figura:

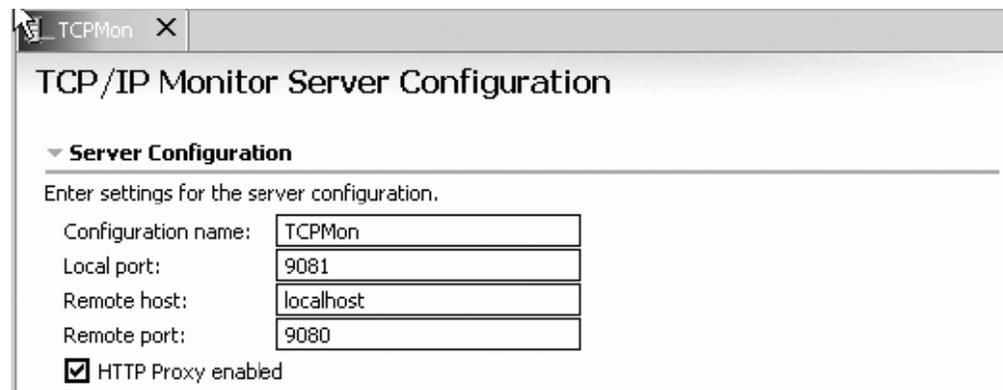


Figura 6. Configuración del servidor de Supervisor TCP/IP (versión en inglés)

En este ejemplo, el puerto local es el puerto en el que escucha el Supervisor TCP y el puerto remoto es el puerto donde escucha el servidor de WebSphere Commerce.

## Rastreo de los mensajes SOAP utilizando el Supervisor TCP/IP

Después de configurar el Supervisor TCP/IP, utilícelo para rastrear los mensajes SOAP. Para hacerlo, inicie el cliente de prueba de ejemplo que se ha generado en el servidor y configúrelo para que envíe la petición al Supervisor TCP/IP. Haga lo siguiente:

1. Asegúrese de que el Supervisor TCP/IP y el servidor de WebSphere Commerce estén ejecutándose.
  - a. Si el servidor de WebSphere Commerce está ejecutándose fuera de WebSphere Studio, inicie el servidor utilizando la consola de administración de WebSphere Application Server.
  - b. Si el servidor de WebSphere Commerce está ejecutándose dentro de WebSphere Commerce Studio, vaya a la perspectiva de **Servidor**. Abra la vista **Servidores**. Si el **Estado** del servidor WebSphereCommerceServer es Detenido, pulse con el botón derecho del ratón sobre el servidor y seleccione **Iniciar**. Espere hasta que el Estado cambie a Iniciado en la vista de servidores.
2. Vaya a la perspectiva Web. Utilice **Examinar** para ir a `sample\TestClient.jsp.Right`, pulse `sample\TestClient.jsp` y seleccione **Ejecutar en servidor**.
3. En la ventana de selección de servidor, asegúrese de que esté seleccionado el servidor en el que debe ejecutarse el archivo JSP. Pulse **Finalizar**.
4. Si se le indica que seleccione un cliente de servidor, seleccione el **Navegador Web** y pulse **Finalizar**. Se abrirá una nueva ventana de navegador con tres marcos, tal como se muestra en la figura siguiente:

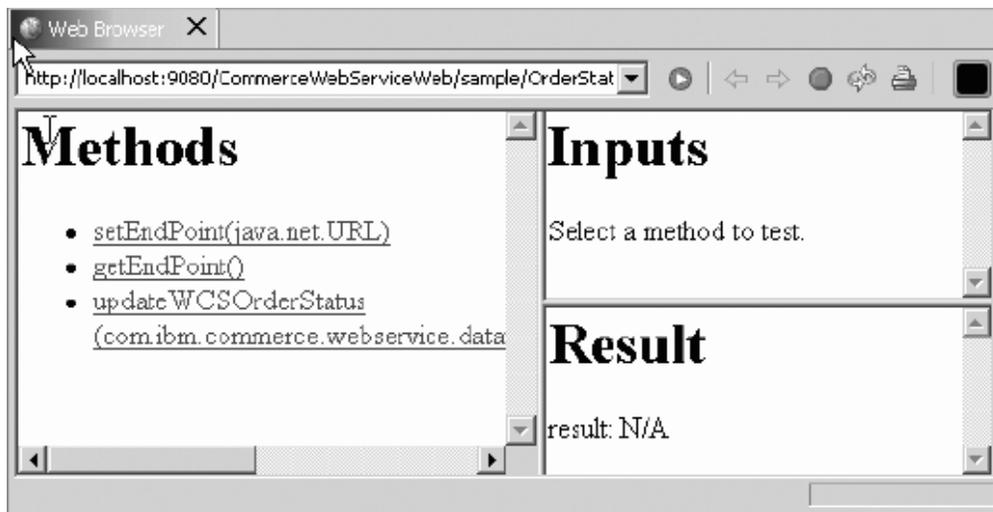


Figura 7. Selección de un navegador Web (versión en inglés)

- a. Seleccione el método `getEndPoint()` desde el marco Métodos, tal como se muestra en la figura siguiente:

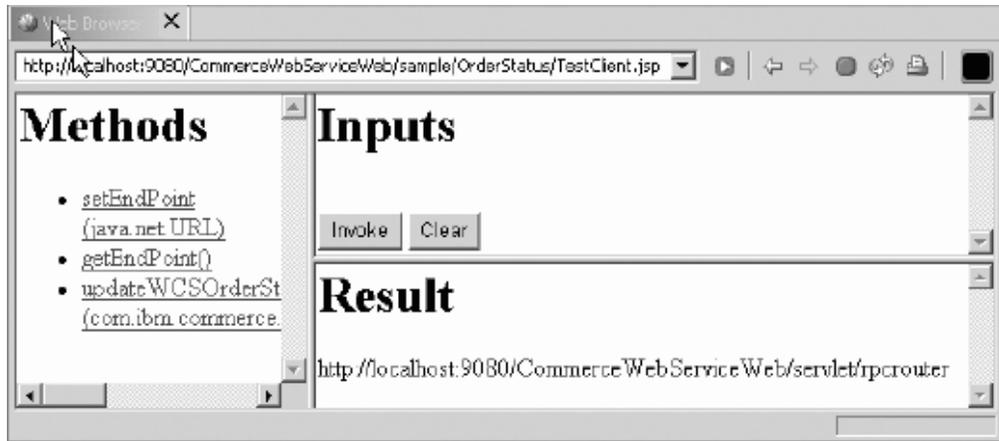


Figura 8. Selección del método `getEndPoint()` (versión en inglés)

- b. Pulse **Invocar** en el marco Entradas. Se mostrarán los resultados en el marco Resultado.
5. Para configurar el cliente de prueba para que envíe la petición al Supervisor TCP debe hacer lo siguiente:
  - a. Seleccione el método **setEndPoint()** en el marco Métodos.
  - b. En el marco Entradas, escriba el **url** del Supervisor TCP. Por ejemplo, `http://localhost:9080/webapp/wcs/stores/servlet/`, donde 9080 es el puerto donde se ejecuta el Supervisor TCP y `webapp/wcs/stores/servlet/` es la vía de acceso Web del servlet de petición de WebSphere Commerce.

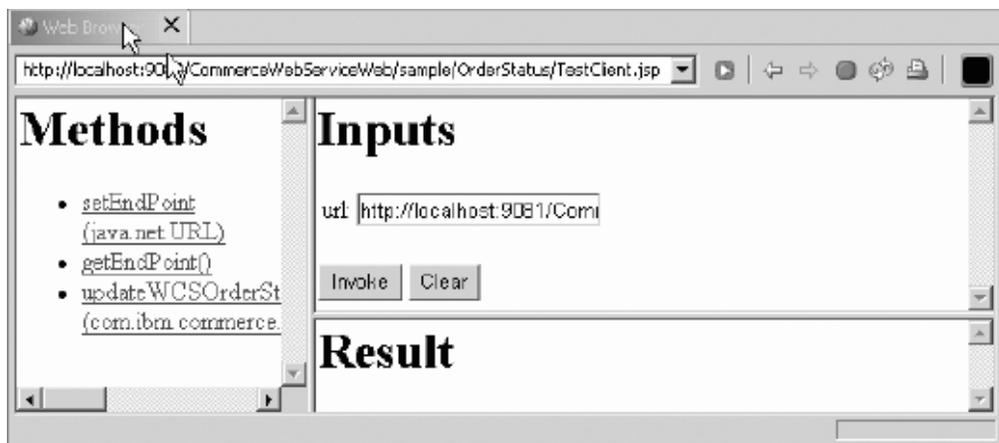


Figura 9. Selección del método `setEndPoint()` (versión en inglés)

- c. Pulse **Invocar**.
6. Vuelva a invocar el método **getEndPoint()** para verificar el punto final.
7. Para invocar la lógica de negocio, haga lo siguiente:
  - a. Pulse sobre el método de lógica de negocio, por ejemplo, **updateWCSOrderStatus()** en el marco Métodos.
  - b. Entre los valores para todos los parámetros obligatorios en el marco Entradas, tal como se muestra en la figura siguiente:

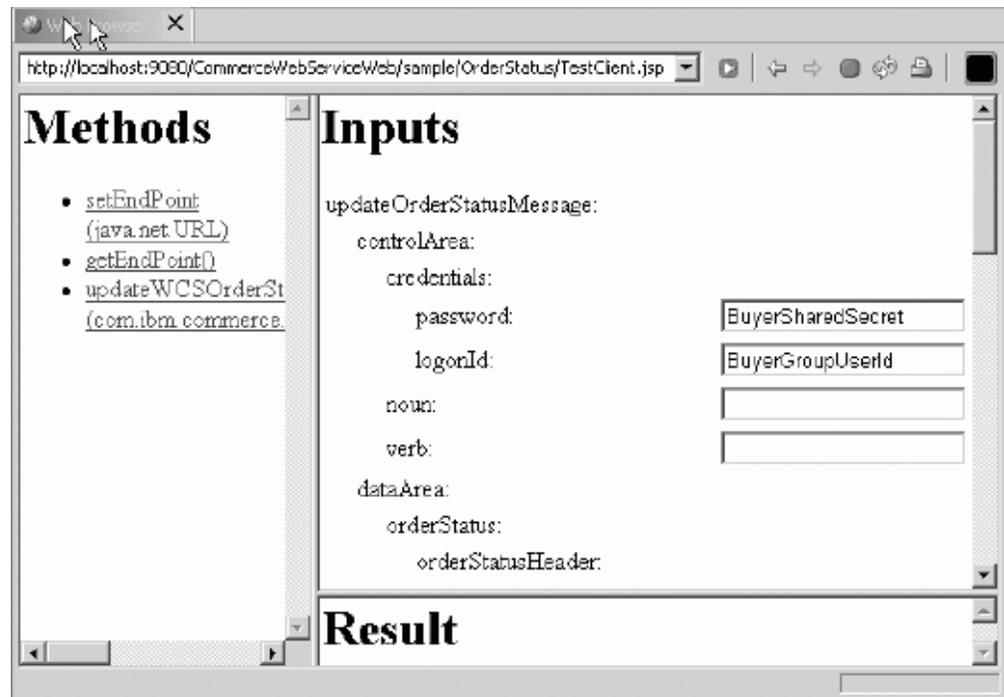


Figura 10. Selección de la lógica de negocio (versión en inglés)

- c. Pulse **Invocar**.
- d. Se mostrarán los resultados en el marco Resultado.
- e. Vaya a la vista del Supervisor TCP/IP y seleccione el mensaje más reciente. Se visualizará la Petición y la Respuesta SOAP, tal como se muestra en la siguiente figura:

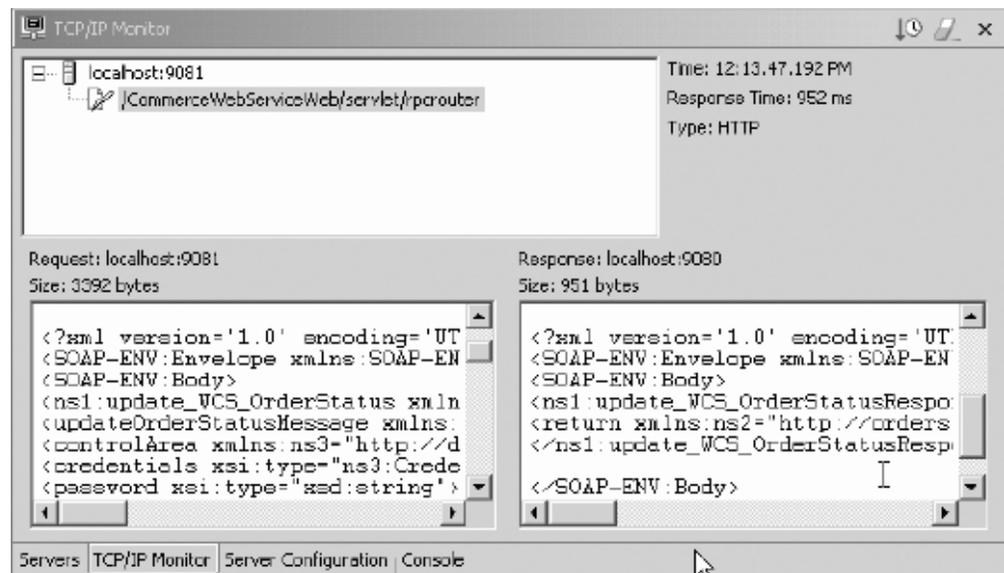


Figura 11. Visualización de la Petición y la Respuesta SOAP (versión en inglés)

---

## Personalización del código de cliente

Cuando genere el código de cliente, debe tener en cuenta que el código de cliente que se genera por omisión puede que no siempre funcione con WebSphere Commerce. Por ejemplo, si está utilizando la clase por omisión `BeanSerializer` para serializar los tipos de datos, algunos de los tipos de datos se convertirán en series XML que WebSphere Commerce podría no comprender en el extremo del servidor. Por ejemplo, el servicio `Web OrderCreate` de ejemplo contiene una estructura de datos de este tipo. Esta estructura de datos se proporciona como una interfaz para el mandato de WebSphere Commerce y se denomina estructura de datos `UserData`. La estructura de datos `UserData` permite pasar datos personalizados al mandato utilizando la interfaz dirigida por mensajes.

La sección siguiente describe cómo serializar la estructura de datos `UserData` de forma que WebSphere Commerce pueda utilizarla.

**Nota:** La estructura de datos `UserData` proporcionada es un ejemplo y es opcional para el servicio `Web OrderCreate`.

WebSphere Commerce permite que los sistemas externos incluyan parejas nombre-valor personalizadas como parte del mensaje XML que se envía. Para dar soporte a esta posibilidad, se utiliza un tipo de XPATH denominado `USERDATA` que se procesará tal como se describe en el siguiente mensaje XML de ejemplo que incluye el elemento `UserData`:

```
<UserData>
  <UserDataField name="abc">xyz</UserDataField>
</UserData>
```

La definición de XPATH para este elemento `UserData` es la siguiente:

```
<Tag XPath='.../UserData/UserDataField'
  XPathType='USERDATA' />
```

El procesador de XPATH convierte el elemento `UserData` como la pareja nombre-valor `"abc=xyz"`.

Si tiene una estructura de datos similar a la siguiente, podría haber un problema con el serializador de beans por omisión que se proporciona con la implementación SOAP de Apache. La implementación SOAP de Apache convierte todos los tipos de datos en una estructura de datos para separar los elementos del documento XML. Si está utilizando el cliente SOAP de Apache para convertir los objetos Java en un mensaje XML SOAP, WebSphere Commerce no podrá analizar este XML y, en consecuencia, no podrá formar un objeto `UserData`.

```
class UserData
(
  private String userDataField = null;
  private String name = null;
  // getter and setter methods here ....
)
```

Esta clase se serializará en una estructura XML.

```
<UserData>
  <UserDataField>xyz</UserDataField>
  <name>abc</name>
</UserData>
```

Como resultado, el correlacionador de mensajes de WebSphere Commerce no podrá interpretarla como una estructura `USERDATA`.

Para poder dar soporte a la serialización de los tipos de datos UserData en el formato necesario, deberá escribir un serializador personalizado y registrarlo en el registro de correlaciones SOAP. El serializador personalizado debe tener el siguiente método marshall:

```

/*
 * serializar com.ibm.commerce.webservices.datatype.UserData
 * al esquema XML correspondiente, tal como espera WC.
 *
 * @param String inScopeEncStyle
 * @param Class javaType
 * @param Object src
 * @param Object context
 * @param Writer sink
 * @param NSStack nsStack
 * @param XMLJavaMappingRegistry xjmr
 * @param SOAPContext ctx
 * @exception IllegalArgumentException
 * @exception IOException
 */
public void marshall(String inScopeEncStyle,
                    Class javaType,
                    Object src,
                    Object context,
                    Writer sink,
                    NSStack nsStack,
                    XMLJavaMappingRegistry xjmr,
                    SOAPContext ctx)
    throws IllegalArgumentException, IOException
{
    if (!javaType.equals(UserData.class))
    {
        throw new IllegalArgumentException
("Sólo se pueden serializar instancias UserData");
    }
    nsStack.pushScope();
    if (src != null)
    {
        SoapEncUtils.generateStructureHeader(inScopeEncStyle,
                                           javaType,
                                           context,
                                           sink,
                                           nsStack,
                                           xjmr);

        UserData data = (UserData)src;
        sink.write("<UserDataField name=\"" + data.getName() + "\">");
        sink.write(data.getValue() +
"</UserDataField>");
        sink.write("</" + context + ">");
    }
    else
    {
        SoapEncUtils.generateNullStructure(inScopeEncStyle,
                                           javaType,
                                           context,
                                           sink,
                                           nsStack,xjmr);
    }
    nsStack.popScope();
}

```

**Nota:** El servicio de ejemplo que se proporciona con WebSphere Commerce Versión 5.5 está diseñado para funcionar con el código anterior.



---

## Avisos

Esta información se ha desarrollado para productos y servicios ofrecidos en los Estados Unidos. Es posible que IBM no ofrezca en otros países los productos, servicios o características descritos en este documento. Solicite información al representante local de IBM acerca de los productos y servicios disponibles actualmente en su zona. Cualquier referencia a un producto, programa o servicio de IBM no pretende afirmar ni implica que sólo pueda utilizarse ese producto, programa o servicio de IBM. En su lugar puede utilizarse cualquier producto, programa o servicio funcionalmente equivalente que no vulnere ninguno de los derechos de propiedad intelectual de IBM. No obstante, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o aplicaciones pendientes de patente que cubran temas tratados en este documento. La posesión de este documento no le otorga ninguna licencia sobre dichas patentes. Puede enviar consultas sobre licencias, por escrito, a:

IBM Director of Licensing  
IBM Corporation  
North  
Castle Drive  
Armonk, NY 10504-1785  
EE.UU.

Para realizar consultas relacionadas con la información de doble byte (DBCS), póngase en contacto con el departamento de propiedad intelectual de IBM en su país o envíe sus consultas, por escrito, a:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokio 106-0032, Japón

El párrafo siguiente no se aplica al Reino Unido ni a ningún otro país donde tales disposiciones estén en contradicción con la legislación local:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL", SIN GARANTÍAS DE NINGUNA CLASE, NI EXPLÍCITAS NI IMPLÍCITAS, INCLUIDAS, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS DE NO INFRACCIÓN, COMERCIALIZACIÓN O IDONEIDAD PARA UNA FINALIDAD DETERMINADA. Algunas legislaciones no contemplan la exclusión de garantías, ni implícitas ni explícitas, en determinadas transacciones, por lo que puede haber usuarios a los que no les afecte dicha norma.

Esta información puede contener imprecisiones técnicas o errores tipográficos. La información aquí contenida está sometida a cambios periódicos; tales cambios se irán incorporando en nuevas ediciones de la publicación. IBM se reserva el derecho de realizar cambios y/o mejoras, cuando lo considere oportuno y sin previo aviso, en los productos y/o programas descritos en esta publicación.

IBM puede utilizar o distribuir la información que se le envíe del modo que estime conveniente sin incurrir por ello en ninguna obligación para con el remitente.

Los propietarios de licencias de este programa que deseen obtener información sobre el mismo con el fin de permitir: (i) el intercambio de información entre programas creados independientemente y otros programas (incluido éste) y (ii) el uso mutuo de la información que se ha intercambiado, deberán ponerse en contacto con:

IBM Canada Ltd.  
Office of the Lab Director  
8200 Warden Avenue,  
Markham, Ontario L6G 1C7  
Canadá

Dicha información puede estar disponible sujeta a los términos y condiciones apropiados, incluyendo, en algunos casos, el pago de una cantidad.

IBM proporciona el programa bajo licencia descrito en esta información, y todo el material bajo licencia disponible para el mismo, bajo los términos del Contrato de cliente IBM, el Acuerdo Internacional de Programas bajo Licencia de IBM o de cualquier acuerdo equivalente entre IBM y el cliente.

Este documento puede contener información sobre productos de otras empresas, incluyendo referencias a los sitios Internet de dichas empresas. IBM no es responsable de la exactitud de esa información, de que esté completa ni del uso que se haga de ella.

#### LICENCIA DE COPYRIGHT:

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente que ilustran las técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir libremente estos programas de ejemplo, sin pagar por ello a IBM, con la finalidad de desarrollar, utilizar, comercializar o distribuir programas de aplicación conformes a la interfaz de programas de aplicación para la plataforma operativa para la cual están escritos los programas de ejemplo. Estos ejemplos no han sido probados en profundidad bajo todas las condiciones. En consecuencia, IBM no puede garantizar ni afirmar la fiabilidad, solidez o funcionalidad de estos programas. Puede copiar, modificar y distribuir libremente estos programas de ejemplo, sin pagar por ello a IBM, con la finalidad de desarrollar, utilizar, comercializar o distribuir programas de aplicación conformes a las interfaces de programas de aplicación de IBM.

---

## Marcas registradas

IBM y el logotipo de IBM son marcas comerciales o marcas registradas de International Business Machines Corporation en los Estados Unidos y/o en otros países.

Los siguientes términos son marcas comerciales o marcas registradas de International Business Machines Corporation en los Estados Unidos y/o en otros países:

<b>AS/400</b>	<b>iSeries</b>
<b>AIX</b>	<b>MQSeries</b>
<b>DB2</b>	<b>S/390</b>
<b>DB2 Universal Database</b>	<b>zSeries</b>
<b>@server</b>	<b>WebSphere</b>
<b>IBM</b>	

Microsoft, Windows y Windows NT son marcas comerciales o marcas registradas de Microsoft Corporation.

Java y todas las marcas registradas basadas en Java son marcas registradas de Sun Microsystems, Inc. en los Estados Unidos y/o en otros países.

Otros nombres de empresas, productos y servicios pueden ser marcas comerciales o marcas de servicio de otras compañías.





**IBM**