

IBM® WebSphere Commerce



Guía del desarrollador de tiendas

Versión 54

IBM® WebSphere Commerce



Guía del desarrollador de tiendas

Versión 54

Aviso:

Antes de utilizar esta información y el producto al que da soporte, lea la información del apartado Avisos.

Segunda edición (mayo de 2002)

Esta edición se aplica a los productos siguientes:

- IBM WebSphere Commerce Business Edition para Windows NT y Windows 2000, Versión 5.4
- IBM WebSphere Commerce Business Edition para AIX, Versión 5.4
- IBM WebSphere Commerce Business Edition para Solaris, Versión 5.4
- IBM WebSphere Commerce Studio, Business Developer Edition para Windows NT y Windows 2000, Versión 5.4
- IBM WebSphere Commerce Professional Edition para Windows NT y Windows 2000, Versión 5.4
- IBM WebSphere Commerce Professional Edition para AIX, Versión 5.4
- IBM WebSphere Commerce Professional Edition para Solaris, Versión 5.4
- IBM WebSphere Commerce Studio, Professional Developer Edition para Windows NT y Windows 2000, Versión 5.4

y a todos los releases y modificaciones posteriores de los productos listados anteriormente, hasta que se indique lo contrario en nuevas ediciones. Asegúrese de utilizar la edición correcta para el nivel del producto.

Efectúe el pedido de publicaciones a través del representante de IBM o de la sucursal de IBM que atiende a su localidad. En la dirección que figura a continuación no hay existencias de publicaciones.

IBM agradece sus comentarios. Puede enviar sus comentarios mediante alguno de estos métodos:

1. Por correo electrónico a la dirección que se indica a continuación. No olvide incluir su dirección de correo electrónico completa si desea obtener una respuesta.

Internet: hojacom@vnet.ibm.com

2. Por correo postal a la siguiente dirección:

IBM S.A.
National Language Solutions Center
Av. Diagonal 571, Edif. L'illa
08029 Barcelona

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. Reservados todos los derechos.

Contenido

Antes de empezar	vii
Convenios utilizados en este manual	vii
Dónde encontrar la información más reciente	viii

Parte 1. Visión general 1

Capítulo 1. Visión general de la arquitectura de la tienda 3

¿Qué es una tienda en línea?	3
Composición de una tienda	3
Arquitectura de la tienda	4

Capítulo 2. Desarrollo de la tienda 7

Opciones de desarrollo de la tienda.	7
Crear una tienda basada en un ejemplo	7
Crear una tienda desarrollando elementos de tienda nuevos	8
Crear una tienda utilizando una combinación de elementos de tienda de ejemplo y de tienda nueva	8
El archivador de tienda.	8
Archivadores de tienda de ejemplo	9
Determinar cuándo utilizar un archivador de tienda	10
Herramientas de desarrollo de tienda.	10
Herramientas para desarrollar el escaparate	10
Herramientas para desarrollar los datos de la tienda	11
Herramientas para desarrollar la oficina interna	12
El rol de Desarrollador de tiendas	13

Parte 2. Desarrollo del escaparate de la tienda 15

Capítulo 3. Desarrollo del escaparate de la tienda. 17

Arquitectura del escaparate	17
Mandatos y vistas por omisión	18
Creación de las páginas de tienda	18
Desarrollar una lista de las páginas de la tienda	18
Desarrollar una lista de los URL de mandatos y vistas	22
Asociar nombres de archivo JSP a las vistas	23

Parte 3. Visión general de los datos de la tienda 25

Capítulo 4. Datos de la tienda 27

¿Qué son los datos de la tienda?	27
Modelo de información de los datos de la tienda	27
Elementos de datos de la tienda	28
Arquitectura de datos de la tienda.	29

Arquitectura de datos de la tienda y las tiendas de ejemplo	32
Herramientas para la creación de datos	32
WebSphere Commerce Loader package	32
Servicios de tienda	32
Consola de administración	32
WebSphere Commerce Accelerator.	32
Consola de administración de la organización	33
Diagrama de herramientas y datos de la tienda	33

Parte 4. Desarrollo de los datos de la tienda 37

Capítulo 5. Elementos de sitio. 39

Los elementos de sitio en WebSphere Commerce	39
Idioma	40
Atributos de miembros	40
Tipos de atributo	40
Tipos de grupos de miembros	40
Usuario.	40
Organización	40
Rol	40
Conversión de unidades de cantidad	41
Unidades de cantidad	41
Tipos de impuestos.	41
Uso de cálculo	41
Moneda	41
Utilización de números	41
Tipos de artículo.	41
Formatos de dispositivo	42
Creación de elementos de sitio en WebSphere Commerce.	42

Capítulo 6. Elementos de tienda 43

Los elementos de tienda en WebSphere Commerce	43
Entidad de tienda	44
Creación de elementos de tienda en WebSphere Commerce.	44
Creación de elementos de datos de tienda en un archivo XML	45

Capítulo 7. Datos de registro de los mandatos, vistas y URL 51

Registro de mandatos, vistas y URL en WebSphere Commerce.	51
Creación de un archivo XML para registrar mandatos, vistas y URL	51

Capítulo 8. Elementos de catálogo. 55

Los catálogos en WebSphere Commerce	55
Catálogos	56
Grupos de catálogo.	56
Entradas de catálogo	57
Productos	57

Artículos	57
Paquetes	57
Paquetes compuestos	58
Kits dinámicos	58
Conjuntos de productos	58
Atributos	58
Valores de atributo	59
Atributos de paquete	59
Valores de atributo de paquete	59
Creación de elementos de catálogo en WebSphere Commerce	59
Creación de un catálogo maestro	60
Visualización de los elementos de catálogo de la tienda	77
Creación de un catálogo de navegación	80
Creación de ciclos de categoría	80
Adición de un producto a una segunda categoría	82
Gestión de elementos de catálogo en WebSphere Commerce	85
Grupos de catálogo	85
Entradas de catálogo	86
Herramientas de Gestión de productos	87
Catalog Manager	88

Capítulo 9. Elementos de cálculo de precios 91

Descripción del cálculo de precios en WebSphere Commerce	91
Oferta	91
Precio de oferta	92
Contenedor de propuestas de comercio	92
Términos y condiciones	92
Tipos de términos y condiciones de fijación de precios	92
Acuerdo de comercio	92
Participante	93
Rol de participante	93
Contrato	93
Política de negocio	93
Política de fijación de precios	93
Envío de entrada de catálogo	94
Otros elementos asociados al cálculo de precio	94
Creación de elementos de cálculo de precio en WebSphere Commerce	94
Creación de elementos de cálculo de precio en un archivo XML	95

Capítulo 10. Elementos de contrato 99

Descripción de los contratos en WebSphere Commerce	100
Cuenta (cuenta de negocio).	100
Contrato	101
Acuerdo de comercio	101
Términos y condiciones	102
Políticas de negocio	102
Adjunto	102
Artículo de pedido	103
Creación de un elemento de contrato por omisión en WebSphere Commerce	103

Creación de archivos XML de políticas de negocio	105
Creación de un archivo XML de contrato por omisión	106

Capítulo 11. Elementos de despacho de pedidos 111

Los elementos de despacho de pedidos en WebSphere Commerce	112
Centro de despacho de pedidos	112
Recepciones	113
RaDetail	113
Inventario	113
Acuerdos de envío	113
Otros elementos relacionados con el despacho de pedidos	113
Creación de elementos de despacho de pedidos en WebSphere Commerce	114
Creación de elementos de despacho de pedidos	116

Capítulo 12. Elementos de campaña 119

Las campañas en WebSphere Commerce	119
Creación de elementos de campaña en WebSphere Commerce	120

Capítulo 13. Elementos de pago 121

Creación de elementos de pago utilizando un archivo XML	121
---	-----

Capítulo 14. Elementos de idioma 123

Los elementos de idioma en WebSphere Commerce	123
Idioma por omisión	123
Idiomas soportados	124
Idioma alternativo	124
Creación de elementos de idioma en WebSphere Commerce	124

Capítulo 15. Elementos de moneda 125

Los elementos de moneda en WebSphere Commerce	125
Formato de moneda	126
Utilización de números	126
Descripción del formato de moneda	126
Moneda soportada	126
Norma de conversión de moneda	126
Moneda de contravalor	126
Creación de elementos de moneda en WebSphere Commerce	127
Creación de elementos de moneda utilizando un archivo XML	127

Capítulo 16. Elementos de unidad de medida 131

Las unidades de medida en WebSphere Commerce	131
Unidad de cantidad y formato de la unidad de cantidad	132
Creación de unidades de medida en WebSphere Commerce	133

Capítulo 17. Elementos de jurisdicción 135

Los elementos de jurisdicción en WebSphere Commerce	135
Creación de elementos de jurisdicción en WebSphere Commerce	136

Capítulo 18. Elementos de envío . . . 137

Los elementos de envío en WebSphere Commerce	137
Modalidades de envío	138
Códigos de cálculo	138
Jurisdicciones y grupos de jurisdicciones	139
Creación de elementos de envío en WebSphere Commerce	139
Creación de elementos de envío utilizando un archivo XML	140
Creación de elementos de despacho de envío	149
Creación de elementos de tienda-catálogo-envío	151
Creación de una modalidad de envío por omisión	153

Capítulo 19. Elementos de impuestos 155

Los elementos de impuestos en WebSphere Commerce	155
Categoría de impuestos	156
Código de cálculo	156
Jurisdicciones y grupos de jurisdicciones	158
Creación de elementos de impuestos en WebSphere Commerce	158
Creación de elementos de impuestos utilizando un archivo XML	159
Creación de elementos de despacho de impuestos	170
Creación de elementos tienda-catálogo-impuestos	171

Capítulo 20. Elementos de descuento 175

Los descuentos en WebSphere Commerce	175
Código de cálculo	175
Creación de elementos de descuento en WebSphere Commerce	176

Capítulo 21. Elementos de inventario 179

Los elementos de inventario en WebSphere Commerce	180
Inventario ATP	180
Inventario no ATP	182
Creación de elementos de inventario en WebSphere Commerce	183

Capítulo 22. Elementos de pedido . . 185

Los elementos de pedido en WebSphere Commerce	185
Pedidos y artículos de pedido	185
Artículos de pedido	186
Creación de elementos de pedido en WebSphere Commerce	188

Capítulo 23. Elementos de cliente y vendedor 189

Los elementos de cliente en WebSphere Commerce	189
--	-----

Información de dirección	190
Listas de intereses	190
Descripción de los elementos del vendedor en WebSphere Commerce	190
Tiendas	191
Cuentas	192
Contratos	192
Conjuntos de productos	192
Listas de precios	192
Catálogos	193
Centros de despacho de pedidos	193
Artículos de inventario	194
Los elementos de miembro en WebSphere Commerce	194
Miembros	195
Atributos de miembro	196
Roles	196
Creación de elementos de miembro en WebSphere Commerce	197

Parte 5. Adición de control de acceso a la tienda 199

Capítulo 24. Control de acceso en la tienda. 201

Descripción del control de acceso en WebSphere Commerce	201
Políticas de control de acceso	202
Control de acceso en las tiendas	203
Adición de control de acceso a la tienda	208
Creación de control de acceso en la tienda	208
Edición de los archivos de control de acceso del archivador de la tienda	211

Parte 6. Empaquetado de la tienda 215

Capítulo 25. Empaquetado de una tienda. 217

Creación de un archivador de tienda	217
Creación de un archivador de tienda de ejemplo	219

Parte 7. Publicación de la tienda 225

Capítulo 26. Publicación de una tienda completa 227

Descripción de la publicación en WebSphere Commerce	227
Iniciar la publicación	228
Comprobaciones previas a la publicación	229
Publicar elementos	231
Configurar el pago	238
Archivos de anotaciones cronológicas de la publicación	238

Capítulo 27. Visión general de la carga de datos de tienda 241

Descripción de la carga de datos en WebSphere Commerce	242
Mandatos de Loader package para cargar datos de tienda	245
Mandatos de Loader package para transformar y extraer datos	259
Herramientas relacionadas con los mandatos de Loader package	268
Carga de los datos de la tienda	268
Utilización de los mandatos y scripts de Loader package	269
Ejemplos de resolución de identificadores	269
Ejemplo de carga de datos	276

Capítulo 28. Carga de grupos de elementos de base de datos de WebSphere Commerce 279

Grupos de elementos de base de datos	279
Secuencia de carga de elementos de base de datos	280
Carga de una tienda	281
Carga de grupos de elementos de base de datos	287

Capítulo 29. Publicación de cuentas de negocio y contratos 293

Publicación de cuentas de negocio y contratos utilizando los Servicios de tienda o la línea de mandatos.	294
Publicación de cuentas de negocio y contratos utilizando mandatos	294
Publicación de elementos de cuenta de negocio	294
Publicación de elementos de contrato	295

Capítulo 30. Publicación de elementos de escaparate y de archivos de configuración de tienda 297

Publicación de elementos de escaparate y de archivos de configuración de tienda utilizando los Servicios de tienda o la línea de mandatos	297
Publicación de elementos de escaparate y archivos de configuración de tienda copiándolos en WebSphere Commerce Server	298

Parte 8. Adición de características de WebSphere Commerce a la tienda 301

Capítulo 31. Adición del servicio de atención al cliente a la tienda 303

Descripción del servicio de atención al cliente en una tienda	303
---	-----

Utilización del conjunto de marcos	305
Supervisión de los clientes que utilizan el servicio de atención al cliente	306
Adición del servicio de atención al cliente a la tienda	310
Parte 1: Instalación de requisitos previos	310
Parte 2: Copia de los archivos de integración del servicio de atención al cliente de la tienda de ejemplo	311
Parte 3: Adición del conjunto de marcos a la tienda	312
Parte 4: Adición del código para obtener el nombre o ID del cliente	313
Parte 5: Adición de código para determinar qué página está examinando el cliente	314
Parte 6: Adición de código para hacer el seguimiento del número de artículos del carro de la compra	314
Parte 7: Adición de un enlace al servicio de atención al cliente	315
Parte 8: Cambio de mensajes que se muestran al cliente	315

Capítulo 32. Adición de zonas de e-marketing a la tienda 317

Zona de e-marketing	317
Bean e-MarketingSpot	320
Adición de una zona de e-marketing a las páginas de la tienda	320

Parte 9. Apéndices 323

Apéndice A. Descripción de UML . . . 325

Apéndice B. Creación de los datos . . . 327

Creación de datos para tiendas de ejemplo	327
Servicios de tienda y tiendas de ejemplo	328

Apéndice C. sarinfo.xml 329

Ejemplo de sarinfo.xml	329
----------------------------------	-----

Apéndice D. sarrule.xml 335

Ejemplo de sarrule.xml	335
----------------------------------	-----

Apéndice E. Grupos de elementos de base de datos 339

Dependencias de grupos de elementos de base de datos	339
--	-----

Apéndice F. Avisos 345

Marcas registradas.	347
-----------------------------	-----

Antes de empezar

La publicación *IBM WebSphere Commerce, Guía del desarrollador de tiendas* proporciona información sobre la arquitectura de las tiendas de WebSphere Commerce y el proceso de desarrollo de las tiendas. En particular, proporciona detalles sobre los siguientes temas:

- Opciones de desarrollo de la tienda
- El archivador de tienda
- Herramientas de desarrollo de la tienda
- Desarrollo del escaparate
- Desarrollo de los datos de la tienda
- Arquitectura de los datos de la tienda
- Modelo de información de los datos de la tienda
- Adición de control de acceso a la tienda
- Empaquetado de la tienda
- Publicación de la tienda
- Adición de características de WebSphere Commerce a la tienda

Convenios utilizados en este manual

Este manual utiliza los siguientes convenios para resaltar el texto:

La **negrita** indica mandatos o controles de interfaz gráfica de usuario (GUI) como, por ejemplo, nombres de campos, botones u opciones de menú.

El monoespaciado indica ejemplos de texto que se escriben exactamente como se muestra, además de vías de acceso a directorios.

La *cursiva* se utiliza para dar énfasis y para variables que se sustituyen por sus propios valores.



Este icono indica una sugerencia: información adicional que puede ayudarle a realizar una tarea.

▶ **NT** indica información específica de Windows NT.

▶ **2000** indica información específica de Windows 2000.

▶ **AIX** indica información específica de AIX.

▶ **Solaris** indica información específica de Solaris Operating Environment.

▶ **400** indica información específica de IBM eserver iSeries 400 (anteriormente denominado AS/400).


▶ **Linux** indica información específica de Linux.


▶ **Business** indica información específica de WebSphere Commerce Business Edition.

▶ **Professional** indica información específica de WebSphere Commerce Professional Edition.

Dónde encontrar la información más reciente

Es posible que esta publicación se actualice más adelante. Encontrará las actualizaciones en el siguiente sitio Web de WebSphere Commerce:

 http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html

Las actualizaciones pueden incluir información nueva.

Parte 1. Visión general

Capítulo 1. Visión general de la arquitectura de la tienda

Este capítulo proporciona una introducción a la arquitectura de la tienda de WebSphere Commerce Server.

¿Qué es una tienda en línea?

Una tienda en línea es una tienda que utiliza tecnologías de Internet para vender o intercambiar mercancías o servicios. Se compone de un conjunto de páginas Web que visualizan los productos y permiten a los clientes comprarlos. La página de presentación recibe a los clientes a la tienda y les dirige a los productos y servicio. Las páginas del catálogo en línea agrupan los productos y dirigen a los clientes a las páginas de productos, donde pueden encontrar información detallada acerca del producto. En las tiendas de empresa a cliente, la página Carro de la compra realiza el mismo rol que un carro de la compra físico: el cliente va incluyendo en el carro los productos que desea comprar y luego los paga en las páginas Pasar por caja. En los sitios de empresa a empresa, determinadas páginas le permiten someter pedidos y solicitudes de presupuesto (RFQ).

Las páginas Web dinámicas se crean utilizando la tecnología JavaServer Pages (JSP). Cada página contiene HTML para el contenido estático, JavaScript de la parte del cliente para manejar los datos de entrada y ofrecer una presentación sofisticada de los datos, los URL para invocar mandatos y otras vistas de WebSphere Commerce Server, así como códigos JSP y código Java para generar contenido dinámico. Con WebSphere Commerce Studio y WebSphere Commerce se incluye un conjunto de beans de datos de comercio que pueden utilizar los archivos JSP, y que le permiten acceder a la información de la base de datos, por ejemplo al precio de un producto o a los atributos del producto.

La tienda también incluye los elementos de base de datos necesarios para crear una tienda en funcionamiento. Por ejemplo, una tienda en funcionamiento debe incluir datos sobre catálogos, impuestos, envío y moneda.

Composición de una tienda

Una tienda en línea está formada por los siguientes elementos:

- Escaparate

La parte externa de la tienda o la parte que se muestra a los clientes se conoce como escaparate. El escaparate se compone de elementos Web tales como páginas HTML, archivos JSP, hojas de estilo, imágenes, gráficos y otros tipos de archivos multimedia.

Esta guía describe los conceptos y las tareas necesarias para crear los archivos JSP que generan las páginas de la tienda. Para obtener más información, consulte el Capítulo 3, "Desarrollo del escaparate de la tienda" en la página 17.

- Oficina interna

La parte de la tienda que los clientes no ven, los mandatos, el código personalizado y la implementación de la lógica de negocio que permite a un cliente comprar un producto del escaparate se conoce como la oficina interna.

Para obtener información más detallada sobre la creación de lógica de negocio o código personalizado, consulte la publicación *IBM WebSphere Commerce, Guía del programador*.

- Datos de la tienda

Los elementos de datos que forman la tienda. Para que una tienda funcione correctamente debe tener los datos en una ubicación que dé soporte a todas las actividades de los clientes. Por ejemplo, para que un cliente efectúe una compra, la tienda debe contener un catálogo de los productos en venta y debe incluir un proceso para el manejo de pedidos, el inventario para satisfacer el pedido y un proceso de envío. También debe tener métodos de proceso y de recogida de pagos.

Los conceptos y las tareas involucradas en la creación de datos de tienda se tratan en la Parte 4, "Desarrollo de los datos de la tienda" en la página 37.

Arquitectura de la tienda

La arquitectura de la tienda de WebSphere Commerce consta de los siguientes componentes:

- WebSphere Commerce Server
- Instancia de WebSphere Commerce Server
- Configuraciones de la tienda

WebSphere Commerce Server

WebSphere Commerce Server es el servidor que maneja las funciones de tienda y las funciones relacionadas con el comercio de una solución de comercio electrónico. Los elementos del escaparate y la lógica de negocio de la oficina interna residen en una aplicación Web incluida en WebSphere Commerce Server. Puede utilizar la aplicación Web por omisión que proporciona WebSphere Commerce (WCS Stores) o puede crear la suya propia.

Una aplicación Web puede contener los elementos para una tienda o para varias tiendas. Cuando una aplicación Web contiene varios escaparates y oficinas internas, los elementos de cada tienda se separan en directorios de tienda (dirtienda).

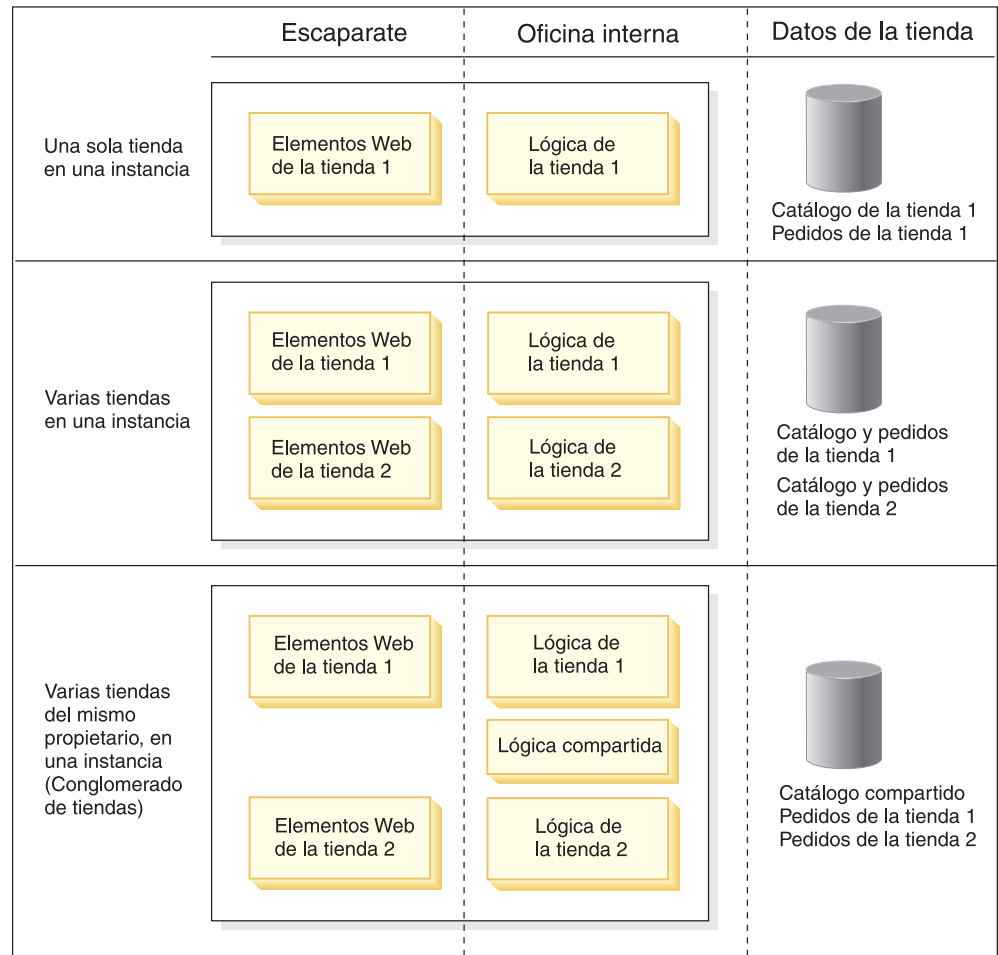
Instancia de WebSphere Commerce Server

Una instancia de WebSphere Commerce Server es una aplicación WebSphere Application Server con una base de datos asociada. Una instancia puede dar soporte a varias tiendas. Todas las tiendas de una instancia comparten la misma base de datos y pueden compartir algunos tipos de datos, por ejemplo, el catálogo, el despacho de pedidos o las recepciones. Todas las tiendas en una instancia también comparten el mismo contenedor EJB.

Configuraciones de la tienda

WebSphere Commerce da soporte a varias configuraciones de tienda. Es decir, con WebSphere Commerce puede crear una sola tienda en una instancia o puede crear varias tiendas en una instancia con escaparates, oficinas internas y datos de tienda diferentes. O puede crear varias tiendas en una instancia con escaparates diferentes y oficinas internas y catálogos compartidos. El siguiente diagrama ilustra algunas

configuraciones de tienda posibles.



Nota: Cada tienda tiene su propio identificador. La licencia de WebSphere Commerce establece un límite específico sobre el número de tiendas que puede crear. Puede adquirir licencias adicionales. Consulte el acuerdo de licencia para obtener información más detallada.

Capítulo 2. Desarrollo de la tienda

Este capítulo proporciona una visión general del proceso de desarrollo de la tienda en WebSphere Commerce.

Opciones de desarrollo de la tienda

WebSphere Commerce le proporciona varias opciones para desarrollar la tienda:

- Crear una tienda basándose en un ejemplo
- Crear una tienda desarrollando elementos de tienda nuevos
- Crear una tienda utilizando una combinación de elementos de tienda de ejemplo y de tienda nueva

Crear una tienda basada en un ejemplo

En WebSphere Commerce, el modo más rápido y fácil de crear una tienda en línea es copiar una de las tiendas de ejemplo que se proporcionan con WebSphere Commerce y, a continuación, personalizarla según sus necesidades.

Tiendas de ejemplo


WebSphere Commerce incluye varias tiendas en línea de ejemplo, totalmente funcionales, que se pueden utilizar como base para crear una tienda propia. Estos ejemplos, que incluyen tiendas de empresa a cliente y de empresa a empresa, implementan muchas de las características más utilizadas en los sitios de comercio electrónico más importantes actualmente y proporcionan todos los elementos de tienda necesarios. Para obtener más información sobre las tiendas en línea proporcionadas con WebSphere Commerce, consulte la ayuda en línea de WebSphere Commerce.


Por qué empezar con una tienda de ejemplo: Aunque con WebSphere Commerce es posible crear una tienda en línea totalmente nueva, si utiliza una copia de una de las tiendas de ejemplo como base para su propia tienda, podrá crearla y ponerla en funcionamiento con mucha más rapidez.

WebSphere Commerce necesita que se carguen determinados datos en la base de datos de WebSphere Commerce Server para crear una tienda operativa, y que estos datos se carguen en el orden determinado por el esquema. Dado que las tiendas de ejemplo incluyen todos los datos obligatorios en el orden y la estructura que necesita la base de datos de WebSphere Commerce Server, si utiliza una de dichas tiendas como base para su propia tienda, se ahorrará una cantidad de tiempo importante durante el periodo de creación inicial.

Después de crear una copia de una tienda de ejemplo, puede modificar muchos o pocos elementos, según sus necesidades. Por ejemplo, es posible que sólo tenga que editar los datos utilizando las herramientas disponibles con WebSphere Commerce y cambiar el aspecto de las páginas de la tienda utilizando WebSphere Commerce Studio. O es posible que tenga que editar los archivos XML o la base de datos directamente para efectuar cambios de más envergadura en los datos y volver a escribir las páginas de la tienda para cambiar las características y el flujo de la tienda. Para obtener más información sobre la modificación de tiendas, consulte el tema "*Cambiar los elementos de base de datos de la tienda*" en la ayuda en línea de WebSphere Commerce.

WebSphere Commerce también proporciona varias tiendas de referencia que están diseñadas para que se utilicen como ejemplos de código de las características destacadas. Una tienda de referencia es una tienda en línea que contiene código totalmente funcional para características seleccionadas de una tienda en línea como, por ejemplo, cupones. Hay tiendas de referencia disponibles en

 http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html

 http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Para obtener más información sobre cómo crear una tienda basándose en un ejemplo, consulte la ayuda en línea de WebSphere Commerce.

Crear una tienda desarrollando elementos de tienda nuevos

No todos los usuarios querrán crear su tienda a partir de una tienda de ejemplo. Por ejemplo, si el flujo de las páginas de tienda es significativamente diferente del de cualquiera de las tiendas de ejemplo proporcionadas o si piensa personalizar de forma significativa el esquema de base de datos de WebSphere Commerce Server, puede que desee crear la tienda desarrollando sus propios elementos de escaparate, de oficina interna y de datos de la tienda. Consulte “Herramientas de desarrollo de tienda” en la página 10 para obtener la lista de herramientas proporcionadas con WebSphere Commerce.

Crear una tienda utilizando una combinación de elementos de tienda de ejemplo y de tienda nueva

La utilización de una combinación de la tienda de ejemplo junto con el desarrollo de elementos de tienda nuevos puede constituir el método de desarrollo de tienda que funcione mejor en su caso. Por ejemplo, si algunos de los elementos de base de datos de una de las tiendas de ejemplo coinciden exactamente con las necesidades de la tienda, pero el flujo de las páginas de dicha tienda no coincide, puede copiar los elementos de base de datos de la tienda y personalizarlos mientras desarrolla elementos Web totalmente nuevos.

El archivador de tienda

Las tiendas de ejemplo que se incluyen con WebSphere Commerce se proporcionan en formato de archivador de tienda. Un archivo archivador de tienda (.sar) es un archivo comprimido (por ejemplo, un archivo ZIP) que contiene todos los elementos necesarios para crear una tienda. Se utiliza principalmente como vehículo para empaquetar y entregar tiendas en un formato que puede copiarse fácilmente y luego utilizarse como base sobre la cual se crearán tiendas nuevas. Sólo tiene que publicar un archivador de tienda en WebSphere Commerce Server para crear una tienda funcional que pueda ver, visitar y en la que pueda comprar.

Normalmente, un archivador de tienda está compuesto de los archivos siguientes:

- Elementos Web: los archivos utilizados para crear las páginas de tienda como, por ejemplo, archivos HTML, archivos JSP, imágenes, gráficos y archivos de inclusión. Los elementos Web se agrupan formando un archivo comprimido en el archivador de tienda.
- Paquete de recursos de propiedades (opcional): Contiene el texto para las páginas de la tienda. Si la tienda soporta más de un idioma, el paquete de recursos contendrá varios paquetes, uno por idioma.

- Elementos de datos de tienda: Los datos que se deben cargar en la base de datos. Los elementos de datos de tienda como las campañas, las entradas de catálogo, las monedas, la información de despacho de pedidos, de cálculo de precios, de envío, de la tienda y de los impuestos. Para obtener una lista más detallada de los elementos de datos de tienda, consulte la Parte 4, “Desarrollo de los datos de la tienda” en la página 37.

Los elementos de base de datos de tienda de los archivadores de tienda de ejemplo proporcionados con WebSphere Commerce tienen el formato de archivos XML válidos para Loader package. Los archivos XML del archivador de tienda están pensados para ser transportables y no deben contener claves primarias generadas que sean específicas de una instancia determinada de la base de datos. En lugar de ello, utilizan alias internos que ID Resolver resuelve en el momento en que se publica la tienda. La utilización de estos convenios permite copiar y publicar los archivadores de tienda de ejemplo varias veces. Para obtener más información, consulte el Apéndice B, “Creación de los datos” en la página 327.

Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225.

- Elementos de pago: Información de configuración para IBM Payment Manager.
- Un descriptor: Archivo XML, `sarinfo.xml`, que describe el archivador de tienda, incluidos los nombres del archivo archivador comprimido de elementos Web, de los paquetes de recursos y de los archivos XML de elementos de base de datos de la tienda. El archivo `sarinfo.xml` también contiene los nombres de los archivos de inclusión y de los archivos de comprobación de coherencia, así como información acerca del archivo archivador que se necesita durante el proceso de publicación. `sarinfo.xml` es el único archivo obligatorio en un archivador de tienda.

Nota: Los archivadores de las tiendas de ejemplo  ToolTech y NewFashion también incluyen los archivos siguientes:

- `tools_properties.zip`
- `tools_xml.zip`
- `runtime_xml.zip`

Los Servicios de tienda utilizan estos archivos para configurar las tiendas. Estos archivos no deben cambiarse, eliminarse ni copiarse a otras tiendas.

Archivadores de tienda de ejemplo

Un archivo archivador de tienda de ejemplo (`.sar`) es un archivador de tienda que está pensado para que se copie y se utilice como base para crear tiendas nuevas. Los archivadores de tienda de ejemplo incluyen unos cuantos convenios que les permiten ser copiados y publicados varias veces. Estos convenios incluyen lo siguiente:

- Ninguna referencia a claves externas o claves primarias generadas: los archivadores de tienda de ejemplo no contienen claves primarias generadas que sean específicas de una instancia determinada de la base de datos. En lugar de ello, utilizan alias internos que ID Resolver resuelve en el momento en que se publica la tienda. Para obtener más información, consulte el Apéndice B, “Creación de los datos” en la página 327.

Las tiendas de ejemplo proporcionadas con WebSphere Commerce son archivadores de tienda de ejemplo. Estas tiendas están disponibles en la lista **Ejemplos** de la página Crear archivador de tienda de los Servicios de tienda.

Determinar cuándo utilizar un archivador de tienda

El archivador de tienda está diseñado como un vehículo para empaquetar y entregar tiendas. Si desea utilizar su tienda como ejemplo para mostrar a otros, desplegarla en otro servidor o plataforma, o si desea utilizarla como base para crear otras tiendas, puede empaquetarla con el formato de archivador de tienda.

Es posible que también desee utilizar el archivador de tienda si su tienda se basa, en gran manera, en una de las tiendas de ejemplo y no necesita hacer muchos cambios en el archivador.

Si opta por utilizar el archivador de tienda y ha creado una tienda utilizando una tienda de ejemplo, su tienda ya tiene el formato de archivador de tienda. A continuación, puede mantener su tienda nueva en formato de archivador de tienda asegurándose que todos los cambios que efectúe durante el desarrollo de la tienda se reflejen en el archivador de tienda.

También puede empaquetar en formato de archivador de tienda, una tienda que haya creado utilizando otros métodos. Para obtener más información sobre cómo crear un archivador de tienda, consulte la Parte 6, "Empaquetado de la tienda" en la página 215.

Cuándo no utilizar un archivador de tienda

Quizá no desee utilizar el archivador de tienda si crea una sola instancia de la tienda o si está efectuando muchos cambios en el esquema de WebSphere Commerce existente. Los cambios en el esquema de WebSphere Commerce no están soportados por el archivador de tienda ni por las herramientas de desarrollo de la tienda de WebSphere Commerce, tal como se entrega el producto. Si desea mantener el archivador de tienda después de efectuar muchas modificaciones, póngase en contacto con el representante de IBM para obtener más información.

Herramientas de desarrollo de tienda

WebSphere Commerce proporciona varias herramientas para ayudarle a desarrollar la tienda. Las herramientas que utilice dependerán de cómo elija desarrollar y empaquetar la tienda.

Herramientas para desarrollar el escaparate

Para desarrollar los elementos del escaparate es posible que sea necesario personalizar las páginas de la tienda, sustituirlas por páginas propias, crear páginas nuevas o una combinación de estas tres posibilidades.


WebSphere Commerce proporciona las siguientes herramientas para crear o editar elementos de escaparate:

- **WebSphere Commerce Studio:** Commerce Studio incluye las herramientas necesarias para crear y editar los elementos del escaparate, que incluyen archivos HTML, gráficos, multimedia y archivos JSP (JavaServer Pages). Page Designer, que se incluye en Commerce Studio, le permite crear archivos HTML o JSP, así como imágenes animadas. También puede configurar WebSphere Commerce Studio para utilizar otra herramienta de desarrollo Web de su elección. Consulte la ayuda en línea de WebSphere Studio para obtener más información sobre cómo registrar sus propias herramientas con WebSphere Commerce Studio.

Si piensa trabajar con la tienda en formato de archivador de tienda, WebSphere Commerce Studio le permite importar los elementos Web del archivador de tienda a un proyecto de Studio, manteniendo intacta la estructura del archivador

de tienda. Después de realizar cambios en los archivos JSP, los archivos HTML y las imágenes utilizando las herramientas de Studio, puede volver a exportar los archivos al archivador de tienda de WebSphere Commerce Server y volver a publicar los elementos Web.

Si no desea mantener el archivador de tienda, puede publicar los archivos directamente en la tienda en funcionamiento utilizando WebSphere Commerce Studio.

- Servicios de tienda: El diálogo Elementos Web de los Servicios de tienda le permite sustituir el archivo archivador comprimido de elementos Web del archivador de tienda por otro conjunto de elementos Web o descargar los elementos Web existentes en una ubicación de su elección, donde puede editarlos con la herramienta de desarrollo Web que prefiera. Si está trabajando con la tienda en formato de archivador de tienda, puede utilizar el diálogo Elementos Web para volver a poner los elementos cambiados en el archivador de tienda. Las páginas de configuración de tienda le permiten habilitar o inhabilitar las distintas características en los archivos JSP de las tiendas publicadas. Actualmente, los Servicios de tienda sólo dan soporte a la configuración de las características de colaboración: Espacios de trabajo de colaboración y Atención al cliente. Estas características sólo están disponibles para configurarse en tiendas basadas en las tiendas de ejemplo  ToolTech y NewFashion.

Para obtener más información sobre el uso de herramientas en WebSphere Commerce Studio y los Servicios de tienda para crear y editar los elementos de escaparate, consulte la ayuda en línea de WebSphere Commerce. Para obtener más información sobre la creación de escaparates en WebSphere Commerce, consulte la Parte 2, “Desarrollo del escaparate de la tienda” en la página 15.

Herramientas para desarrollar los datos de la tienda

Tiene varias opciones para desarrollar y editar los elementos de base de datos en la tienda.

- Servicios de tienda

Los Servicios de tienda son un conjunto de herramientas basadas en navegador para trabajar en archivadores de tienda. Mediante los Servicios de tienda, puede crear rápidamente un archivador de tienda basándose en uno de los archivadores de ejemplo que se proporcionan con WebSphere Commerce. Una vez que ha creado un archivador de tienda, los Servicios de tienda le permiten realizar las tareas siguientes:

- Publicar el archivador de tienda para crear una tienda funcional.
- Cambiar los valores de impuestos utilizando el cuaderno Impuestos.
- Cambiar los valores de envío utilizando el cuaderno Envío.
- Cambiar los valores generales de tienda utilizando el cuaderno Perfil de tienda.

Los Servicios de tienda no le permiten editar todos los elementos de datos de tienda en el archivador de tienda. Para obtener una lista de los elementos que puede editar utilizando los Servicios de tienda, consulte el tema “Cambiar los elementos de base de datos de la tienda” de la ayuda en línea de WebSphere Commerce. Para editar otros elementos en el archivador de tienda, edite los elementos XML directamente.

Para obtener más información sobre el uso de los Servicios de tienda, consulte la ayuda en línea de WebSphere Commerce.

Cuándo utilizar los Servicios de tienda: Utilice los Servicios de tienda para copiar archivos de tienda de ejemplo y para editar los elementos de base de datos en formato de archivador de tienda.

- WebSphere Commerce Loader package

WebSphere Commerce Loader package consta principalmente de utilidades para preparar y cargar datos en una base de datos de WebSphere Commerce. Puede utilizar Loader package para cargar grandes cantidades de datos y actualizar datos de la base de datos de WebSphere Commerce. El programa de utilidad Loader de este paquete utiliza XML válido y con formato correcto como entrada para cargar datos en la base de datos. Los elementos del documento XML se correlacionan con los nombres de tabla de la base de datos y los atributos de elementos se correlacionan con las columnas.

Para obtener información sobre cómo utilizar Loader package para desarrollar y cargar elementos de datos, consulte la Parte 7, "Publicación de la tienda" en la página 225.

Cuándo se ha de utilizar WebSphere Commerce Loader package: utilice WebSphere Commerce Loader package para cargar por primera vez los elementos de base de datos en la base de datos de WebSphere Commerce y para actualizarlos.

Importante: si ha cambiado el esquema de base de datos, la única opción para cargar datos en la base de datos es utilizar Loader package.

- WebSphere Commerce Accelerator

WebSphere Commerce Accelerator es un área de trabajo con herramientas en línea que se utiliza principalmente para efectuar el mantenimiento de tiendas en línea mediante diversas operaciones de tienda. Sin embargo, puesto que WebSphere Commerce Accelerator le permite editar datos que ya están en la base de datos, puede utilizarlo como herramienta de desarrollo de tienda una vez ya ha realizado la primera inserción de datos en la base de datos, tanto si es con los datos de la tienda de ejemplo como si es con datos que usted haya creado. Para obtener una lista de los elementos que puede editar con WebSphere Commerce Accelerator, consulte el tema de la ayuda en línea de WebSphere Commerce "*Cambiar los elementos de base de datos de la tienda*".

Cuándo utilizar WebSphere Commerce Accelerator: Utilice WebSphere Commerce Accelerator después de que haya insertado datos en la base de datos de WebSphere Commerce.

- Editar la base de datos directamente

También tendrá siempre la opción de editar la base de datos directamente utilizando inserciones SQL.

Nota: SQL es un lenguaje específico de base de datos. Es posible que para Oracle se requiera una sintaxis de SQL diferente. Tenga en cuenta que las sentencias SQL tienen necesariamente valores específicos de base de datos y que las sentencias SQL quizá no puedan volverse a utilizar en otra instancia de WebSphere Commerce Server.

Herramientas para desarrollar la oficina interna

Las herramientas para desarrollar la oficina interna, incluido crear y ampliar mandatos, crear código personalizado e implementar la lógica de negocio, se describen en la publicación *IBM WebSphere Commerce, Guía del programador*.

El rol de Desarrollador de tiendas

Los desarrolladores de tienda desarrollan los tres tipos de elementos de tienda. Diseñan e implementan los elementos del escaparate, incluidos los archivos JSP y la oficina interna, y la creación de nuevos mandatos y del código personalizado que sea necesario. También crean los datos de la tienda y pueden modificar cualquier funcionalidad estándar que se incluya con WebSphere Commerce.

Los desarrolladores de tienda que creen el escaparate y los datos de la tienda deberán tener conocimientos de programación en Java, JavaScript, HTML y tecnología JSP, y estar familiarizados con la arquitectura de tienda, los datos de tienda y los archivadores de tienda de WebSphere Commerce.

Los Desarrolladores de tienda que creen la oficina interna deben tener conocimientos de programación en Java, JavaBeans, VisualAge para Java, programación J2EE y estar familiarizados con el modelo de programación y el modelo de objetos de WebSphere Commerce. La publicación *IBM WebSphere Commerce, Guía del programador* proporciona más información sobre cómo personalizar la oficina interna.

Los desarrolladores de tienda pueden trabajar con desarrolladores de base de datos y diseñadores Web. Los desarrolladores de base de datos modifican y amplían el esquema de base de datos de WebSphere Commerce con el propósito de implementar funciones de tienda personalizadas o integrar información de base de datos ya existente. Este miembro suele tener conocimientos de administrador de bases de datos para DB2 u Oracle.

Los diseñadores Web crean el aspecto del sitio y, junto con los desarrolladores de tiendas, las páginas de tienda. Los diseñadores Web deben tener experiencia en la utilización de herramientas multimedia, conocimientos de HTML y JavaScript, y estar familiarizados con la tecnología JSP.

Nota: Los roles de desarrollador de base de datos y diseñador Web no están definidos en WebSphere Commerce Server. Si fuera necesario, a los desarrolladores de base de datos y a los diseñadores Web se les debe asignar acceso de Desarrollador de tiendas.

Una vez que se ha creado un archivador de tienda, los desarrolladores de tiendas tienen autorización para realizar cambios en él manualmente o utilizando el cuaderno Perfil de tienda y los cuadernos Impuestos y Envío, pero no tienen autorización para publicar el archivador de tienda en WebSphere Commerce Server.

Parte 2. Desarrollo del escaparate de la tienda

Capítulo 3. Desarrollo del escaparate de la tienda

Este capítulo proporciona una visión general de la arquitectura de los escaparates de WebSphere Commerce, incluido el modo en que se muestra a los clientes la parte externa de la tienda, los elementos Web como, por ejemplo, las páginas HTML, los archivos JSP, las hojas de estilo, las imágenes, los gráficos y otros tipos de archivo multimedia.

Arquitectura del escaparate

WebSphere Commerce utiliza un sistema de *mandatos* y *vistas* para mostrar a los clientes los elementos Web de un escaparate.

- Los *mandatos* efectúan un proceso de negocio específico como, por ejemplo, añadir un producto al carro de la compra, procesar un pedido, actualizar el listín de un cliente o mostrar una página de producto específica. Cuando la acción finaliza, el mandato devuelve una vista.
- Las *vistas* muestran el resultado de los mandatos y las acciones de usuario, es decir, presentan las páginas de la tienda (archivos JSP) a los clientes. Para que la vista invoque a un archivo JSP, el nombre del archivo JSP debe estar registrado con la vista en la tabla de registro de vistas (VIEWREG). El archivo JSP correspondiente se almacena utilizando el nombre del archivo JSP en el subdirectorío de la tienda (dirienda), bajo el subdirectorío raíz de documentos de la aplicación Web WCS Stores.

Tanto a los mandatos como a las vistas se les invoca utilizando los URL. Por ejemplo, cuando un cliente pulsa **Carro de la compra** en la tienda de ejemplo, el cliente invoca el URL `https://nombresistpral/vía/OrderItemDisplay?`, que se pasa a WebSphere Commerce Server. Éste llama al mandato `OrderItemDisplay` y la página de carro de la compra se muestra al cliente.

Cuando un cliente pulsa **Ayuda** en la tienda de ejemplo, el cliente invoca el URL `https://nombresistpral/vía/HelpView?`, que se pasa a WebSphere Commerce Server. Éste llama a `HelpView`, que devuelve la página de ayuda.







WebSphere Commerce Server también puede correlacionar varios mandatos con un URL, lo cual permite que cada tienda tenga, opcionalmente, su propia implementación de ese mandato.

De forma parecida, WebSphere Commerce Server también le permite correlacionar varios archivos JSP con una sola vista, en la que cada tienda puede, opcionalmente, registrar distintos nombres de archivo JSP para distintos tipos de dispositivo.

Nota: Los mandatos para mostrar productos y categorías devuelven vistas, así como nombres de archivos JSP. Estos nombres de archivos JSP que muestran productos y categorías están almacenados en los datos de catálogo. Para obtener más información, consulte “Visualización de los elementos de catálogo de la tienda” en la página 77. Opcionalmente, puede asignar distintos nombres de archivo JSP para que muestren productos y categorías para cada grupo de miembros o idioma soportado por la tienda.

Mandatos y vistas por omisión

WebSphere Commerce proporciona mandatos y vistas por omisión que puede utilizar en la tienda. Encontrará una lista de estos mandatos y vistas en el archivo `wcs.bootstrap.xml`. Los archivos de rutina de carga se encuentran en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\schema\xml
-  unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml
-  /usr/WebSphere/CommerceServer/schema/xml
-  /opt/WebSphere/CommerceServer/schema/xml
-  /opt/WebSphere/CommerceServer/schema/xml
-  /qibm/proddata/WebCommerce/schema/xml

Si necesita un mandato o una vista que no se proporciona, puede crearlo. Para obtener información sobre la creación de mandatos y vistas, consulte la publicación *IBM WebSphere Commerce, Guía del programador*.

Creación de las páginas de tienda

La tarea más extensa cuando se crea el escaparate es crear las páginas de la tienda. Antes de empezar el trabajo de desarrollo en las páginas de la tienda, debe completar las actividades planificadas siguientes:

- Desarrollar una lista de las páginas de tienda que se necesitan
- Desarrollar una lista de los URL de mandatos y vistas
- Asociar nombres de archivo JSP a las vistas

Desarrollar una lista de las páginas de la tienda

Para desarrollar una lista de las páginas necesarias para crear la tienda, debe conocer los requisitos funcionales y de negocio de la tienda, así como los procesos de negocio que se hayan definido.

Trabajar con ejemplos de uso

Muchas personas recopilan los requisitos en forma de ejemplos de uso. Los ejemplos de uso definen los procesos de negocio de la tienda, en forma de interacciones entre el cliente y el sistema propuesto. En el caso de una tienda en línea, los ejemplos de uso pueden definir cómo se registra un cliente en la tienda, cómo examina el catálogo o cómo hace el pedido de un artículo.

En la ayuda en línea se proporciona un conjunto de ejemplos de uso, que detallan los procesos de negocio para las tiendas de ejemplo. Estos ejemplos de uso pueden ayudarle a comprender con más exactitud el flujo de las tiendas de ejemplo y se pueden utilizar como guía si desea crear ejemplos de uso para su propia tienda.

A continuación se muestra un ejemplo de uso de Registro:

Ejemplo de uso de Registro: El proceso de registro permite a los clientes entrar información personal en la base de datos.

Actor:

- Cliente

Flujo principal: El cliente selecciona **Regístrate** en el marco lateral. A continuación, el sistema muestra una página con los campos siguientes:

- Correo electrónico
- Contraseña
- Verificar contraseña
- Nombre
- Apellido
- Edad (opcional)
- Sexo (opcional)

El cliente entra la información adecuada en los campos anteriores y selecciona **Someter**. El sistema crea un nuevo cliente y guarda la información del cliente (E1, E2, E3). El sistema solicita al cliente que gestione su cuenta siguiendo el proceso que se describe en el Ejemplo de uso de Gestionar cuenta personal

Flujos alternativos: Ninguno.

Flujos de excepción: E1: La dirección de correo electrónico ya existe:

- Si la dirección de correo electrónico ya existe en el sistema, se muestra un mensaje de error solicitando al usuario que escriba otra dirección de correo electrónico. El ejemplo de uso se reanuda desde el principio.

E2: Faltan campos obligatorios:

- Si no se ha cumplimentado alguno de los campos siguientes (Dirección de correo electrónico, Contraseña, Verificar contraseña, Nombre, Apellido), el sistema emite un mensaje de error. El ejemplo de uso se reanuda desde el principio.

E3: Contraseña no válida:

- Si la contraseña no es válida o no coincide con la contraseña de verificación, el sistema emite un aviso.

Determinar el flujo de compra de la tienda: Tanto si desarrolla ejemplos de uso para ilustrar los procesos de negocio de la tienda como si usa otro método, cuando los procesos de negocio están disponibles se puede crear el flujo de compra de su tienda.

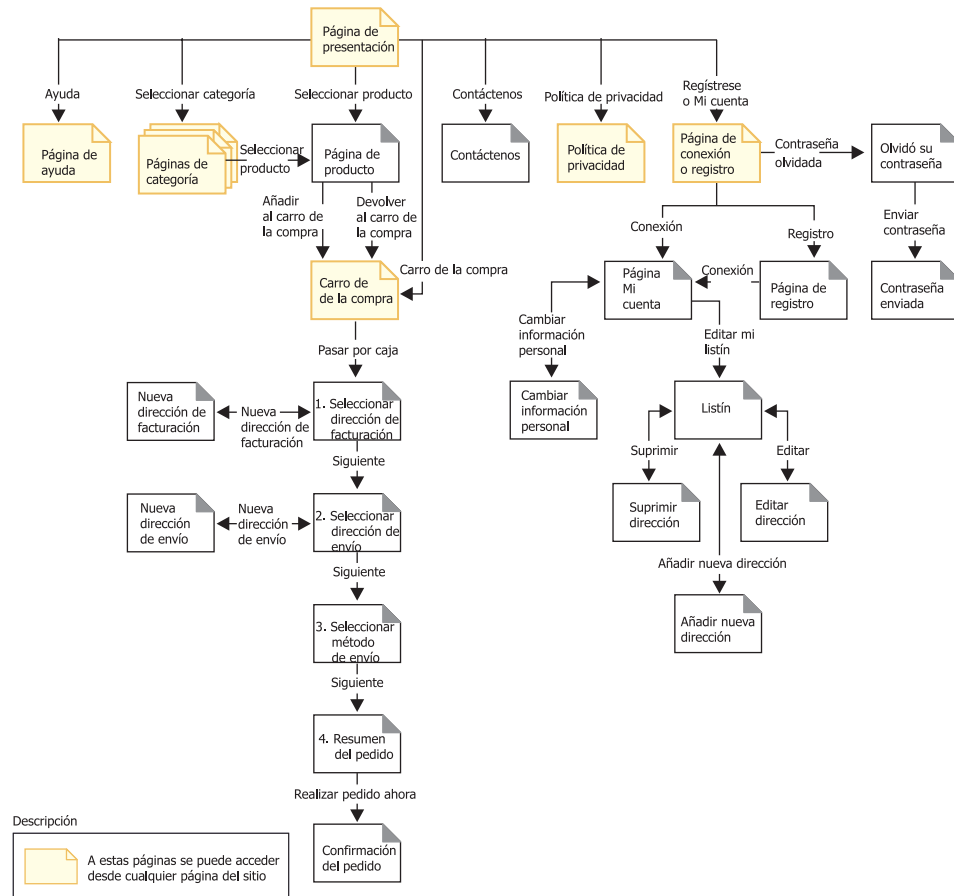
Nota: Puesto que los ejemplos de uso a menudo contienen información sobre el flujo, del tipo, "Si el cliente selecciona **Someter**, se muestra la página Pedido", los ejemplos de uso pueden proporcionar información muy útil para la creación de diagramas de flujo de compra.

El flujo de compra refleja los requisitos y los procesos de negocio definidos para la tienda, que ilustran los movimientos de un cliente en la tienda. Por ejemplo, un cliente puede entrar en su sitio Web por la página de presentación y le puede solicitar que se registre antes de examinar el catálogo, o puede elegir que los clientes puedan ver el catálogo como invitados, sin tener que registrarse. Algunos flujos de compra permiten a los clientes pasar por una "caja rápida", mientras que otros flujos de compra requieren que el cliente realice todos los pasos de pasar por caja cada vez que efectúa una compra. O bien, su flujo de compra puede ofrecer a los clientes las dos opciones de pasar por caja.



Para comprobar que el diagrama de flujo de la tienda está completo, asegúrese de que todos los pasos en los ejemplos de uso de su tienda están ilustrados en el diagrama de flujo de la tienda.

La correlación visual del flujo de compra, como se hace en el siguiente diagrama del flujo de compra de la tienda de ejemplo InFashion, permite ver cómo se moverán los clientes en la tienda.



El diagrama del flujo de compra de InFashion es muy sencillo. Aunque incluye el flujo principal del recorrido de un cliente por la tienda, no incluye las situaciones de error. Por ejemplo, lo que sucede en caso de que el cliente se conecte utilizando una contraseña incorrecta o escriba un número de tarjeta de crédito no válido. Sin embargo, incluso un diagrama sencillo como éste le permite desarrollar una lista de las páginas necesarias para la tienda. Para empezar, deberá crear una vista para cada página que se lista en el diagrama de flujo de compra.

Por ejemplo, si fuera a crear una tienda con el mismo flujo de compra que en el diagrama de InFashion, debería crear las páginas siguientes:

Nota: La tabla siguiente lista los nombres de vista utilizados para la tienda InFashion

Páginas del diagrama de flujo de compra de la tienda InFashion (según las ve el cliente)	Vista correspondiente
Página de presentación	StoreCatalogDisplayView
Página de ayuda	HelpView
Contáctenos	ContactView
Política de privacidad	PrivacyView
Página de conexión o registro	LogonForm

Páginas del diagrama de flujo de compra de la tienda InFashion (según las ve el cliente)	Vista correspondiente
Olvidó su contraseña	LogoffView
Contraseña enviada	ResetPasswordForm
Página Mi cuenta	LogonForm
Cambiar información personal	UserRegistrationForm
Listín	AddressBookForm
Añadir nueva dirección	AddressForm
Suprimir dirección	AddressBookForm
Editar dirección	AddressForm
Página de registro	UserRegistrationForm
Carro de la compra	OrderItemDisplayViewShiptoAssoc
Seleccionar dirección de facturación	OrderItemDisplayViewShiptoAssoc
Nueva dirección de facturación	OrderItemDisplayViewShiptoAssoc
Seleccionar dirección de envío	OrderItemDisplayViewShiptoAssoc
Nueva dirección de envío	AddressForm
Seleccionar método de envío	OrderItemDisplayViewShiptoDsp
Resumen del pedido	OrderDisplayPendingView
Confirmación del pedido	OrderOKView

Nota: Muchas de las vistas utilizadas en InFashion se crearon específicamente para InFashion. Estas vistas se listan en el archivo `command.xml` en el archivador de tienda de InFashion. Para obtener más información, consulte “Registro de mandatos, vistas y URL en WebSphere Commerce” en la página 51.

La tabla anterior sólo muestra el conjunto básico de páginas que tiene que crear. Para determinar las demás páginas que debe crear, puede observar detenidamente los ejemplos de uso u otros métodos utilizados para definir los procesos de negocio.

Páginas de error: Los flujos de excepción de los ejemplos de uso también pueden ayudarle a determinar las páginas de error que tiene que crear para la tienda. El ejemplo de uso del registro para InFashion especifica los siguientes flujos de excepción:

- La dirección de correo electrónico ya existe: Si la dirección de correo electrónico ya existe en el sistema, éste muestra un mensaje de error solicitándole al usuario que entre otra dirección de correo electrónico. El ejemplo de uso se reanuda desde el principio.
- Faltan campos obligatorios: Si no se ha cumplimentado alguno de los campos siguientes (Dirección de correo electrónico, Contraseña, Verificar contraseña, Nombre, Apellido), el sistema emite un mensaje de error. El ejemplo de uso se reanuda desde el principio.
- Contraseña no válida: Si la contraseña no coincide con la contraseña de verificación, el sistema emite un aviso.

Como consecuencia, deberá crear una página de error o un mensaje de error para cada flujo de excepción.

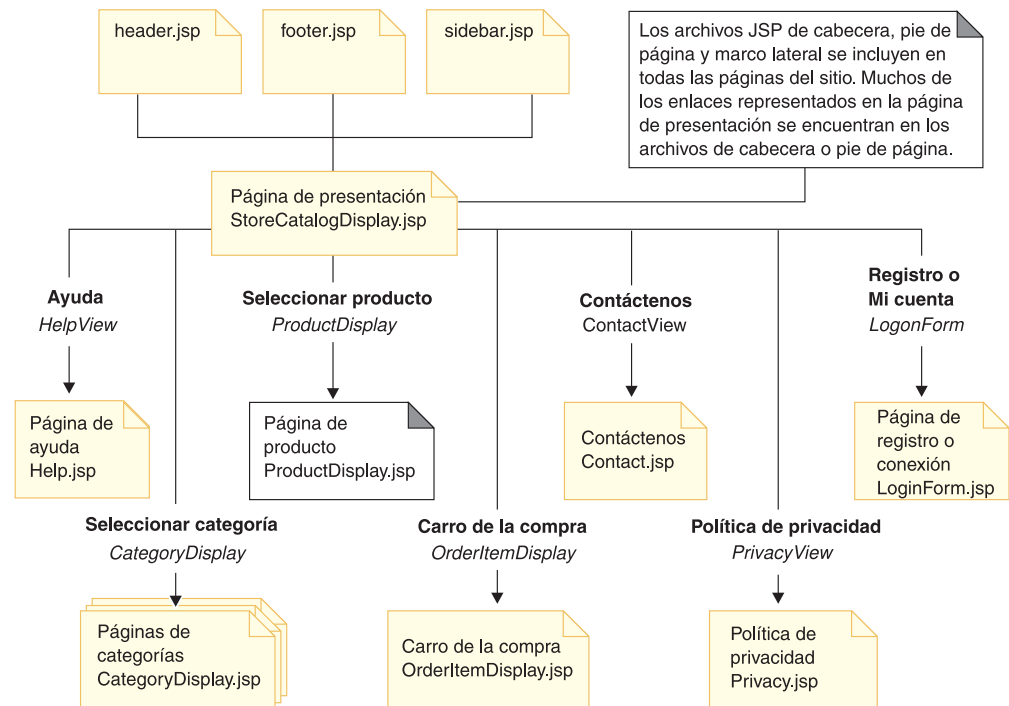
Desarrollar una lista de los URL de mandatos y vistas

Como se ha visto en el diagrama de flujo de compra de InFashion, los procesos de negocio como el registro o pasar por caja, pueden requerir varias páginas. Para combinar estas páginas en un flujo o proceso de negocio operativo, en lugar de que sean solamente un conjunto de páginas, debe incluir mandatos y vistas en sus páginas.

Desarrollar una lista de los URL necesarios

Así como ha desarrollado una lista de las páginas que se necesitan para crear la tienda, también debe desarrollar una lista de los URL de mandato y de vista necesarios para implementar los procesos de negocio para su tienda. Utilizando el diagrama de flujo de compra para la tienda y la lista de mandatos y vistas por omisión, identifique los URL necesarios para completar cada acción.

Conocer los URL de mandatos y de vistas que se utilizan en las tiendas de ejemplo puede ayudarle a determinar los URL que necesita en su tienda. La ilustración siguiente identifica los URL para algunas de las acciones que se muestran en el diagrama de flujo de compra de la tienda InFashion. Para obtener más detalles, consulte la información sobre tiendas de ejemplo en la ayuda en línea de WebSphere Commerce.



Leyenda

	Estas páginas pueden accederse desde cualquier página del sitio.
Negrita	Acción del cliente
<i>Cursiva</i>	URL

Asociar nombres de archivo JSP a las vistas

WebSphere Commerce Server utiliza mandatos de vista para componer una vista como respuesta a una petición de cliente. WebSphere Commerce Server proporciona los siguientes mandatos de vista:

- `HttpForwardViewCommandImpl`: Este mandato de vista reenvía la petición de vista a un archivo JSP.
- `HttpRedirectViewCommandImpl`: Este mandato de vista reenvía la petición de vista a otro URL.
- `HttpDirectViewCommandImpl`: Este tipo de mandato de vista envía la vista de respuesta directamente al cliente. No llama a un archivo JSP. Las vistas directas permiten que los mandatos de controlador generen respuesta de salida (en lugar del mandato de vista).

Utilice el mandato de vista `HttpForwardViewCommandImpl` para representar directamente los archivos JSP. Por ejemplo, en el diagrama que muestra los URL que se utilizan en la tienda InFashion, para visualizar la página de ayuda (`Help.jsp`), se registra `HelpView` en el registro de vistas y se asocia a `Help.jsp` y al mandato `HttpForwardViewCommandImpl`. Esto se muestra en el ejemplo siguiente:

```
<viewreg
viewname="HelpView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
properties="docname=help.jsp"
internal="0"
https="0"
/>
```

Tenga en cuenta que se utiliza el nombre de clase (`classname`) totalmente calificado para la interfaz y la clase de implementación.

Nota: En este ejemplo, el URL que invoca la vista no está restringido. Es decir, cualquiera puede acceder al URL directamente. Si utiliza esta técnica, asegúrese de que el archivo JSP sólo muestra datos públicos.

Utilice el mandato de vista `HttpForwardViewCommandImpl` para representar las vistas que devuelve un mandato de visualización. Un mandato de visualización lee los datos de la base de datos pero no los modifica. Por ejemplo, en el diagrama que ilustra los URL utilizados en InFashion, el mandato `OrderItemDisplay` devuelve la vista `OrderItemDisplayViewShiptoAssoc`. Durante el registro de esta vista en el registro de vistas, se asocian a la misma `OrderItemDisplay.jsp` y `HttpForwardViewCommandImpl`. Esto se muestra en el ejemplo siguiente:

```
<viewreg
viewname="OrderItemDisplayViewShiptoAssoc"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
properties="docname=OrderItemDisplay.jsp"
internal="0"
https="0"
/>
```

Nota: En este ejemplo, el URL que invoca la vista está restringido. Es decir, sólo aquellos con el acceso designado pueden invocar directamente el URL. El acceso a esta vista está controlado por aquel que tenga acceso al URL para el mandato.

Debe asociar un nombre de archivo JSP a cada vista asociada a todo mandato de visualización que va a utilizar (por ejemplo, `OrderItemDisplay`). Para obtener más información acerca de cómo asociar nombres de archivo JSP con vistas, consulte el apartado “Registro de mandatos, vistas y URL en WebSphere Commerce” en la página 51.

Nota: `ProductDisplay` y `CategoryDisplay` listan el nombre de archivo JSP asociado en los datos del catálogo en lugar de en el registro de vistas.

Utilice el mandato de vista `HttpRedirectViewCommandImpl` para representar las vistas que devuelve un mandato que modifica la base de datos. Para utilizar la vista de redirección, especifique el nombre de vista utilizando el parámetro `&URL=` en el URL. Por ejemplo, cuando añada información sobre la dirección en `AddressForm` de la tienda de ejemplo `InFashion` y pulsa **Someter**, llama al mandato `AddressAdd`. El URL que se utiliza para invocar el mandato `AddressAdd` especifica `AddressBookForm` como la vista mediante el parámetro `&URL=`. El resultado es que se redirige a la vista `AddressBookForm`. Durante el registro de la vista `AddressBookForm` en el registro de vistas, se asocian al mismo `AddressBookForm.jsp` y `HttpForwardViewCommandImpl`.

Debe utilizar la técnica `URL=parámetro` para todos los mandatos que no son de visualización. Los mandatos que no son de visualización son mandatos que generan cambios en los datos de la base de datos.

Parte 3. Visión general de los datos de la tienda

Capítulo 4. Datos de la tienda

Este capítulo proporciona una visión general de la arquitectura de datos de la tienda de WebSphere Commerce Server y de los elementos de datos que crean una tienda. El modelo de información de WebSphere Commerce Server también se trata en este capítulo.

¿Qué son los datos de la tienda?

Los datos de la tienda es la información que se carga en la base de datos de WebSphere Commerce Server y que permite el funcionamiento de la tienda. Para que una tienda funcione correctamente debe tener los datos en una ubicación que dé soporte a todas las actividades de los clientes. Por ejemplo, para que un cliente efectúe una compra, la tienda debe contener un catálogo de los productos en venta (datos de catálogo), los datos asociados al proceso de los pedidos (datos sobre impuestos y envío) y el inventario para satisfacer la petición (datos de inventario y de despacho de pedidos).

Modelo de información de los datos de la tienda

Esta guía utiliza un modelo de información para ilustrar cómo se estructuran los datos de la tienda en WebSphere Commerce Server. El modelo de información de WebSphere Commerce Server es un resumen de alto nivel de la información que contienen los modelos de objetos y de datos de WebSphere Commerce Server. El modelo de información destaca las características más importantes de los modelos de datos y de objetos, pero no incluye los detalles específicos de las implementaciones del objeto y el esquema.

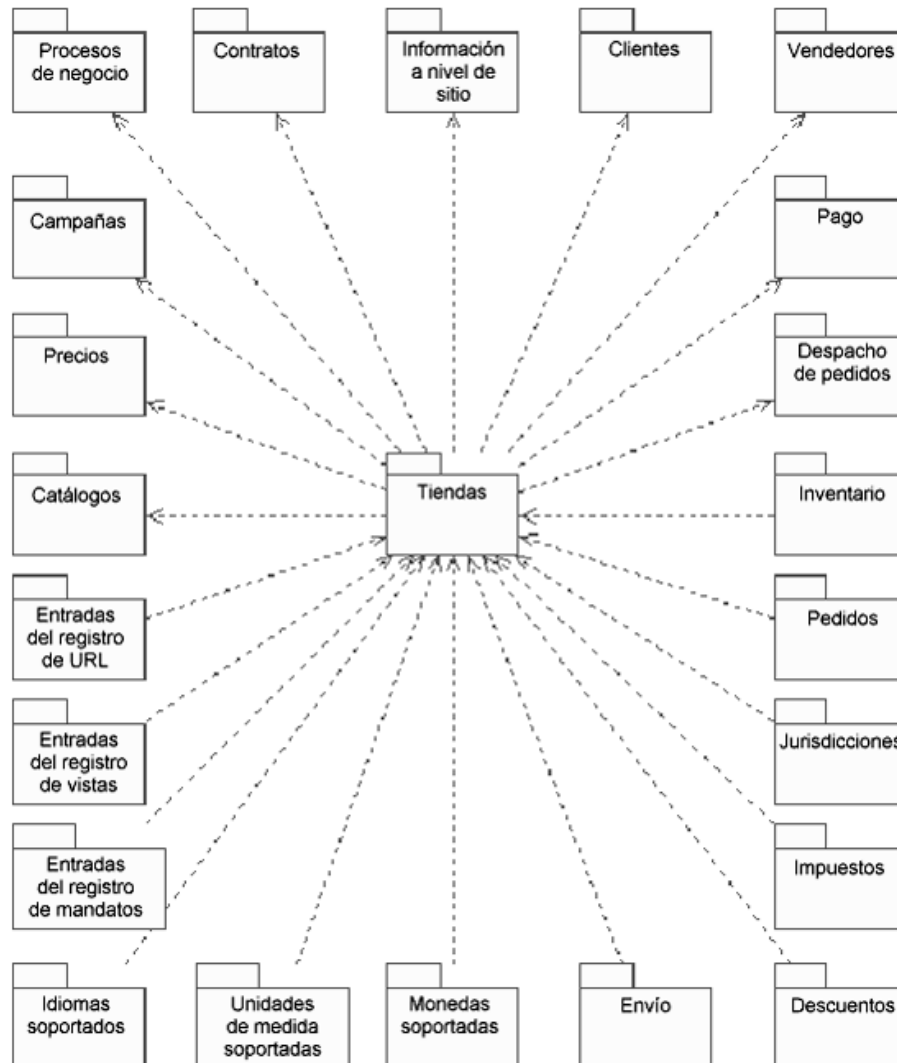
Por ejemplo, ciertas tablas y objetos en los modelos de datos u objetos que contienen datos de relaciones entre entidades (como, por ejemplo, las parejas de clave externa) no se muestran como entidades en el modelo de información. En su lugar, estas relaciones de entidades están implementadas por las líneas de relaciones entre entidades en los modelos de información. El modelo de información tampoco ilustra las extensiones de *detalles* (atributos adicionales de datos de una entidad que se almacenan en una tabla aparte como resultado de consideraciones de implementación: por ejemplo, la descripción de los productos es una extensión de la entidad del producto que se almacena por separado). Para obtener más información sobre los objetos de relaciones y las extensiones de detalles, consulte el modelo de objeto en la ayuda en línea de WebSphere Commerce.



Para obtener más información sobre los modelos de objetos y de datos de WebSphere Commerce, consulte la ayuda en línea de WebSphere Commerce.

Elementos de datos de la tienda

El diagrama siguiente ilustra los elementos de datos de una tienda de WebSphere Commerce.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

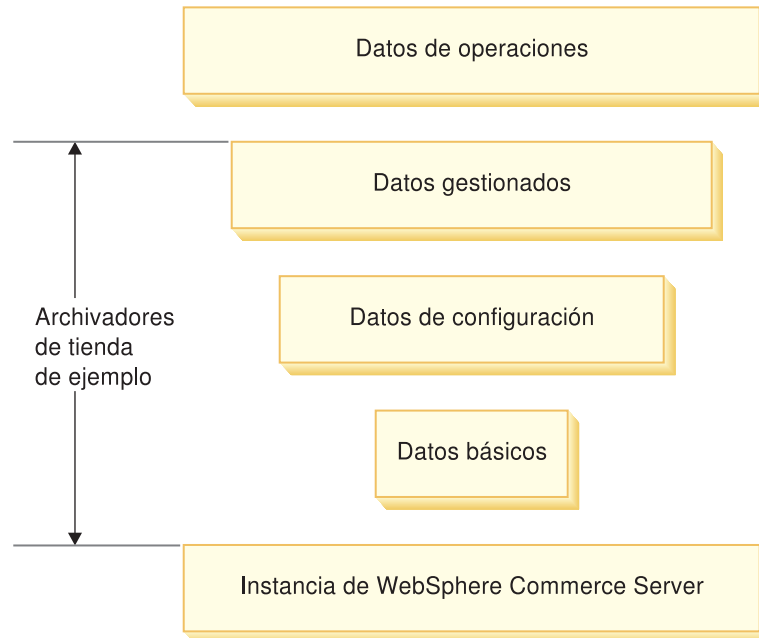
Observe las dos direcciones de las flechas del diagrama. En algunos casos, la flecha apunta a la tienda, por ejemplo, de moneda a tiendas. En este caso, los elementos de moneda son exclusivos de esta tienda en particular y forman parte de la tienda. Estas son las monedas a las que da soporte esta tienda. Si se suprime la tienda, también se suprimirá la lista de monedas que soporta.

Cuando una flecha va de las tiendas a un elemento, por ejemplo, el catálogo, otras tiendas pueden compartir ese elemento. Varias tiendas pueden compartir un catálogo. Sin embargo, si se suprime la tienda, el catálogo no se suprime.

Cada uno de los elementos de datos ilustrados en el diagrama anterior se trata en profundidad en los capítulos contenidos en la Parte 4, "Desarrollo de los datos de la tienda" en la página 37.

Arquitectura de datos de la tienda

Los datos de las tiendas de WebSphere Commerce se ajustan a la arquitectura que se muestra en el siguiente diagrama. Cada uno de los elementos de datos de la tienda ilustrados en el diagrama del apartado "Elementos de datos de la tienda" en la página 28, puede clasificarse como perteneciente a uno o más de los tipos de datos de tienda ilustrados a continuación.



Instancia de WebSphere Commerce Server

El nivel más básico de datos se encuentra en la instancia de WebSphere Commerce Server. Cuando se crea una instancia, los archivos de rutina de carga que tienen formato XML, llenan la base de datos con información. Los archivos de rutina de carga crean los siguientes tipos de datos:

- los tipos de usos de cálculo, tipos de dispositivo, (navegadores, correo electrónico, I-Mode, etc.), roles y tipos de mensajes
- el ID de administración por omisión, WCSADMIN
- los mandatos, URL y vistas por omisión
- las políticas de negocio por omisión
- los idiomas y las monedas soportadas por la instancia
- la organización por omisión, que se puede utilizar como el propietario de la tienda
- la organización por omisión del sitio
- el grupo de tiendas por omisión

Esta información está disponible para todas las tiendas que hay en esa instancia, y se identifica como Información a nivel de sitio en el diagrama que se muestra en "Elementos de datos de la tienda" en la página 28.

Para obtener información más detallada sobre los archivos de rutina de carga y las tablas de base de datos que rellenan, consulte la ayuda en línea de WebSphere Commerce.

Datos básicos

El siguiente nivel de datos de tienda son los datos básicos. Estos datos crean los datos imprescindibles para la tienda, incluyendo:

- El identificador de tienda en la tabla STOREENT. Esto crea una tienda en la base de datos.
- El contrato por omisión.
- El identificador de tienda en las tablas de base de datos de contratos.
- El identificador de miembro de la organización propietaria de la tienda según las tablas de base de datos de contrato
- El directorio de tienda en la tabla STORE. El directorio de la tienda es el directorio en el que se encuentran los elementos web de la tienda.
- El apodo o identificador de la dirección de la tienda en la tabla STADDRESS. El apodo es exclusivo para cada tienda.

Si utiliza los Servicios de tienda para crear una tienda, esta información se crea automáticamente con el nuevo archivador de tienda. Los Servicios de tienda le permiten seleccionar la organización por omisión que actuará como propietaria de la tienda; también puede crear una organización que actúe como propietaria, utilizando la Consola de administración. Si no utiliza los Servicios de tienda para crear una tienda, tendrá que cargar esta información en la base de datos utilizando Loader package o editar la base de datos directamente.

Los datos de la tienda del diagrama que se muestra en “Elementos de datos de la tienda” en la página 28 son datos básicos.

Datos de configuración

Los datos de configuración controlan la ejecución del servidor de comercio. La ejecución de servidor común proporciona un entorno en el que se despliegan y se ejecutan las aplicaciones de comercio. El entorno de trabajo se compone del modelo de programación, del modelo de proceso, del manejo de excepciones, del control de transacciones, del acceso a datos y del modelo de persistencia. La ejecución de servidor común aprovecha los servicios de ejecución proporcionados por WebSphere Application Server para dar soporte a las aplicaciones de WebSphere Commerce Server. Los datos de configuración determinan los mandatos, vistas y archivos JSP que utilizará su tienda para mostrar las páginas de la tienda

Los siguientes elementos de datos identificados en el diagrama que se muestra en “Elementos de datos de la tienda” en la página 28 están clasificados como Datos de configuración:

- Entradas del registro de mandatos
- Entradas del registro de vistas
- Entradas del registro de URL

Datos gestionados

Los datos gestionados son los datos que crea el vendedor y que los clientes sólo pueden leer en el sitio del vendedor. Puesto que el vendedor tiene el control absoluto sobre el estado de estos datos, los datos gestionados pueden administrarse mediante un sistema de gestión de contenido.

Los siguientes elementos de datos identificados en el diagrama que se muestra en “Elementos de datos de la tienda” en la página 28 están clasificados como Datos gestionados:

- Campañas
- Políticas de negocio
- Contratos
- Centros de despacho de pedidos
- Jurisdicciones
- Impuestos
- Descuentos
- Envío
- Monedas
- Unidades de medida
- Idiomas
- Catálogos
- Precios
- Clientes
- Vendedores
- Pago

Datos de operaciones

Los datos de operaciones son los datos que los clientes del sitio crean o modifican (directa o indirectamente) como resultado de sus interacciones con el sitio. Por ejemplo, los pedidos de los clientes se consideran datos de operaciones, al igual que los niveles de inventario, que suben y bajan según las operaciones de la tienda. Los clientes también se consideran datos de operaciones. Asimismo, los datos creados por el vendedor pueden ser datos de operaciones.

Puesto que los cambios de los datos de operaciones no están bajo el control absoluto del vendedor, no tendría sentido administrar estos datos mediante un sistema de gestión de contenido.

Los siguientes elementos de datos identificados en el diagrama que se muestra en “Elementos de datos de la tienda” en la página 28 están clasificados como Datos de operaciones:

- Pedidos
- Inventario
- Despacho de pedidos
- Clientes

Nota: En algunas instancias, la línea que separa los datos de operaciones de los datos gestionados puede ser difícil de determinar. Por ejemplo, en una tienda, los datos de clientes y de contratos pueden considerarse datos gestionados, mientras que en otra tienda, el mismo tipo de datos puede ser considerado de operaciones. La primera tienda puede gestionar los datos de los clientes y sus contratos relacionados porque tienen un conjunto específico de clientes (es decir, los clientes no se pueden registrar en línea). Sin embargo, la segunda tienda permite a los clientes registrarse en línea y crear información de los contratos en línea.

Un segundo ejemplo trata de los datos de catálogo. En un sitio de un solo vendedor, los datos del catálogo se consideran datos gestionados. En un sitio de un centro de compras, los datos del catálogo pueden considerarse datos de operaciones.

En algunos sitios, ciertos registros del mismo tipo de datos pueden considerarse gestionados, mientras que otros registros se consideran de operaciones. Por ejemplo, el contrato por omisión puede considerarse como datos gestionados, pero los contratos específicos negociados en línea pueden formar parte de los datos de operaciones.

Arquitectura de datos de la tienda y las tiendas de ejemplo

Las tiendas de ejemplo que se proporcionan con WebSphere Commerce incluyen la mayoría de los tipos de datos de la arquitectura de datos de la tienda. Por ejemplo, para poder crear una tienda utilizando una tienda de ejemplo o para poder publicar una tienda de ejemplo, debe haber una instancia de WebSphere Commerce Server. Entonces, cuando crea una tienda basándose en una tienda de ejemplo y utilizando las herramientas de los Servicios de tienda, se crean los datos básicos. Las tiendas de ejemplo incluyen todos los datos de configuración necesarios y la mayor parte de los datos gestionados requeridos para una tienda en funcionamiento. Al crear tiendas basadas en determinadas tiendas de ejemplo, es posible que se le indique que realice algunos pasos de configuración de datos, utilizando las herramientas de WebSphere Commerce Accelerator.

Herramientas para la creación de datos

WebSphere Commerce proporciona varias herramientas para crear y manipular los datos de su tienda. Estas herramientas se listan a continuación:

WebSphere Commerce Loader package

Loader package consta principalmente de utilidades para preparar y cargar datos en una base de datos de WebSphere Commerce. Para obtener más información, consulte la Parte 7, “Publicación de la tienda” en la página 225.

Servicios de tienda

Los Servicios de tienda editan los datos antes de la publicación en formato de archivador de tienda, en lugar de editar los datos mismos en la base de datos. Los Servicios de tienda también le permiten publicar todos los elementos de datos de la tienda en la base de datos. Para obtener más información, consulte la Parte 7, “Publicación de la tienda” en la página 225.

Consola de administración

La Consola de administración le permite controlar el sitio o la tienda realizando operaciones administrativas y tareas de configuración. También puede utilizar la Consola de administración para crear nuevas organizaciones y usuarios, así como para asignar usuarios a roles (Desarrollador de tiendas, Administrador de tienda, Administrador de sitio, etc.). La Consola de administración también le permite identificar los tipos de mensajes y notificaciones que estarán disponibles en la tienda.

WebSphere Commerce Accelerator

WebSphere Commerce Accelerator consta de diversas herramientas en línea que le permiten crear y efectuar el mantenimiento de varios elementos de la tienda. Se

puede crear y gestionar una gran parte de los datos de la tienda utilizando las herramientas de WebSphere Commerce Accelerator. Sin embargo, en algunos casos, para poder crear datos mediante WebSphere Commerce Accelerator, ciertos datos deben haberse ya cargado en la base de datos utilizando las herramientas de publicación de los Servicios de tienda o Loader package. Para obtener más información, consulte el “Diagrama de herramientas y datos de la tienda”.

Consola de administración de la organización

La Consola de administración de la organización le permite gestionar las organizaciones que acceden al sitio o a la tienda. La Consola de administración de la organización también permite al administrador del comprador gestionar a los compradores de la organización.

Diagrama de herramientas y datos de la tienda

El diagrama siguiente lista las herramientas que puede utilizar para crear cada tipo de datos.

Herramientas para crear datos	Datos básicos	Datos de configuración	Datos gestionados	Datos de operaciones
WebSphere Commerce Loader package	Utilice Loader package para cargar datos básicos en forma de un archivo XML. Para obtener más información, consulte “Creación de elementos de datos de tienda en un archivo XML” en la página 45.	Utilice Loader package para cargar datos de configuración en forma de un archivo XML. Para obtener más información, consulte “Creación de un archivo XML para registrar mandatos, vistas y URL” en la página 51.	Utilice Loader package para cargar datos gestionados en forma de un archivo XML. Para obtener más información, consulte los capítulos correspondientes a los elementos de datos gestionados.	En general, los datos operativos no se pueden cargar con Loader package.

Herramientas para crear datos	Datos básicos	Datos de configuración	Datos gestionados	Datos de operaciones
Servicios de tienda (solamente para datos no publicados todavía con formato de archivador de tienda)	Cuando crea un nuevo archivador de tienda utilizando los Servicios de tienda, se crean los datos básicos. Para obtener más información sobre el uso de los Servicios de tienda, consulte la ayuda en línea de WebSphere Commerce.	No aplicable.	Los Servicios de tienda le permiten crear y editar partes de los siguientes elementos gestionados: <ul style="list-style-type: none"> • Jurisdicciones • Impuestos • Envío • Moneda • Idiomas Para obtener más información sobre qué partes de los elementos de base de datos se pueden editar o crear con los Servicios de tienda, consulte la ayuda en línea de WebSphere Commerce, "Cambiar los elementos de base de datos de la tienda".	No aplicable.
Consola de administración	Utilice la Consola de administración para crear una organización que actúe como propietaria de la tienda.	No aplicable.	No aplicable.	No aplicable.

Herramientas para crear datos	Datos básicos	Datos de configuración	Datos gestionados	Datos de operaciones
WebSphere Commerce Accelerator	No aplicable.	No aplicable.	<p>Utilice WebSphere Commerce Accelerator para crear o editar los siguientes datos:</p> <ul style="list-style-type: none"> • Campañas • Contratos (debe haber un contrato por omisión en la base de datos para poder utilizar las herramientas de Gestión de relaciones comerciales de WebSphere Commerce Accelerator para crear contratos adicionales o cambiar los ya existentes. Utilice Loader package o los Servicios de tienda para crear un contrato por omisión en la base de datos) 	<p>Los clientes crean datos de operaciones cuando se registran en la tienda o efectúan compras. Sin embargo, en algunos casos, puede utilizar WebSphere Commerce Accelerator para realizar pedidos para un cliente o para crear una devolución.</p> <p>WebSphere Commerce Accelerator también le permite gestionar el inventario</p>

Herramientas para crear datos	Datos básicos	Datos de configuración	Datos gestionados	Datos de operaciones
WebSphere Commerce Accelerator (continuación)	No aplicable.	No aplicable.	<ul style="list-style-type: none"> • Despacho de pedidos • Descuentos • Catálogos (debe haber un catálogo maestro en la base de datos para poder utilizar las herramientas de Gestión de productos de WebSphere Commerce Accelerator y así crear un catálogo por el que puedan navegar los clientes y que permita añadir o cambiar la información de los productos. Utilice Loader package o los Servicios de tienda para crear un catálogo maestro en la base de datos.) • Precios 	No aplicable.
Consola de administración de la organización	No aplicable.	No aplicable.	No aplicable.	Los clientes y compradores se crean cuando entran en la tienda. Sin embargo, utilizando la Consola de administración de la organización puede aprobar los compradores o crear de nuevos.

Parte 4. Desarrollo de los datos de la tienda

Los capítulos de esta sección explican cada uno de los elementos de datos de la tienda con más detalle. Estos elementos de datos se han organizado según la estructura de la arquitectura de datos de la tienda de WebSphere Commerce:

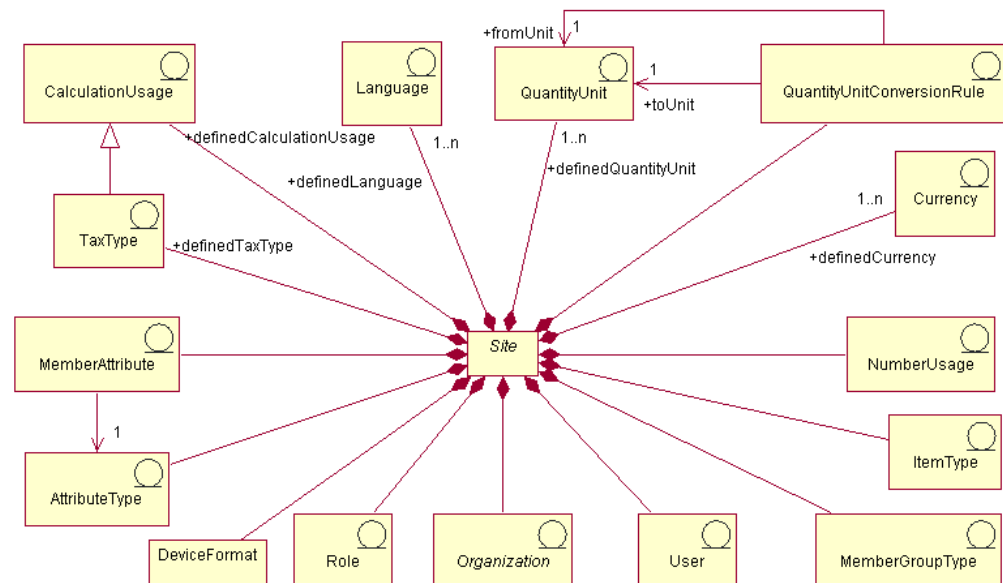
- Instancia del servidor WebSphere Commerce
 - Sitio
- Datos básicos
 - Tienda
- Datos de configuración
 - Registro de mandatos
 - Ver registro
 - Registro de URL
- Datos gestionados
 - Elementos compartidos
 - Catálogo
 - Precios
 - Contratos (incluidas las políticas de negocio)
 - Despacho de pedidos
 - Campañas
 - Pago
 - Exclusivo de los elementos de tienda
 - Idiomas soportados
 - Monedas soportadas
 - Unidades de medida soportadas
 - Jurisdicciones
 - Envío
 - Impuestos
 - Descuentos
- Datos de operaciones
 - Inventario
 - Pedidos
 - Clientes

Capítulo 5. Elementos de sitio

Cada instancia de WebSphere Commerce Server tiene su propia base de datos de información relacional. Los archivos de rutina de carga crean una instancia e insertan información en las tablas de base de datos, después de que se cree el esquema. Una vez que se han cargado los datos, puede ver la información cargada previamente en la tabla de base de datos apropiada. Muchas de las tablas de base de datos contienen información, a nivel de tienda o de grupo de tiendas, específica de una tienda o grupo de tiendas. Normalmente, esta información la gestionan los administradores de tienda. Sin embargo, algunas tablas contienen información que representa las funciones a nivel de sitio de WebSphere Commerce que todas las tiendas de la instancia pueden utilizar y que gestiona el administrador de sitio de WebSphere Commerce. En este capítulo se describen estas funciones. Para obtener más información sobre los archivos de rutina de carga, consulte la ayuda en línea de WebSphere Commerce. Para obtener información acerca de los elementos específicos de tienda, consulte el Capítulo 6, "Elementos de tienda" en la página 43.

Los elementos de sitio en WebSphere Commerce

El diagrama siguiente ilustra los tipos de datos que contiene el sitio y su relación con el sitio.



Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, "Descripción de UML" en la página 325. Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce. Para obtener más información sobre el modelo de información, consulte "Modelo de información de los datos de la tienda" en la página 27.

Idioma

Un sitio puede tener muchos *idiomas* definidos en la tabla LANGUAGE y éstos pueden estar descritos en la tabla LANGUAGEDS. Normalmente, cada tienda da soporte a un subconjunto de estos idiomas añadiendo filas a la tabla STORELANG. Los diez idiomas definidos previamente son: alemán, chino tradicional y chino simplificado, japonés, coreano, italiano, francés, español, portugués de Brasil e inglés.

Atributos de miembros

Los *atributos de los miembros* se almacenan en la tabla MBRATTR y representan el conjunto de nombres de atributo definidos para los que se pueden almacenar valores para organizaciones o usuarios. Algunos ejemplos de nombres de atributo son JobFunction, ProcurementCard, SpendingLimit, ReferredBy y CountryOfOperation. Los valores de los atributos para organizaciones o usuarios específicos se pueden almacenar en la tabla MBRATTRVAL, y estos valores pueden ser distintos para diferentes tiendas o grupos de tiendas.

Tipos de atributo

Los *tipos de atributo* se almacenan en la tabla ATTRTYPE y representan los tipos de datos definidos que se pueden utilizar para representar los valores de los atributos. Algunos ejemplos de tipos de datos son INTEGER, STRING y FLOAT.

Tipos de grupos de miembros

Los *tipos de grupos de miembros* se almacenan en la tabla MBRGRPTYPE y representan el conjunto de usos del grupo de miembros definido. Los usos se asignan a los grupos de miembros añadiendo filas a la tabla MBRGRPUSG. Algunos ejemplos de usos de grupo de miembros son AccessGroup (para usar con las políticas de control de acceso) y UserGroup (para propósitos generales como, por ejemplo, grupos de clientes).

Usuario

El *usuario* representa la identidad autenticada del usuario. Normalmente, los usuarios representan a clientes formalizando o aprobando pedidos en nombre de organizaciones compradoras, agentes de venta procesando los pedidos para las organizaciones vendedoras o efectuando el mantenimiento de elementos a nivel de tienda, o administradores de sitio efectuando el mantenimiento de la instancia de WebSphere Commerce Server. Cada usuario está asociado a un sitio y definido en la tabla USERS.

Organización

La *organización* representa a organizaciones y unidades de organización dentro de las organizaciones. Normalmente, las organizaciones representan a las entidades de negocio responsables de comprar y vender. Los pedidos que formalizan los clientes en una organización compradora de empresa a empresa, se registran como si las formalizara la organización compradora. Las tiendas, los catálogos y los centros de despacho de pedidos son propiedad de las organizaciones responsables de ciertos aspectos de venta. Las organizaciones están definidas en la tabla ORGENTITY.

Rol

El *rol* representa el conjunto de roles definidos que se pueden asignar a los usuarios dentro de las organizaciones. Por ejemplo, a un usuario se le puede asignar el rol de representante de atención al cliente dentro de una organización

vendedora, o el rol de aprobador de compradores dentro de una organización compradora. Los nombres y las descripciones de los roles por omisión se insertan en la tabla ROLE. Para obtener más información sobre roles específicos, consulte la ayuda en línea de WebSphere Commerce.

Conversión de unidades de cantidad

Todos los sitios tienen *conversiones de cantidad*. Estas conversiones representan operaciones de multiplicación o división que se utilizan para la conversión entre distintas unidades de medida. Se almacenan en la tabla QTYCONVERT.

Unidades de cantidad

Las *unidades de cantidad* representan el conjunto de unidades de medida para el sitio. Están definidas en la tabla QTYUNIT y descritas en la tabla QTYUNITDSC. Cada tienda puede especificar cómo se redondea y formatea cada unidad de medida para su visualización, según el uso que se le quiera dar, añadiendo filas a la tabla QTYFORMAT.

Tipos de impuestos

Los *tipos de impuestos* representan usos de cálculo para calcular impuestos. Los impuestos sobre la venta y los impuestos sobre el envío son dos métodos de cálculo distintos que calculan impuestos. Los tipos de impuestos están definidos en la tabla TAXTYPE.

Uso de cálculo

Los *usos de cálculo* representan los diferentes tipos de cálculos que puede efectuar el mandato OrderPrepare. Los usos de cálculo se definen para los descuentos, costes de envío, impuestos sobre la venta, impuestos sobre el envío y cupones de venta electrónica. Los usos de cálculo están definidos en la tabla CALUSAGE.

Moneda

Todo sitio tiene un número de *monedas* definido en la tabla SETCURRE y descrito en la tabla SETCURRDSC. Cada tienda da soporte a un subconjunto de estas monedas añadiendo filas a la tabla CURLIST, una fila para cada moneda soportada.

Utilización de números

La *utilización de números* representa cómo se van a utilizar los números. Las tiendas pueden especificar distintas normas de redondeo y formateo de los números que se visualizan, según la forma en que se utilicen. Por ejemplo, una tienda puede redondear los precios por unidad con cuatro decimales especificando la utilización de "precio unitario", y redondear otros importes en dos decimales especificando la utilización "por omisión". La utilización de números se define en la tabla NUMBRUSG y se describe en la tabla NUMBRUSGDS.

Tipos de artículo

Los *tipos de artículo* representan las diferentes clases de artículos. Los dos tipos de artículo base de WebSphere Commerce son kits dinámicos y artículos normales. Los tipos de artículos están definidos previamente en la tabla ITEMTYPE. Para obtener más información sobre los artículos base, consulte el Capítulo 21, "Elementos de inventario" en la página 179.

Formatos de dispositivo

Los *formatos de dispositivo* se almacenan en la tabla DEVICEFMT y representan los diversos tipos de formato de dispositivo que utiliza un sitio como, por ejemplo, navegadores, I_MODE, correo electrónico, MQXML y MQNC. Todos estos tipos de dispositivo permiten a los usuarios interactuar con el sitio a través de diferentes medios.

Nota: Para algunos de los elementos de sitio, como por ejemplo, el idioma, la moneda, la unidad de la cantidad y la norma de conversión de la unidad de la cantidad, el Administrador de sitio puede ampliar las posibilidades a nivel de sitio añadiendo filas a las tablas adecuadas. Para otros, pueden necesitarse personalizaciones relacionadas para ampliar las posibilidades a nivel de sitio que representan. Si un Administrador de sitio ha añadido una nueva utilización de números para mostrar subtotales con un símbolo de moneda personalizado, el programa que muestra los subtotales deberá personalizarse para especificar la nueva utilización de números para el valor de subtotal cuando se formateen los importes de subtotal para su visualización.

Creación de elementos de sitio en WebSphere Commerce

Al crear una instancia en WebSphere Commerce Server se crean los elementos de sitio. Para obtener más información sobre cómo crear una instancia en WebSphere Commerce Server, consulte el Capítulo 5 "Creación o modificación de una instancia", de la publicación *IBM WebSphere Commerce, Guía de instalación*.

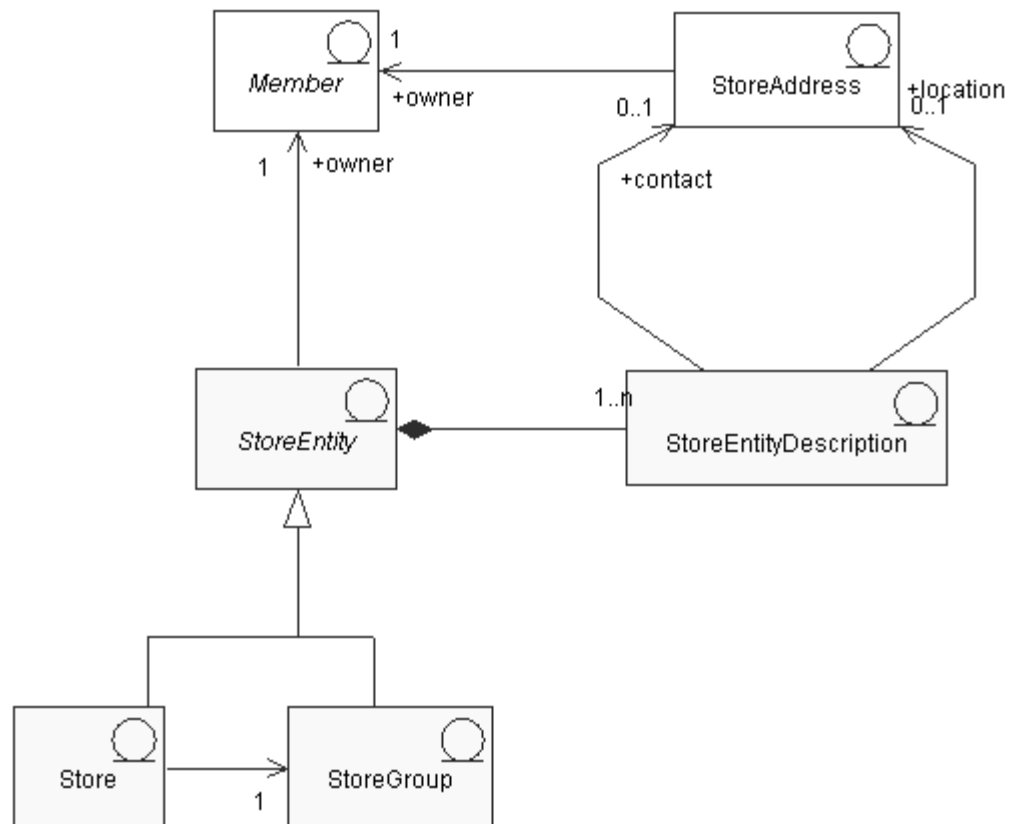
Capítulo 6. Elementos de tienda

Para crear una tienda en WebSphere Commerce, primero debe crear los siguientes elementos en la base de datos:

- la tienda
- el grupo al que pertenece
- el objeto abstracto de entidad de tienda que representa, de forma dual, una tienda o grupo de tiendas.

Los elementos de tienda en WebSphere Commerce

El diagrama siguiente ilustra los elementos de tienda en WebSphere Commerce Server.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte "Modelo de información de los datos de la tienda" en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, "Descripción de UML" en la página 325.

Entidad de tienda

Una *entidad de tienda* es una superclase abstracta que puede representar una tienda o un grupo de tiendas.

Una entidad de tienda tiene un propietario (un miembro). Para obtener más información sobre los miembros, consulte el Capítulo 23, "Elementos de cliente y vendedor" en la página 189.

Descripción de la entidad de tienda

La *descripción de la entidad de tienda* describe la entidad de tienda. Una entidad de tienda puede tener una descripción. Si la tienda da soporte a varios idiomas, la descripción de la entidad de tienda puede hacerse en varios idiomas. La descripción puede incluir una dirección de contacto para la entidad de tienda y también una dirección de ubicación para la entidad de tienda.

Tienda

Una *tienda* es una entidad de tienda. Toda tienda debe pertenecer a un grupo de tiendas.

Grupo de tiendas

Un *grupo de tiendas* es un conjunto de tiendas. Un grupo de tiendas es una entidad de tienda. El grupo de tiendas actúa como contenedor de la información, que se puede almacenar a nivel de grupos de tienda y que pueden compartir todas las tiendas del grupo de tiendas. Por ejemplo, las tiendas del mismo grupo de tiendas pueden compartir información como categorías de impuestos, idiomas soportados, monedas soportadas, códigos de cálculo y jurisdicciones de envío.

Actualmente, sólo puede haber y mantenerse un grupo de tiendas a nivel de administración de sitio dentro de WebSphere Commerce Server.



Para obtener información más detallada sobre la estructura de los elementos de tienda en WebSphere Commerce Server, consulte los modelos de datos y de objeto de tienda en la ayuda en línea de WebSphere Commerce.

Creación de elementos de tienda en WebSphere Commerce

Las herramientas de los Servicios de tienda de WebSphere Commerce le permiten crear o editar los siguientes elementos de tienda:

- el identificador de tienda y el identificador de miembro en los elementos de contacto
- el identificador de tienda en la tabla STOREENT
- el directorio de tienda en la tabla STORE
- el apodo de la dirección en la tabla STADDRESS
- la descripción de tienda
- la dirección de la tienda

Nota: Las herramientas de los Servicios de tienda trabajan con archivos XML con datos preinsertados, en forma de archivador de tienda.

Por esto, tiene dos opciones para crear elementos de tienda:

- Editar los elementos de tienda ya existentes de una de las tiendas de ejemplo que se proporcionan con WebSphere Commerce o de un archivador de tienda ya existente.

- Crear elementos de tienda con el formato de un archivo XML que se puede publicar como parte de un archivador de tienda o que se puede cargar utilizando Loader package.

Para obtener más información sobre la edición de elementos de tienda de un archivador de tienda ya existente, consulte la ayuda en línea de WebSphere Commerce. Para obtener información sobre la creación de elementos de tienda en forma de archivo XML, consulte el apartado “Creación de elementos de datos de tienda en un archivo XML”.

Creación de elementos de datos de tienda en un archivo XML







Cree los elementos de tienda en formato de archivos XML que se puedan cargar en la base de datos utilizando Loader package. Si crea una tienda multicultural, deberá crear archivos XML para cada entorno nacional al que la tienda dé soporte. El archivo específico de entorno nacional debe especificar toda la información de las descripciones, para que se pueda traducir fácilmente. Para obtener más información sobre el paquete Loader, consulte la Parte 7, “Publicación de la tienda” en la página 225.

Las tiendas de ejemplo, de las cuales se toman muchos de los ejemplos de estas tareas, utilizan un archivo `store.xml` para toda la información que no es necesario traducir y otro archivo `store.xml` para cada entorno nacional que la tienda soporta, con la información que debe traducirse. Los archivos específicos del entorno nacional contienen toda la información de las descripciones.

Para crear elementos de tienda, haga lo siguiente:

1. Revise los archivos XML que se utilizan para crear elementos de tienda para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda se encuentran en el directorio siguiente:







-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye dos archivos `store.xml`, que incluyen la información sobre la tienda. Para ver los archivos `store.xml` en el archivador de tienda, descomprímalo utilizando un programa ZIP. Los archivos `store.xml` se encuentran en el directorio de datos. El archivo `store.xml` específico de un idioma está situado en el subdirectorío específico del entorno nacional del directorio de datos.

2. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
3. Cree un archivo `store.xml`, copiando uno de los archivos `store.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más

información, consulte el archivo DTD que corresponde a store.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Cree una entidad de tienda.

- a. Utilizando el ejemplo siguiente como muestra, defina una entidad de tienda en su archivo XML para la tabla STOREENT.

```
<storeent
  storeent_id="@storeent_id_1"
  member_id="%MEMBER_ID"
  type="S"
  identifier="ToolTech"
  setcurr="EUR"
/>
```

donde

- storeent_id es una clave exclusiva generada.
- member_id es el propietario de la entidad de tienda.
- type es el tipo de entidad de tienda: G = GrupoTiendas, S = Tienda.
- identifier es una serie que, junto con el propietario, identifica de forma exclusiva la entidad de tienda.
- setcurr es la moneda por omisión de una entidad de tienda; es decir, la moneda que utilizará un cliente que todavía no tiene una moneda preferida. Si es NULL para una tienda, la moneda por omisión se obtiene de su grupo de tiendas.

5. Cree una dirección de tienda.

- a. Utilizando el ejemplo siguiente como muestra, cree la dirección o direcciones de tienda en el archivo XML para la tabla STADDRESS. Si crea un catálogo multicultural, tendrá que incluir esta información en un archivo XML específico de entorno nacional.

```
<staddress
  staddress_id="@staddress_id_es_ES_1"
  member_id="%MEMBER_ID"
  nickname="storeaddress_Spanish"
  address1="Avenida Meridiana 12xx"
  address2="Suite 9xx"
  businesstitle="ToolTech"
  city="Hospitalet"
  state="Barcelona"
  zipcode="0xx1x"
  country="España"
  phone1="34-900-555-1234"
  fax1="34-900-555-4321"
  email1="info@tooltech.xxx"
/>
```

donde

- staddress_id es una clave exclusiva generada.
- member_id es el propietario de la entidad de tienda.

6. Cree una descripción de la entidad de tienda.

- a. Utilizando el ejemplo siguiente como muestra, cree la descripción de la entidad de tienda en el archivo XML para la tabla STOREENTDS. Si crea un catálogo multicultural, tendrá que incluir esta información en un archivo XML específico de entorno nacional.

```
<storeentds
  description="Entidad de tienda de modelos de comercio"
  language_id="&es_ES"
  displayname="ToolTech"
  storeent_id="@storeent_id_1"
  staddress_id_cont="@staddress_id_es_ES_1"
  staddress_id_loc="@staddress_id_es_ES_1"
```

donde

- `description` es una descripción más larga de la entidad de tienda, adecuada para mostrarla a los clientes.
- `language_id` es el idioma por omisión para la información que se muestra a los clientes que compran en la tienda.
- `displayname` es una descripción breve de la entidad de tienda, adecuada para mostrarla a los clientes.
- `storeent_id` es la entidad de tienda.
- `staddress_id_cont` es la dirección de contacto de la entidad de tienda.
- `staddress_id_loc` es la ubicación física de la entidad de tienda.

7. Cree una tienda en la base de datos.

- a. Utilizando el ejemplo siguiente como muestra, defina una tienda en su archivo XML para la tabla STORE.

```
<store
  store_id="@storeent_id_1"
  directory="ToolTech"
  ffmcenter_id="@ffmcenter_id_1"
  language_id="&es_ES"
  storegrp_id="-1"
  allocationgoodfor="43200"
  bopmpadfactor="0"
  defaultboffset="2592000"
  ffmselectionflags="0"
  maxboffset="7776000"
  rejectedordexpiry="259200"
  rtnffmctr_id="@ffmcenter_id_1"
  pricererefflags="0"
  storetype="B2B"
/>
```

donde

- `store_id` es una clave exclusiva generada.
- `directory` es el directorio en el que se encuentran los elementos Web específicos de la tienda. La ubicación real de estos elementos en el sistema de archivos se basa en el valor de esta columna, más varios parámetros de configuración del archivo de configuración de WebSphere Commerce: `StoresDocRoot`, `StoresWebPath` y `StoresPropertiesPath`. Por ejemplo, si `StoresDocRoot` es `D:\WebSphere\wcs\stores`, `StoresWebPath` es `web`, `StoresPropertiesPath` es `properties` y el valor de esta columna es `mitienda`, los archivos JSP estarán situados en el directorio `D:\WebSphere\wcs\stores\web\mitienda` y los archivos de propiedades estarán situados en `D:\WebSphere\wcs\stores\properties\mitienda`.

- `ffmcenter_id` es el centro de despacho de pedidos por omisión para la tienda.
 - `language_id` es el idioma por omisión para la información que se muestra a los clientes que compran en la tienda.
 - `storegrp_id` es el grupo de tiendas al que está asociada la tienda. Este número se genera en la tabla `STOREGRP`.
 - `allocationgoodfor` significa que el planificador `ReleaseExpiredAllocations` puede utilizarse para anular asignaciones de inventario ATP cuando haya transcurrido el número de segundos especificado desde que se efectuaron las asignaciones.
 - `bopmpadfactor` indica si esta tienda calcula los importes de pedido (como impuestos o costes de envío) de forma diferente para los distintos centros de despacho de pedidos; el importe de un pedido sometido previamente puede cambiar cuando se asignen finalmente centros de despacho a los artículos de pedido en espera. Este factor de relleno representa el porcentaje que puede aumentar el importe de pedido presentado a `Payment Manager`, si fuera necesario. Por ejemplo, especifique 5 para permitir un incremento de hasta el 5 por ciento.
 - `defaultboffset` se utiliza cuando no se puede determinar una fecha estimada de disponibilidad para un artículo de pedido en espera; en ese caso, en el futuro se establecerá en este número de segundos.
 - `maxboffset` significa que, si el tiempo de disponibilidad estimado para un artículo de pedido en espera va a exceder normalmente este número de segundos en el futuro, se establecerá en este número de segundos en el futuro.
 - `rejectedordexpiry` son los pedidos con el pago en estado Denegado por un tiempo superior a este número de segundos y son candidatos para ser cancelados.
 - `rtnffmctr_id` es el centro de despacho de pedidos por omisión para devolver mercancía a la tienda.
 - `pricerefflags` contiene distintivos de bit que controlan en qué Acuerdos comerciales (`TradingAgreements`) y Ofertas (`Offers`) se buscará cuando la implementación por omisión del mandato de tarea `GetContractUnitPrices` renueve los precios:
 - 1 = `usePreviousOnly` - Utilizar aquellos a los que hacen referencia los artículos de pedido. Da error si ya no se pueden utilizar.
 - 2 = `usePreviousOrSearchAgain` - Igual que `usePreviousOnly`, pero en lugar de dar error cuando ya no se pueden utilizar, buscar en los que están guardados en las tablas `ORDIOFFER` y `ORDITRD`
 - 4 = `alwaysSearchAgain` - Buscar siempre en los que están guardados en las tablas `ORDIOFFER` y `ORDITRD`.
 - `storetype` indica uno de los siguientes tipos de tienda; lo utiliza una interfaz de usuario que proporciona las funciones adecuadas según el tipo de tienda: B2B = Empresa a empresa. B2C = Empresa a consumidor.
8. Defina un idioma soportado para la tienda.
- a. Utilizando el ejemplo siguiente como guía, defina un idioma soportado para la tienda en su archivo XML para añadir información a la tabla `STORELANG`. Si la tienda da soporte a varios idiomas, debe incluir esta información en un archivo XML específico del entorno nacional (uno para cada idioma soportado por la tienda).

```
<storelang
  language_id="&es_ES"
  storeent_id="@storeent_id_1"
/>
```

donde

- language_id es el idioma soportado por la entidad de tienda.
 - storeent_id es la entidad de tienda.
- b. Utilizando el ejemplo siguiente como muestra, añade información sobre el idioma a la tabla STORELANGDS. Si la tienda da soporte a varios idiomas, debe incluir esta información en un archivo XML específico del entorno nacional (uno para cada idioma soportado por la tienda).

```
<storlangds
  description="Español"
  language_id="&es_ES"
  storeent_id="@storeent_id_1"
  language_id_desc="& es_ES"
/>
```

donde

- description es una descripción breve del idioma, adecuada para mostrarla a los clientes en una lista de selección.
- language_id es el idioma de la descripción.
- storeent_id es la entidad de tienda que da soporte al idioma.
- language_id_desc es el idioma que se describe.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Capítulo 7. Datos de registro de los mandatos, vistas y URL

Cuando crea una instancia de WebSphere Commerce Server, los mandatos, las vistas y los URL que se proporcionan con WebSphere Commerce se registran en la base de datos de WebSphere Commerce Server, en las tablas correspondientes: CMDREG, VIEWREG y URLREG. Estos mandatos, vistas y URL están disponibles para que los utilicen todas las tiendas que residen en la instancia.

WebSphere Commerce también proporciona archivos JSP por omisión para mostrar las vistas por omisión. Estos archivos JSP están asociados a las vistas en la tabla VIEWREG.

Si crea mandatos, vistas o URL nuevos, o personaliza alguno ya existente, debe registrarlos en las tablas de base de datos correspondientes (CMDREG, VIEWREG y URLREG) para que puedan utilizarse en la tienda. Si crea nuevos archivos JSP para utilizarlos en la tienda, debe asociarlos a la vista correspondiente en la tabla VIEWREG.

Nota: Si crea un nuevo archivo JSP, pero le da el mismo nombre que el archivo JSP por omisión asociado a la vista, no es necesario registrar el nuevo archivo JSP en la tabla VIEWREG.

Para obtener más información sobre la creación o personalización de mandatos, vistas o URL, consulte la publicación *IBM WebSphere Commerce, Guía del programador*. La *Guía del programador* también contiene información sobre cómo y cuándo registrar mandatos, vistas, URL y archivos JSP.



Para obtener información más detallada sobre la estructura de elementos de mandato y de vista en WebSphere Commerce Server, consulte los modelos de datos de mandatos y vistas en la ayuda en línea de WebSphere Commerce.

Registro de mandatos, vistas y URL en WebSphere Commerce

Si crea o personaliza varios mandatos, vistas, URL o archivos JSP nuevos para la tienda, quizá desee registrarlos utilizando un archivo XML, que puede cargar posteriormente en la base de datos utilizando Loader package, o como parte de un archivador de tienda que se puede publicar mediante los Servicios de tienda. Para obtener más información sobre el paquete Loader, consulte la Parte 7, "Publicación de la tienda" en la página 225.

Nota: Antes de crear un archivo XML para cargar mandatos nuevos o personalizados, consulte la *Guía del programador* para obtener más detalles sobre cómo funcionan los mandatos.

Creación de un archivo XML para registrar mandatos, vistas y URL

Para crear un archivo XML para registrar los mandatos, vistas y archivos JSP nuevos para la tienda, haga lo siguiente:

1. Revise los archivos XML que se utilizan para registrar mandatos, vistas y archivos JSP para las tiendas de ejemplo. Cada tienda de ejemplo incluye un

archivo `command.xml`, que contiene la información de registro. Los archivos de archivador de tienda se encuentran en el directorio siguiente:

- ▶ **NT** unidad:\WebSphere\CommerceServer\samplstores
- ▶ **2000** unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
- ▶ **AIX** /usr/WebSphere/CommerceServer/samplstores
- ▶ **Solaris** /opt/WebSphere/CommerceServer/samplstores
- ▶ **Linux** /opt/WebSphere/CommerceServer/samplstores
- ▶ **400** /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Para ver el contenido del archivador de tienda, utilice un programa de descompresión. El archivo `command.xml` está ubicado en el directorio `data`.

2. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
3. Cree un archivo `command.xml`, copiando uno de los archivos `command.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a `command.xml`. Los archivos DTD se encuentran en el directorio siguiente:

- ▶ **NT** unidad:\WebSphere\CommerceServer\xml\sar
- ▶ **2000** unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
- ▶ **AIX** /usr/WebSphere/CommerceServer/xml/sar
- ▶ **Solaris** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **Linux** /opt/WebSphere/CommerceServer/xml/sar
- ▶ **400** /qibm/proddata/WebCommerce/xml/sar

4. Los mandatos de controlador deben registrarse en las tablas URLREG y CMDREG. Para registrar un mandato de controlador nuevo o personalizado en la tabla URLREG, cree una entrada en el archivo XML para cada mandato de controlador nuevo o personalizado, utilizando el ejemplo siguiente como muestra:

```
<urlreg
url="MiVisualizacióndeproductos"
storeent_id="@storeent_id_1"
interfacename="com.mitienda.commerce.catalog.commands.ProductDisplayCmd"
https="0"
description="Mandato de visualización de productos para mi tienda"
authenticated="0"
internal="0" />
```

donde

- `urlreg` es el nombre de la tabla de base de datos (URLREG) en la que se va a insertar esta información.
- `url` es el nombre de URI
- `storeent_id` es el identificador de entidad de tienda; el uso del símbolo `@` se denomina resolución de alias interno. Al utilizar la resolución de alias

interno, la clave primaria (identificador) se sustituye por un alias en el documento XML. Este alias se utiliza luego en cualquier otra parte del archivo XML para hacer referencia a ese elemento. De este modo, no es necesario saber los índices exclusivos necesarios para poder crear el archivo XML. Durante la publicación, ID Resolver sustituye el símbolo @ por un valor exclusivo. Para obtener más información, consulte el Apéndice B, "Creación de los datos" en la página 327.

- `interfacename` es el nombre de la interfaz de mandatos de controlador
 - `https` es el HTTP seguro necesario para esta petición de URL. Utilice 1 cuando sea necesario HTTP seguro y 0 cuando no lo sea.
 - `authenticated` indica si es necesario que el usuario esté conectado para esta petición de URL. Utilice 1 cuando sea necesaria la autenticación y 0 cuando no lo sea.
 - `internal` indica si el mandato es interno de WebSphere Commerce. Los URL internos los utilizan las herramientas de WebSphere Commerce. Utilice 1 cuando sea interno y 0 cuando sea externo. Los URL que cree deberán ser externos.
5. Para registrar un nuevo mandato de controlador o de tarea en la tabla `CMDREG`, cree una entrada en el archivo XML para cada mandato de controlador o de tarea nuevo o personalizado, utilizando el ejemplo siguiente de un mandato de tarea como muestra (del archivo `command.xml` de la tienda de ejemplo ToolTech):

```
< cmdreg
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.payment.commands.DoPaymentCmd"
classname="com.ibm.commerce.payment.commands.DoPaymentMPFCmdImpl"/>
donde
```

- `cmdreg` es el nombre de la tabla de base de datos (`CMDREG`) en la que se va a insertar esta información.
 - `storeent_id` es el identificador de entidad de tienda y el uso del símbolo @ se denomina resolución de alias interno. Al utilizar la resolución de alias interno, la clave primaria (identificador) se sustituye por un alias en el documento XML. Este alias se utiliza luego en cualquier otra parte del archivo XML para hacer referencia a ese elemento. De este modo, no es necesario saber los índices exclusivos necesarios para poder crear el archivo XML. Durante la publicación, ID Resolver sustituye el símbolo @ por un valor exclusivo. Para obtener más información, consulte el Apéndice B, "Creación de los datos" en la página 327.
 - `interfacename` es el nombre de la interfaz de mandatos
 - `classname` es el nombre de la clase de implementación del mandato. Normalmente, es el nombre de la interfaz añadiendo `Impl` al final.
6. Para registrar nuevas vistas, o para asociar nuevos archivos JSP a una vista, cree una entrada en la tabla `VIEWREG`, utilizando el ejemplo siguiente como muestra (del archivo `command.xml` de la tienda de ejemplo ToolTech):

```
<viewreg
viewname="OrderOptionsView"
devicefmt_id="-1"
storeent_id="@storeent_id_1"
interfacename="com.ibm.commerce.command.ForwardViewCommand"
classname="com.ibm.commerce.command.HttpForwardViewCommandImpl"
```

```
properties="docname=Shipping.jsp"
internal="0"
https="0"/>
```

donde

- `viewreg` es el nombre de la tabla de base de datos (VIEWREG) en la que se va a insertar esta información.
- `viewname` es el nombre de la vista.
- `devicefmt_id` es el tipo de dispositivo en el que se utilizará esta vista, por ejemplo, un navegador.
- `storeent_id` es el identificador de entidad de tienda; el uso del símbolo `@` se denomina resolución de alias interno. Al utilizar la resolución de alias interno, la clave primaria (identificador) se sustituye por un alias en el documento XML. Este alias se utiliza luego en cualquier otra parte del archivo XML para hacer referencia a ese elemento. De este modo, no es necesario saber los índices exclusivos necesarios para poder crear el archivo XML. Durante la publicación, ID Resolver sustituye el símbolo `@` por un valor exclusivo. Para obtener más información, consulte el Apéndice B, "Creación de los datos" en la página 327.
- `interfacename` es el nombre de la interfaz de mandatos de vista. Las opciones por omisión son `ForwardView`, `DirectView` y `RedirectView`.
- `classname` es el nombre de la clase de implementación de la vista. Normalmente, es el nombre de la interfaz añadiendo `Impl` al final.
- `properties` son las parejas nombre-valor por omisión definidas como propiedades de entrada del mandato. Si siempre se visualiza la misma página, establezca el nombre de archivo JSP en esta propiedad; por ejemplo, `docname=Shipping.jsp`.
- `internal` indica si la vista es interna de WebSphere Commerce. Las vistas internas las utilizan las herramientas de WebSphere Commerce. Utilice 1 cuando sea interna y 0 cuando sea externa. Las vistas que cree deberán ser externas.
- `https` es el HTTP seguro necesario para esta petición de URL. Utilice 1 cuando sea necesario HTTP seguro y 0 cuando no lo sea.



Para obtener más información sobre cómo utilizar `@` y `&` consulte el Apéndice B, "Creación de los datos" en la página 327.

Capítulo 8. Elementos de catálogo

Al igual que un catálogo tradicional, el catálogo en línea contiene los bienes y servicios que se ofrecen para la venta. Aunque el tamaño y la estructura de los catálogos en línea pueden variar mucho de una tienda a otra, dependiendo del tipo y la cantidad de mercancía disponible para la venta, los catálogos requieren lo siguiente:

- Los artículos y servicios que se ofrecen, incluido:
 - Los precios que normalmente se incluyen en el catálogo en línea
 - Los datos de los productos, como por ejemplo, descripciones e imágenes de la mercancía.
 - Las categorías, ya que la mayoría de catálogos aunque no todos, dividen la mercancía en categorías, para que a los clientes les resulte más fácil examinarlos.
- Un método de visualización para mostrar lo que se va a vender. Las páginas de visualización del catálogo muestran a los clientes las páginas y todas las páginas del catálogo tienen un diseño coherente. La estructura del catálogo depende de la mercancía.

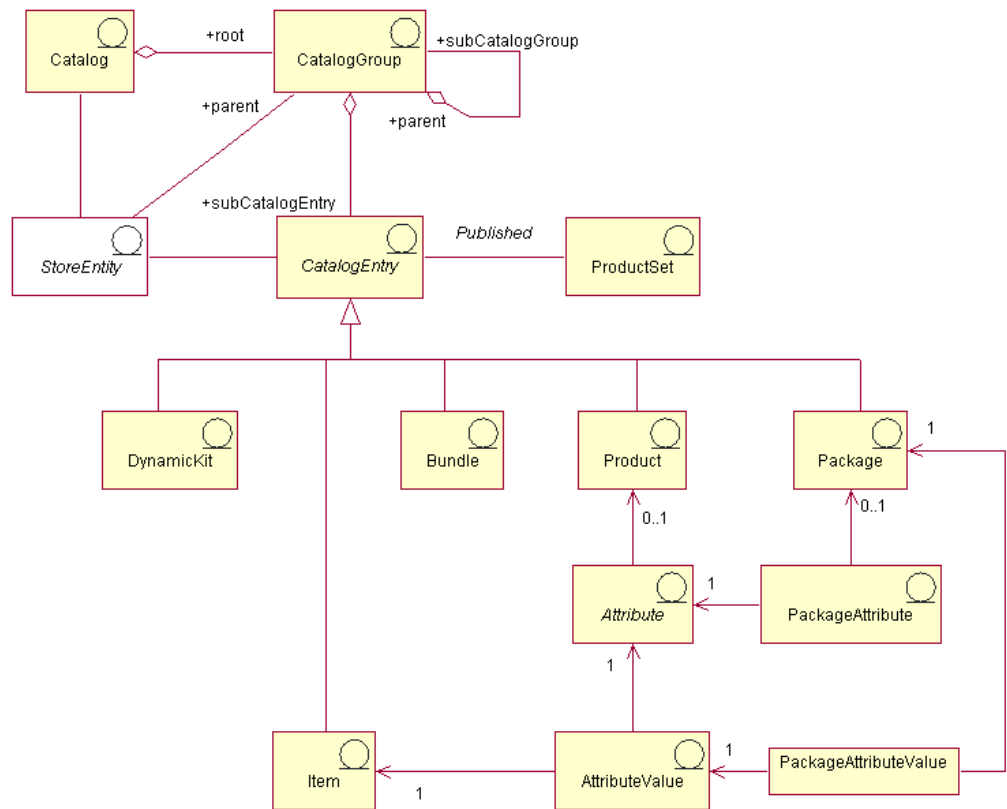
Los catálogos en WebSphere Commerce

WebSphere Commerce impone varios requisitos al catálogo en línea de la tienda. Cada tienda del sistema WebSphere Commerce debe tener un catálogo maestro. El catálogo maestro es la ubicación central para gestionar la mercancía de la tienda. Es el catálogo individual que contiene todos los productos, artículos, relaciones y precios estándar para todo lo que está a la venta en la tienda.

Se puede compartir el catálogo maestro entre las tiendas y definir tantas tiendas como se necesiten. Además de crear un catálogo maestro para la gestión de catálogo, también puede elegir crear uno o más catálogos de navegación para visualizarlos. Un catálogo de navegación puede contener las mismas entradas que el catálogo maestro, pero con una estructura mucho más flexible para que lo visualicen los clientes. Puede tener tantos catálogos de navegación como desee. Sin embargo, dado que el catálogo maestro es el que se utiliza para gestionar la mercancía en línea, recomendamos que también utilice el catálogo maestro como catálogo de navegación a fin de minimizar la actividad general de mantenimiento.

Si crea un nuevo catálogo maestro para utilizarlo con una tienda de WebSphere Commerce o utiliza un catálogo maestro ya existente de una tienda de ejemplo de WebSphere Commerce, como ToolTech, tendrá que modificar su catálogo para satisfacer estos requisitos. El diagrama siguiente muestra la estructura básica de un

catálogo en WebSphere Commerce.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Catálogos

El *catálogo* es el punto de partida del modelo de información. El catálogo contiene toda la información jerárquica y de navegación para el catálogo en línea y es un conjunto de grupos de catálogo y de entradas de catálogo que se pueden visualizar y adquirir en una tienda en línea.

En WebSphere Commerce, un catálogo se representa en la base de datos mediante una *entidad de catálogo*. Una entidad de catálogo consta de un ID de catálogo exclusivo y una descripción de catálogo, por ejemplo, el nombre de catálogo. Puesto que cada catálogo es una entidad aparte y exclusiva, puede asociarse fácilmente a una o más tiendas. Cada tienda del sistema WebSphere Commerce debe estar relacionada por lo menos con una entidad de catálogo. El catálogo maestro es un catálogo especial que contiene todos los artículos que están a la venta en la tienda en línea.

Grupos de catálogo

Los *grupos de catálogo* son agrupaciones genéricas de las entradas de catálogo, que se crean por motivos de navegación y de partición del catálogo. Un grupo de

catálogo pertenece a un catálogo y puede contener más de un grupo de catálogo o entradas de catálogo. Puede asociar grupos de catálogo con más de un catálogo. Los grupos de catálogo también se conocen como *categorías*.

Un catálogo plano es un catálogo que no agrupa sus productos en categorías sino que muestra una lista de productos. Aunque es posible crear un catálogo plano en WebSphere Commerce, es recomendable crear grupos en el catálogo por motivos de estructura y de navegación.

Para crear grupos de catálogo, en primer lugar debe ordenar el catálogo en una jerarquía o árbol invertido. El árbol empieza en un grupo de catálogo general (denominado categoría raíz) y se ramifica en subcategorías cada vez más específicas hasta que no se puede dividir más. Los grupos de catálogo de nivel inferior, que sólo contienen productos, son categorías hoja (también denominadas final). Un grupo de catálogo es el padre de las categorías que están inmediatamente debajo de él y el hijo del grupo de catálogo que está por encima. Por ejemplo, Moda para caballero es la agrupación de las categorías de ropa para hombre, mientras que los grupos de catálogo Pantalones y Camisas son agrupaciones de productos.

Entradas de catálogo

Cada grupo de catálogo contiene *entradas de catálogo*. Las entradas de catálogo representan mercancía que se puede solicitar en un catálogo en línea. Normalmente, las entradas tienen un nombre o número de pieza, una descripción, uno o más precios, imágenes y otros detalles. Una entrada de catálogo puede ser un producto, un artículo, un paquete, un paquete compuesto o un kit dinámico. Si es necesario, puede crear nuevos tipos de entrada de catálogo que no se ajusten a ninguno de los cinco modelos existentes. A continuación encontrará más información sobre cada uno de estos tipos de entrada de catálogo.

Productos

Un *producto* es un tipo de entrada de catálogo. Un producto actúa como una plantilla para un grupo de artículos que muestran los mismos atributos. Por ejemplo, una camisa es un producto en el catálogo. Después de añadir atributos y valores de atributo a la camisa, cada variación se convierte en un artículo, como por ejemplo, una camisa negra de talla pequeña.

Artículos

Un *artículo* es una unidad tangible de mercancía que tiene un nombre, un número de pieza y un precio específicos. Por ejemplo, una camisa negra de talla pequeña es un artículo, mientras que una camisa es un producto. Todos los artículos que están relacionados con un producto determinado presentan el mismo conjunto de atributos y se distinguen por los valores de sus atributos.

Nota: Para los usuarios de WebSphere Commerce Accelerator, artículos y *códigos de artículo* se consideran sinónimos. Cuando se utilizan las herramientas de Gestión de productos en WebSphere Commerce Accelerator, el artículo que puede pedirse recibe el nombre de código de artículo. En el esquema de base de datos de WebSphere Commerce, este tipo específico de entrada de catálogo se denomina artículo.

Paquetes

Un *paquete* es un conjunto indivisible de entradas de catálogo. Por ejemplo, un paquete de sistema informático puede contener una unidad central de proceso, un

monitor y una unidad de disco duro específicos que no se pueden vender por separado. Del mismo modo que los productos, un paquete tiene atributos que lo definen y es un contenedor de paquetes totalmente resueltos. Un paquete totalmente resuelto es comparable a un código de artículo. Un paquete tiene su propio precio y es un código de artículo real que se puede pedir y que se puede añadir al carro de la compra. Un paquete no se puede dividir ni modificar durante la navegación o después de haberse colocado en el carro de la compra.

Paquetes compuestos

Un *paquete compuesto* es un conjunto de entradas de catálogo que permite a los clientes comprar varios artículos de una sola vez. Por ejemplo, un paquete compuesto para un sistema informático puede estar formado por una unidad central de proceso, un monitor, una unidad de disco duro y una unidad de CD-ROM. Un paquete compuesto es una agrupación de artículos o una combinación de productos, artículos y paquetes totalmente resueltos. Si selecciona un paquete compuesto que sólo contenga artículos, el paquete compuesto se descompone en códigos de artículo que se pueden pedir por separado y que se añaden individualmente al carro de la compra. Sin embargo, si selecciona un paquete compuesto que contiene productos, será necesario convertir dichos productos en artículos mediante la resolución del código de artículo para poderlos añadir al carro de la compra. En cualquiera de los casos, una vez se ha descompuesto un paquete compuesto y los artículos que lo componían se han añadido al carro de la compra, los artículos pueden modificarse o eliminarse.

Kits dinámicos

Un *kit dinámico* es un tipo de entrada de catálogo que el cliente puede configurar dinámicamente. Esta configuración (o agrupación) de productos se basa en los requisitos del cliente y se vende como una sola unidad. Los componentes de un kit dinámico los combina un configurador de productos externo mediante un conjunto de normas predefinidas y la interacción del usuario. La adición de un kit dinámico a un pedido es similar a la adición de un paquete. Al igual que sucede en un paquete, los componentes individuales de un kit dinámico no pueden modificarse y se debe llevar a cabo la configuración entera en su conjunto. Sin embargo, puede cambiar los componentes del kit dinámico volviendo a configurarlo mediante un configurador de productos externo.

Conjuntos de productos

Los *conjuntos de productos* están asociados a los objetos de entradas de catálogo publicados. Los conjuntos de productos proporcionan un mecanismo que permite dividir el catálogo en subconjuntos lógicos. Esta división le permite mostrar diferentes partes del catálogo a usuarios diferentes. Puede crear un contrato y especificar que los participantes del contrato sólo tienen autorización para comprar productos que estén incluidos en un conjunto de productos predefinido. WebSphere Commerce proporciona herramientas para crear conjuntos de productos a partir del catálogo maestro y utilizarlos en contratos para filtrar la autorización.

Atributos

Los *atributos* son las propiedades de los productos de una tienda en línea, como el color y el tamaño. Un atributo pertenece a un producto. Cada combinación posible de atributos y valores de atributo define un artículo.

Valores de atributo

Los *valores de atributo* son las propiedades de un atributo, como por ejemplo, un color específico (azul o amarillo) o el tamaño (pequeño, medio o grande). Deberá predefinir los valores de atributo antes de asignarlos a los artículos. Cada combinación posible de atributos y valores de atributo define un artículo.

Atributos de paquete

Los *atributos de paquete* se heredan de los atributos de los productos que contienen los paquetes. Un paquete puede tener un atributo de cero o varios paquetes. Un atributo de paquete hace referencia a un atributo.

Valores de atributo de paquete

Los *valores de atributo de paquete* son los valores asignados a los atributos de paquete. Estos valores se heredan de los valores de atributo de los productos que contienen los paquetes. Un valor de atributo de paquete hace referencia a un valor de atributo.



Para obtener información más detallada sobre la estructura de los elementos de catálogo en WebSphere Commerce, consulte los modelos de datos de catálogo en la ayuda en línea de WebSphere Commerce.

Creación de elementos de catálogo en WebSphere Commerce

Para crear los elementos de catálogo para su tienda, debe crear un catálogo maestro añadiendo información a varias tablas de base de datos de WebSphere Commerce. Puede crear su catálogo utilizando archivos XML que se cargan en la base de datos utilizando Loader package. Si crea un catálogo multicultural, necesitará archivos XML para cada entorno nacional al que la tienda dé soporte. Cada archivo XML específico de entorno nacional añade al catálogo, grupos de catálogo y entradas de catálogo, la información que se puede traducir, por ejemplo, las descripciones.

A continuación se ofrece una visión general del proceso de creación del catálogo:

1. En WebSphere Commerce, un catálogo se crea utilizando archivos XML. La creación de un catálogo empieza con una entidad de catálogo, que equivale a un catálogo de papel en la base de datos.
2. Cree la estructura y el flujo de navegación añadiendo grupos de catálogo que determinen las categorías y la distribución de la mercancía.
3. Cree información de inventario como base para las entradas de catálogo.
4. Añada la mercancía en forma de entradas de catálogo, que representan productos, artículos, paquetes, paquetes compuestos y kits dinámicos.
5. A los productos del catálogo se les añaden atributos y valores de atributo para diferenciar los distintos artículos.
6. Puede crear paquetes y paquetes compuestos para agrupar ciertas entradas de catálogo, con fines promocionales.
7. A continuación se crean las relaciones entre los grupos de catálogo y las entradas de catálogo. Estas relaciones determinan qué entradas pertenecen a un grupo de catálogo.
8. Puede crear asociaciones de comercialización para las entradas de catálogo, como estrategias de recomendación de productos.
9. Asocie el catálogo, los grupos de catálogo y las entradas de catálogo a la tienda de WebSphere Commerce.







10. En el paso final, debe crear:
 - a. impuestos para su mercancía,
 - b. métodos de envío,
 - c. un centro de despacho de pedidos que actúe como almacén de las existencias, y centro de envío y recepción. Una tienda puede tener definidos más de un centro de despacho de pedidos.
 - d. precios de los productos.

Creación de un catálogo maestro

Para crear un catálogo maestro que contenga varios niveles de categorías, realice las tareas siguientes:







Parte 1: Preparación para la creación del catálogo

1. Revise la información del catálogo y sus modelos de objetos y datos correspondientes incluidos en WebSphere Commerce. La información del catálogo es un componente de WebSphere Commerce Server que hace posible la navegación, la división, el establecimiento de categorías y las asociaciones de la mercancía que se puede adquirir.
2. Revise la información de WebSphere Commerce Loader package. Loader package consta principalmente de utilidades para preparar y cargar datos en una base de datos de WebSphere Commerce. Puede utilizar Loader package para cargar grandes cantidades de datos y actualizar los datos de su base de datos. Para obtener más información sobre Loader package, consulte la Parte 7, "Publicación de la tienda" en la página 225.
3. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
4. Cree una organización mediante la Consola de administración, para que actúe como propietario del catálogo. Para obtener más información, consulte el tema "Crear una organización" de la ayuda en línea de WebSphere Commerce.
5. Cree un nuevo archivo XML para su catálogo maestro utilizando como guía las entradas XML existentes y los archivos `catalog.xml` de la tienda de ejemplo ToolTech. Si crea un catálogo multicultural, cree un archivo `catalog.xml` para cada entorno nacional al que la tienda dé soporte. El archivo específico de entorno nacional debe especificar toda la información de las descripciones, para que se pueda traducir fácilmente. En este ejemplo, se utilizará un archivo `catalog.xml` para toda la información que no es necesario traducir y otro archivo `catalog.xml` para cada entorno nacional soportado por la tienda, con la información que debe traducirse. O, si lo prefiere, puede utilizar el archivo XML existente de la tienda de ejemplo ToolTech y cambiar la información según sea necesario. Los archivos `catalog.xml` de la tienda de ejemplo ToolTech se encuentran en su archivo archivador de tienda. Para ver los archivos `catalog.xml`, descomprima el archivador de tienda utilizando un programa ZIP. Los archivos `catalog.xml` se encuentran en los directorios de datos siguientes:

-  `unidad:\WebSphere\CommerceServer\samplstores`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores`
-  `/usr/WebSphere/CommerceServer/samplstores`
-   `/opt/WebSphere/CommerceServer/samplstores`
-  `/qibm/proddata/WebCommerce/samplstores`

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

El archivo `catalog.dtd` está ubicado en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\xml\sar`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar`
-  `/usr/WebSphere/CommerceServer/xml/sar`
-   `/opt/WebSphere/CommerceServer/xml/sar`
-  `/qibm/proddata/WebCommerce/xml/sar`

Parte 2: Creación de una entrada de catálogo

1. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, cree una entidad de catálogo añadiendo información en las tablas `CATALOG` y `CATALOGDSC`. Una entidad de catálogo representa un catálogo en la base de datos.

```
<catalog
catalog_id="@catalog_id_1"
member_id="&MEMBER_ID;"
identifier="ToolTech"
description="Catálogo de ToolTech"
tpclevel="0"
/>
```

donde

- `catalog_id` es el número de referencia interno.
 - `member_id` es el número de referencia interno que identifica al propietario del catálogo.
 - `identifier` es un nombre externo para el catálogo.
 - `description` es una descripción del catálogo.
2. Utilice el siguiente ejemplo de la tienda ToolTech de ejemplo, y añada la descripción del catálogo en el archivo XML específico del entorno nacional para fines de traducción:

```
<catalogdsc
catalog_id="@catalog_id_1"
language_id="&es_ES;"
name="Catálogo maestro de la tienda"
/>
```

donde

- `catalog_id` es el número de referencia interno que relaciona esta información específica del idioma con un catálogo.
- `language_id` es el identificador del idioma.
- `name` es el nombre dependiente del idioma de la entrada de catálogo.

Parte 3: Creación de grupos de catálogo

1. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, cree grupos de catálogo añadiendo información a las tablas `CATGROUP` y `CATGRPDESC`. Los grupos de catálogo, conocidos también como categorías, son agrupaciones de otros grupos de catálogo o de productos. Complete esta tarea para cada grupo de catálogo en su catálogo:

```
<catgroup
catgroup_id="@catgroup_id_1"
member_id="MEMBER_ID;"
identifier="Herramientas para madera"
markfordelete="0"
/>
```

donde

- `catgroup_id` es el número de referencia interno del grupo de catálogo
- `member_id` es el número de referencia interno que identifica al propietario del catálogo.
- `identifier` es un nombre externo para el catálogo.
- `markfordelete` indica si el grupo de catálogo se ha marcado para que se suprima.
 - 0 = No.
 - 1 = Sí.

2. Utilice el siguiente ejemplo de la tienda ToolTech de ejemplo como guía, y añada la descripción del grupo de catálogo en el archivo XML específico del entorno nacional para fines de traducción. Complete esta tarea para cada grupo de catálogo en su catálogo:

```
<catgrpdesc
language_id="es_ES;"
catgroup_id="@catgroup_id_1"
name="Herramientas para madera"
shortdescription="Herramientas para madera"
longdescription="Herramientas para madera"
published="1"
/>
```

donde

- `language_id` es el identificador del idioma.
- `catgroup_id` es el número de referencia interno del grupo de catálogo.
- `name` es el nombre dependiente del idioma del catálogo.
- `shortdescription` es una descripción breve del grupo de catálogo.
- `longdescription` es una descripción detallada del grupo de catálogo.
- `published` indica si este grupo de catálogo debe visualizarse para el idioma indicado por `language_id`:
 - 0 = No.
 - 1 = Sí.

Nota: Cada vez que crea un grupo de catálogo y su descripción, se cambia `catgroup_id` para representar un nuevo grupo de catálogo. Por ejemplo, `catgroup_id="@catgroup_id_2"`, `catgroup_id="@catgroup_id_3"` y `catgroup_id="@catgroup_id_4"`, etc.

3. Después de crear los grupos de catálogo, asigne al catálogo un grupo de catálogo de nivel superior añadiendo información a la tabla `CATTOGRP`. Este grupo de catálogo es el padre de los grupos de catálogo inmediatamente debajo de él. Complete esta tarea para cada grupo de catálogo de nivel superior en su catálogo. Utilice el siguiente ejemplo de la tienda ToolTech como muestra:

```
<cattogr
catalog_id="@catalog_id_1"
catgroup_id="@catgroup_id_1"
/>
```


donde

- catalog_id es el número de referencia del catálogo.
- catgroup_id es el número de referencia del grupo de catálogo.

Nota: Cada vez que asigna al catálogo grupos de catálogo de nivel superior, se cambia catgroup_id para representar una nueva asociación de grupo de catálogo. Por ejemplo, catgroup_id="@catgroup_id_2", catgroup_id="@catgroup_id_3", catgroup_id="@catgroup_id_4", etc.

4. Una vez determinada la estructura jerárquica para los grupos de catálogo, cree relaciones entre los grupos de catálogo añadiendo información a la tabla CATGRPREL. Complete esta tarea para cada estructura de grupo de catálogo padre e hijo de su catálogo. Utilice el siguiente ejemplo de la tienda ToolTech como muestra:

```
<catgrprel
catgroup_id_parent="@catgroup_id_1"
catgroup_id_child="@catgroup_id_11"
catalog_id="@catalog_id_1"
sequence="0"
/>
```

donde

- catgroup_id_parent es el grupo de catálogo origen de esta relación.
- catgroup_id_child es el grupo de catálogo destino de esta relación.
- catalog_id es el número de referencia del catálogo.
- sequence es el número que determina el orden de visualización del contenido del grupo de catálogo.

Nota: Con cada relación de grupo de catálogo, se cambian catgroup_id_child y sequence para representar una nueva relación. Por ejemplo, las relaciones subsiguientes se visualizarían como catgroup_id_child="@catgroup_id_12" y sequence="1", catgroup_id_child="@catgroup_id_13" y sequence="2", etc. Si no está utilizando una estructura de navegación en su catálogo, puede suprimir la relación CATGRPREL.

Parte 4: Creación de información de inventario

1. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y cree información de inventario añadiendo información a las tablas BASEITEM, BASEITEMDSC, ITEMSPC, ITEMVERSN, VERSIONSPC, DISTARRANG y STOREITEM. Empiece creando los artículos base añadiendo información a la tabla BASEITEM. Los artículos base representan una familia general de productos con un nombre y descripción comunes. Complete esta tarea para cada grupo de artículos de inventario de su catálogo:

```
<baseitem
baseitem_id="@baseitem_id_102"
member_id="&MEMBER_ID;"
markfordelete="0"
partnumber="tooltech_sku_102"
itemtype_id="ITEM"
quantitymeasure="C62"
quantitymultiple="1.0"
/>
```

donde

- baseitem_id es la clave exclusiva generada.
- member_id es el propietario del artículo base.

- `markfordelete` indica si el artículo base se ha marcado para que se suprima.
 - 0 = No.
 - 1 = Sí.
- `partnumber` identifica de forma exclusiva el artículo base para el propietario.
- `itemtype` es el tipo de artículo base:
 - ITEM = artículos, paquetes o paquetes compuestos
 - DNKT = kits dinámicos.
- `quantitymeasure` es la unidad de medida para `quantitymultiple`.
- `quantitymultiple` es la cantidad de artículo base que se mide en unidades integrales. Junto con `quantitymeasure`, indica lo que representa cada unidad integral.

Nota: Debe crear un artículo base para cada producto que cree en el catálogo. Cada vez que crea un artículo base, los números `baseitem_id` y `partnumber` cambian para crear un nuevo artículo base. Por ejemplo, un nuevo artículo base podría contener `baseitem_id="@baseitem_id_147"` y `partnumber="tooltech_sku_147"` como entradas, mientras que otro podría contener `baseitem_id="@baseitem_id_192"` y `partnumber="tooltech_sku_192"`, etc.

2. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, añade información sobre artículos especificados a la base de datos. Un artículo especificado es un artículo con valores para todos sus atributos y representa un artículo, paquete, paquete compuesto o kit dinámico del catálogo. Complete esta tarea para cada artículo especificado del catálogo:

```
<itemspc
itemspc_id="@itemspc_id_106"
baseitem_id="@baseitem_id_102"
markfordelete="0"
partnumber="T0000106"
member_id="MEMBER_ID;"
discontinued="N"
/>
```

donde

- `itemspc_id` es la clave exclusiva generada.
- `baseitem_id` es el artículo base del producto.
- `markfordelete` indica si el artículo especificado se ha marcado para que se suprima.
 - 0 = No.
 - 1 = Sí.
- `partnumber` identifica de forma exclusiva el artículo especificado para el propietario.
- `member_id` es el propietario del artículo especificado.
- `discontinued` indica si el artículo especificado ha dejado de fabricarse:
 - Y=Ha dejado de fabricarse y puede solicitarse si hay suficiente inventario pero no podrán servirse en los pedidos en espera.
 - N=Activo y pueden solicitarse en pedidos en espera si no quedan existencias.

Nota: Debe crear un artículo especificado para cada artículo que cree en el catálogo. Cada vez que define un artículo especificado, los números `itemspc_id="@itemspc_id_107"`, `baseitem_id="@baseitem_id_102"` y

partnumber="T0000107" cambian para crear un nuevo artículo especificado. Por ejemplo, un nuevo artículo especificado podría contener itemspc_id="@itemspc_id_108", baseitem_id="@baseitem_id_102" y partnumber="T0000108" como entradas, mientras que otro podría contener itemspc_id, baseitem_id y partnumber como entradas, etc.

3. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y añada a la base de datos la información siguiente sobre la relación entre la versión de un artículo y el artículo base. Complete esta tarea para cada relación de este tipo del catálogo:

```
<itemversn
itemversn_id="@itemversn_id_102"
baseitem_id="@baseitem_id_102"
expirationdate="2010-01-01 00:00:00.000000"
versionname="version"
/>
```

donde

- itemversn_id es un número de referencia generado que identifica la versión del artículo.
- baseitem_id es el artículo base.
- expirationdate es la hora en que caduca la versión del artículo.
- versionname identifica de forma exclusiva la versión del artículo para su artículo base.

Nota: Cada vez que crea una relación entre la versión de un artículo y el artículo base, se modifican los números itemversn_id y baseitem_id para crear una nueva relación. baseitem_id coincide con un artículo base existente. Por ejemplo, una relación nueva puede contener como entradas itemversn_id="@itemversn_id_107" y baseitem_id="@baseitem_id_107", mientras que otra relación puede contener como entradas itemversn_id="@itemversn_id_108" y baseitem_id="@baseitem_id_108", etc.

4. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, añada la información siguiente sobre la relación entre la versión de un producto y el artículo especificado en la base de datos. Complete esta tarea para cada relación de este tipo del catálogo:

```
<versionspc
versionspc_id="@versionspc_id_106"
itemspc_id="@itemspc_id_106"
itemversn_id="@itemversn_id_102"
/>
```

donde

- versionspc_id es el identificador exclusivo generado.
- itemspc_id es el artículo especificado con el que está relacionada esta entrada del catálogo.
- itemversn_id identifica la versión del artículo.

Nota: Cada vez que crea una relación entre la versión de un producto y el artículo especificado, los números versionspc_id e itemspc_id cambian para crear una nueva relación. itemspc_id coincide con un artículo especificado existente. Por ejemplo, una nueva relación podría contener como entradas versionspc_id="@versionspc_id_107" e

itemspc_id="@itemspc_id_107" , mientras que otra podría contener como entradas versionspc_id="@versionspc_id_108" e itemspc_id="@itemspc_id_108", etc.

5. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y añada el acuerdo de la distribución a la base de datos. Un acuerdo de distribución permite que una tienda venda su propio inventario. Complete esta tarea para cada acuerdo de distribución del catálogo:

```
<distarrang
distarrang_id="@distarrang_id_102"
wholesalestore_id="@storeent_id_1"
merchantstore_id="@storeent_id_1"
baseitem_id="@baseitem_id_102"
pickingmethod="F"
startdate="2000-12-25 00:00:00.000000"
enddate="2010-01-01 00:00:00.000000"
/>
```

donde

- distarrang_id es el número de referencia del acuerdo de distribución.
- wholesalestore_id es la tienda de ventas al mayor propietaria del inventario que la tienda del comerciante puede vender. Este ID debe ser igual que merchantstore_id.
- merchantstore_id es la tienda del comerciante que puede vender el inventario de la tienda de ventas al mayor. Este ID debe ser igual que wholesalestore_id.
- baseitem_id es el producto que cubre el acuerdo de distribución.
- pickingmethod determina la secuencia en la que se elige el inventario de la tabla RECEIPT según este acuerdo:
 - F = FIFO (primero en entrar primero en salir) - el inventario recibido con más anterioridad.
 - L = LIFO (último en entrar primero en salir) - el inventario recibido más recientemente.
- startdate es el hora en que este acuerdo de distribución empieza a entrar en vigor.
- enddate es la hora en que este acuerdo de distribución deja de estar en vigor.

Nota: Cada vez que crea un acuerdo de distribución, se modifican los números distarrang_id y baseitem_id para crear un nuevo acuerdo de distribución. Por ejemplo, un segundo acuerdo de distribución puede contener los valores distarrang_id="@distarrang_id_147" y baseitem_id="@baseitem_id_147", mientras que un tercero puede contener distarrang_id="@distarrang_id_192" y baseitem_id="@baseitem_id_192", etc.

6. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y añada a la base de datos los atributos que afectan cómo una tienda determinada asigna inventario para los artículos especificados de un artículo base determinado. Complete esta tarea para cada artículo base del catálogo:

```
<storeitem
baseitem_id="@baseitem_id_102"
storeent_id="@storeent_id_1"
trackinventory="Y"
forcebackorder="N"
releaseseparately="N"
returnnotdesired="N"
```

```
backorderable="Y"  
creditable="Y"  
minqtyforsplit="0"  
</>
```

donde

- `baseitem_id` es el artículo base.
- `storeent_id` es la tienda del grupo de tiendas.
- `trackinventory` controla si hace un seguimiento o no del inventario en la tabla RECEIPT:
 - N = no se hace un seguimiento del inventario y no hay entradas en la tabla RECEIPT.
 - Y = se hace un seguimiento del inventario en la tabla RECEIPT.
- `forcebackorder` suspende temporalmente la asignación de los artículos especificados para el artículo base:
 - N = se puede asignar el inventario (comportamiento normal)
 - Y = no se puede asignar el inventario incluso si hay inventario suficiente.
- `releaseseparately` controla cómo se liberan los artículos de un pedido especificados para el artículo base:
 - N = se pueden liberar los artículos junto con los artículos de pedido.
 - Y = los artículos de pedido se deben liberar de forma independiente (en sus propias cajas)
- `returnnotdesired` indica que no se desea la devolución de un artículo (por ejemplo, los alimentos perecederos), incluso si el cliente está dispuesto o desea devolverlo:
 - N = se evalúa la petición de abono en base a la intención del cliente de devolver el artículo, aunque no se espera la devolución.
 - Y = se evalúa la petición de crédito como si se esperara la devolución.
- `backorderable` indica que no se puede suspender la asignación de los artículos especificados para el artículo base:
 - N = no se puede suspender la asignación de los artículos.
 - Y = se puede suspender la asignación de los artículos.
- `creditable` indica si el comerciante abonará este artículo, sin ninguna alteración temporal:
 - N = se vende como está.
 - Y = se puede abonar.
- `minqtyforsplit` indica que los artículos del pedido no se dividirán automáticamente durante la asignación de inventario si la cantidad restante que no se ha asignado en el nuevo artículo de pedido es menor que la cantidad mínima especificada.

Nota: Cada vez que define las normas de asignación de inventario para un artículo de tienda, se cambia el número de `baseitem_id` para que represente un nuevo artículo base. Por ejemplo, una asignación nueva puede contener `baseitem_id="@baseitem_id_147"` y una tercera asignación puede contener `baseitem_id="@baseitem_id_192"`, etc.

7. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y añada al archivo XML específico del entorno nacional la descripción del artículo base. Complete esta tarea para cada descripción de artículo base del catálogo:

```

<baseitmdsc
baseitem_id="@baseitem_id_102"
language_id="&es_ES;"
shortdescription="Sierra circular"
longdescription="Sierra de poco peso y gran calidad. La sierra circular
pesa como máximo 4,5 Kg. y tiene un motor de 12 ó 14 amp con una
velocidad de hasta 600 rpms. La base de aleación de aluminio de 220 V,
de fricción reducida, permite que el trabajo se efectúe con gran rapidez."
/>

```

donde

- baseitem_id es la clave exclusiva generada.
- language_id es el idioma de esta información.
- shortdescription es una descripción breve del artículo base.
- longdescription es una descripción detallada del artículo base.

Parte 5: Creación de entradas de catálogo

1. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, cree entradas de catálogo añadiendo información a las tablas CATENTRY y CATENTDESC. Cada tipo de entrada de catálogo — productos, artículos, paquetes, paquetes compuestos y kits dinámicos — representa los artículos de mercancía a la venta en el catálogo. Tendrá que definir un artículo base para cada entrada de catálogo de producto. Complete esta tarea para cada entrada de catálogo de producto de su catálogo:

```

<catentry
catentry_id="@product_id_102"
baseitem_id="@baseitem_id_102"
member_id="&MEMBER_ID"
catenttype_id="ProductBean"
partnumber="T0000102"
mfpartnumber="Torsión-Herramientas-102"
mfname="Herramientas de torsión"
markfordelete="0"
buyable="1"
/>

```

donde

- catentry_id es el número de referencia interno de la entrada de catálogo de productos.
- baseitem_id es el artículo base con el que esta entrada de catálogo está relacionada.
- member_id es el número de referencia que identifica la entrada de catálogo.
- catenttype_id identifica el tipo de entrada de catálogo:
 - ItemBean = identifica un artículo.
 - ProductBean = identifica un producto.
 - PackageBean = identifica un paquete.
 - BundleBean = identifica un paquete compuesto.
 - DynamicKitBean = identifica un kit dinámico.
- partnumber es el número de referencia que identifica el número de pieza de la entrada de catálogo.
- mfpartnumber es el número de pieza que utiliza el fabricante para identificar la entrada de catálogo.
- mfname es el nombre del fabricante de la entrada de catálogo.
- markfordelete indica si la entrada de catálogo se ha marcado para que se suprima.

- 0 = No.
- 1 = Sí.
- buyable indica si se puede adquirir la entrada de catálogo individualmente:
 - 0 = No.
 - 1 = Sí.

Nota: Cada vez que añade un artículo base a una entrada de catálogo de producto, se cambia la secuencia de `catentry_id` y `baseitem_id` para representar una nueva entrada de catálogo. `catenttype_id` cambia según el tipo de la entrada de catálogo.

- Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y defina un artículo especificado para cada entrada de catálogo. Complete esta tarea para cada entrada de catálogo:

```
<catentry
catentry_id="@catentry_id_106"
itemspc_id="@itemspc_id_106"
member_id="&MEMBER_ID"
catenttype_id="ItemBean"
partnumber="T0000106"
mfpartnumber="Torsión-Herramientas-106"
mfname="Herramientas de torsión"
markfordelete="0"
buyable="1"
/>
```

donde

- `catentry_id` es el número de referencia interno de la entrada de catálogo.
- `itemspc_id` es el artículo especificado al que pertenece esta entrada de catálogo.
- `member_id` es el número de referencia que identifica la entrada de catálogo.
- `catenttype_id` identifica el tipo de entrada de catálogo:
 - `ItemBean` = identifica un artículo.
 - `ProductBean` = identifica un producto.
 - `PackageBean` = identifica un paquete.
 - `BundleBean` = identifica un paquete compuesto.
 - `DynamicKitBean` = identifica un kit dinámico.
- `partnumber` es el número de referencia que identifica el número de pieza de la entrada de catálogo.
- `mfpartnumber` es el número de pieza que utiliza el fabricante para identificar la entrada de catálogo.
- `mfname` es el nombre del fabricante de la entrada de catálogo.
- `markfordelete` indica si la entrada de catálogo se ha marcado para que se suprima.
 - 0 = No.
 - 1 = Sí.
- `buyable` indica si se puede adquirir la entrada de catálogo individualmente:
 - 0 = No.
 - 1 = Sí.

Nota: Cada vez que añade un artículo especificado a una entrada de catálogo, se cambia la secuencia de `catentry_id` e `itemspc_id` para

representar una nueva entrada de catálogo. `catenttype_id` cambia según el tipo de la entrada de catálogo. Debido a la limitación de la estructura del catálogo maestro, una entrada de catálogo no puede pertenecer a más de una categoría. Para colocar una entrada de catálogo en más de una categoría, deberá utilizar un catálogo de navegación.

- Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, añada la descripción al archivo XML específico del entorno nacional. Complete esta tarea para cada descripción de entradas de su catálogo:

```
<catentdesc
catentry_id="@product_id_102"
language_id="%es_ES"
name="Circular"
shortdescription="Sierra circular"
longdescription="Sierra de poco peso y gran calidad. La sierra circular
pesa como máximo 4,5 Kg. y tiene un motor de 12 ó 14 amp con una
velocidad de hasta 600 rpms. La base de aleación de aluminio de 220 V,
de fricción reducida, permite que el trabajo se efectúe con gran rapidez."
thumbnail="images/circular_saw_sm.gif"
fullimage="images/circular_saw.gif"
available="1"
published="1"
/>
```

donde

- `catentry_id` es el número de referencia interno que indica la entrada de catálogo a la que hace referencia esta información específica del idioma.
- `language_id` es el identificador del idioma.
- `name` es el nombre dependiente del idioma de la entrada de catálogo.
- `shortdescription` es una descripción breve de la entrada de catálogo.
- `longdescription` es una descripción detallada de la entrada de catálogo.
- `thumbnail` es la vía de acceso de la imagen reducida.
- `fullimage` es la vía de acceso de la imagen de tamaño normal.
- `available` indica el periodo de tiempo que la entrada de catálogo estará disponible.
- `published` indica si esta entrada de catálogo debe mostrarse para el idioma indicado por `language_id`
 - 0 = Visualizar.
 - 1 = No visualizar.

Parte 6: Creación de atributos y valores de atributo

1. Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y cree atributos y valores de atributo para sus productos añadiendo información a las tablas `ATTRIBUTE` y `ATTRVALUE` en el archivo XML específico del entorno nacional para fines de traducción. Cada producto del catálogo tiene un conjunto de atributos específicos, como por ejemplo, la talla y el color de una camisa o unos pantalones. Los artículos se definen con los valores de atributo. Por ejemplo, una camisa es un producto, pero una camisa negra de talla pequeña es un artículo. Complete esta tarea para cada atributo del catálogo:

```
<attribute
attribute_id="@attribute_id_103"
language_id="%es_ES"
attrtype_id="STRING"
name="Amps"
sequence="0"
```



```
description="Amps"  
catentry_id="@product_id_102"  
description2="Amps"  
</>
```

donde

- `attribute_id` es el número de referencia interno del atributo.
- `language_id` es el idioma al que pertenece este atributo.
- `attrtype_id` es el tipo del valor de atributo correspondiente.
- `name` es el nombre del atributo.
- `sequence` es un número de secuencia que determina el orden de visualización de los atributos de un producto específico.
- `description` es la descripción del atributo.
- `catentry_id` es el número de referencia del producto al que pertenece este atributo.
- `description2` es una descripción adicional del atributo.

Nota: Cada vez que añade un atributo a un producto definido mediante `catentry_id`, la secuencia de `attribute_id` cambia para representar un nuevo atributo.

2. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, añada los valores de atributo. Complete esta tarea para cada valor de atributo del catálogo:

```
<attrvalue  
attrvalue_id="@attrvalue_id_114"  
language_id="&es_ES"  
attribute_id="@attribute_id_103"  
name="14 amp"  
attrtype_id="STRING"  
stringvalue="14 amp"  
sequence="0"  
catentry_id="@catentry_id_106"  
>
```

donde

- `attrvalue_id` es el número de referencia interno del valor de atributo
- `language_id` es el idioma al que pertenece este valor de atributo
- `attribute_id` es el número de referencia interno del atributo asociado al valor
- `name` es el nombre del valor del atributo
- `attrtype_id` es el tipo del valor del atributo
- `stringvalue` es el valor del atributo
- `sequence` es un número de secuencia que determina el orden de visualización de los valores de un atributo específico.
- `catentry_id` es el ID de artículo que describe este valor de atributo

Nota: Cada vez que añade un valor a un atributo, la secuencia de `attrvalue_id` cambia para representar valores diferentes. La secuencia de `attribute_id` cambia para representar un atributo diferente. `sequence` aumenta con cada nuevo valor de atributo. Por ejemplo, los valores de atributo subsiguientes serían `sequence="1"`, `sequence="2"`, `sequence="3"`, etc.

Parte 7: Creación de relaciones entre productos y artículos

1. Después de crear productos y artículos para el catálogo, debe definir las relaciones entre ellos añadiendo información en la tabla CATENTREL. Utilice el siguiente ejemplo de la tienda ToolTech como muestra. Complete esta tarea para cada valor de relación entre producto y artículo del catálogo:

```
<catentrel
  catentry_id_parent="@product_id_147"
  catreltype_id="PRODUCT_ITEM"
  catentry_id_child="@catentry_id_152"
  sequence="2"
  quantity="1"
/>
```

donde

- `catentry_id_parent` es el número de referencia de la entrada de catálogo origen de esta relación, es decir, el producto.
- `catreltype_id` es el tipo de relación: `PRODUCT_ITEM`
- `catentry_id_child` es el número de referencia de la entrada de catálogo destino de esta relación, es decir, el artículo.
- `sequence` es el número de secuencia que se utiliza para determinar el orden de visualización.
- `quantity` es una cantidad que se puede asociar con la relación.

Nota: Cada vez que añade una relación entre un producto y un artículo, los números `catentry_id_parent` y `catentry_id_child` cambian para crear relaciones distintas, basados en `catreltype_id`. Con cada nueva relación, el número de `sequence` es distinto. Por ejemplo, si `sequence="2"`, el valor para la relación siguiente será `sequence="3"`, después `sequence="4"`, etc.

Parte 8: Creación de paquetes y paquetes compuestos

1. Una vez haya creado los productos y los artículos, debe crear los paquetes y paquetes compuestos añadiendo información a las tablas CATENTRY, CATENTDESC y CATENTREL. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, empiece creando un paquete o un paquete compuesto añadiendo información a la tabla CATENTRY. Complete esta tarea para cada paquete y paquete compuesto de su catálogo:

```
<catentry
  catentry_id="@package_id_102"
  member_id="@MEMBER_ID"
  catenttype_id="PackageBean"
  partnumber="sku-@package_id_102"
  mfpartnumber="sku-@package_id_102"
  mfname="ToolTech"
  markfordelete="0"
  buyable="1"
/>
```

donde

- `catentry_id` es el número de referencia de la entrada de catálogo.
- `member_id` es el número de referencia que identifica al propietario de la entrada de catálogo.
- `catenttype_id` identifica el tipo de entrada de catálogo:
 - `PackageBean` = identifica un paquete.
 - `BundleBean` = identifica un paquete compuesto.

- partnumber es el número de referencia que identifica el número de pieza de la entrada de catálogo.
- mfpnumber es el número de pieza que utiliza el fabricante para identificar la entrada de catálogo.
- mfname es el nombre del fabricante de la entrada de catálogo.
- markfordelete indica si la entrada de catálogo se ha marcado para que se suprima.
 - 0 = No.
 - 1 = Sí.
- buyable indica si la entrada de catálogo puede adquirirse individualmente:
 - 0 = No.
 - 1 = Sí.

Nota: Cada vez que crea un paquete o un paquete compuesto, se cambian los números de catentry_id, partnumber y mfpnumber para crear un paquete o paquete compuesto diferente. Por ejemplo, para crear un paquete nuevo puede utilizar catentry_id="@package_id_103", partnumber="sku-@package_id_103" y mfpnumber="sku-@package_id_103", incluyendo catenttype_id="PackageBean" para identificar la entrada como un paquete. Para crear un paquete compuesto nuevo, puede utilizar catentry_id="@package_id_110", partnumber="sku-@package_id_110" y mfpnumber="sku-@package_id_110" incluyendo catenttype_id="BundleBean" para identificar la entrada como un paquete compuesto, etc.

- Utilice el siguiente ejemplo de la tienda de ejemplo ToolTech como guía y añada la descripción del paquete o paquete compuesto añadiendo información a la tabla CATENTDESC del archivo XML específico del entorno nacional para fines de traducción. Complete esta tarea para cada descripción de paquete o paquete compuesto del catálogo:

```
<catentdesc
catentry_id="@catentry_id_102"
language_id="-1"
name="PC"
shortdescription="PC"
longdescription="Conjunto formado por una unidad central de proceso,
un monitor, un disco duro y una impresora a color. Un PC ideal
para principiantes."
thumbnail="images/package_system_sm.gif"
fullimage="images/package_system.gif"
available="1"
published="1"
/>
```

donde

- catentry_id es el número de referencia interno que indica la entrada de catálogo a la que hace referencia esta información específica del idioma.
- language_id es el identificador del idioma.
- name es el nombre dependiente del idioma de la entrada de catálogo.
- shortdescription es una descripción breve de la entrada de catálogo.
- longdescription es una descripción detallada de la entrada de catálogo.
- thumbnail es la vía de acceso a la imagen reducida de la entrada de catálogo.
- fullimage es la vía de acceso a la imagen de tamaño normal de la entrada de catálogo.

- available indica el periodo de tiempo que la entrada de catálogo estará disponible.
- published indica si la entrada de catálogo debe mostrarse para el idioma indicado por language_id
 - 0=No mostrar la entrada de catálogo.
 - 1=Mostrar la entrada de catálogo.
- Utilizando el siguiente ejemplo de la tienda de ejemplo ToolTech como guía, cree relaciones entre paquetes y paquetes compuestos añadiendo información a la tabla CATENTREL. Complete esta tarea para cada relación de componente de paquete y paquete compuesto del catálogo:

```
<catentrel
catentry_id_parent="@catentry_id_102"
catreltype_id="PACKAGE_COMPONENT"
catentry_id_child="@catentry_id_97"
sequence="1.0"
quantity="1.0"
/>
```

donde

- catentry_id_parent es el número de referencia de la entrada de catálogo origen de esta relación, es decir, el paquete o paquete compuesto.
- catreltype_id es el tipo de relación:
 - PACKAGE_COMPONENT representa una relación entre un paquete y sus componentes.
 - BUNDLE_COMPONENT representa una relación entre un paquete compuesto y sus componentes.
- catentry_id_child es el número de referencia de la entrada de catálogo destino de esta relación, es decir, el componente.
- sequence es el número de secuencia que se utiliza para determinar el orden de visualización.
- quantity es una cantidad que se puede asociar con la relación.

Nota: Cada vez que crea una relación entre un paquete y un paquete compuesto, los números de catentry_id_parent y catentry_id_child cambian para que coincidan con entradas de catálogo existentes. Con cada relación nueva, el número de sequence es diferente. Por ejemplo, si empieza con sequence="1.0", el valor de la relación siguiente será sequence="2.0", después sequence="3.0", etc.

Parte 9: Creación de relaciones entre grupos de catálogo y entradas de catálogo

1. Después de crear grupos de catálogo y entradas de catálogo, debe definir las relaciones entre ellos, añadiendo información a la tabla CATGPENREL. Debido a la limitación de la estructura del catálogo maestro, una entrada de catálogo no puede pertenecer a más de una categoría. Para colocar una entrada de catálogo en más de una categoría, deberá utilizar un catálogo de navegación. Utilice el siguiente ejemplo de la tienda ToolTech como muestra. Complete esta tarea para cada relación entre un grupo de catálogo y una entrada de catálogo:

```
<catgpenrel
catgroup_id="@catgroup_id_11"
catalog_id="@catalog_id_1"
catentry_id="@product_id_102"
sequence="0"
/>
```

donde

- `catgroup_id` es el grupo de catálogo origen de esta relación.
- `catalog_id` es el catálogo dentro del cual se encuentra esta relación.
- `catentry_id` es la entrada de catálogo destino de esta relación.
- `sequence` es el número de secuencia que determina el orden de visualización del contenido del grupo de catálogo.

Nota: Cada vez que crea una relación entre grupos de catálogo y entradas de catálogo, los números `catgroup_id` y `catentry_id` cambian para formar nuevas relaciones con distintos grupos de catálogo y entradas de catálogo. Con cada nueva relación, el número de `sequence` es distinto. Por ejemplo, si empieza con `sequence="0"`, el valor para la relación siguiente será `sequence="1"`, después `sequence="2"`, etc.

Parte 10: Creación de asociaciones de comercialización

1. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, cree asociaciones de comercialización entre entradas de catálogo, añadiendo información a la tabla MASSOCECE. Complete esta tarea para cada asociación de comercialización del catálogo:

```
<massoccece
massoccece_id="@relationship_id_100"
massoctype_id="X-SELL"
catentry_id_from="@product_id_1"
catentry_id_to="@product_id_15"
massoc_id="REQUIRES"
quantity="2.0"
rank="1.00000"
/>
```

donde

- `massoccece_id` es el número de referencia de esta entrada.
- `massoctype_id` es el identificador del tipo de asociación:
 - X-SELL = venta cruzada.
 - UPSELL = venta ascendente.
 - ACCESSORY = accesorio.
- `catentry_id_from` es la entrada de catálogo origen de la asociación.
- `catentry_id_to` es la entrada de catálogo destino de la asociación.
- `massoc_id` es el identificador del especificador semántico:
 - REQUIRED
 - OPTIONAL
 - COMES WITH
- `quantity` es la cantidad relacionada con esta asociación.
- `rank` es el número de secuencia que se utiliza para el orden de visualización.

Nota: Cada vez que añade una asociación de comercialización, el número de `massoccece_id` cambia para representar una nueva relación. Los números de `catentry_id_from` y `catentry_id_to` varían para crear nuevo contenido de mercancía para la asociación.

Parte 11: Asociación del catálogo a una tienda

1. Asocie el catálogo a una tienda, asignando el catálogo, sus grupos de catálogo y entradas de catálogo a una tienda en la base de datos, utilizando como muestra el archivo `storecatalog.xml` existente de la tienda de ejemplo

ToolTech. También debe asignar páginas de visualización a los grupos de catálogo y a las entradas de catálogo. Añada esta información a las tablas STORECAT, STORECENT, STORECGRP, DISPCGPREL y DISPENTREL. Si crea un catálogo multicultural, cree un archivo XML de relación entre el catálogo y la tienda para cada entorno nacional al que la tienda dé soporte:

```
<storecat
catalog_id="@catalog_id_1"
storeent_id="@storeent_id_1"
mastercatalog="1"
/>
```

donde

- catalog_id es el número de referencia del catálogo.
- storeent_id es el número de referencia de la entidad de tienda en la base de datos.
- mastercatalog especifica un catálogo maestro para la tienda. El valor 1 indica que este catálogo está designado como catálogo maestro.

2. Utilizando como guía el siguiente ejemplo de la tienda de ejemplo ToolTech, añada entradas de catálogo a la relación entre tienda y catálogo. Complete esta tarea para cada entrada de catálogo:

```
<storecent
storeent_id="@storeent_id_1"
catentry_id="@product_id_102"
/>
```

donde

- storeent_id es el número de referencia de la entidad de tienda en la base de datos.
- catentry_id es el número de referencia de la entrada de catálogo.

Nota: Cada vez que añada un valor para catentry_id a la entidad de tienda, el número de referencia cambia para que coincida con una entrada de catálogo existente.

3. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, añada grupos de catálogo a la entidad de tienda. Complete esta tarea para cada grupo de catálogo en su catálogo:

```
<storecgrp
storeent_id="@storeent_id_1"
catgroup_id="@catgroup_id_1"
/>
```

donde

- storeent_id es el número de referencia de la entidad de tienda en la base de datos.
- catgroup_id es el número de referencia del grupo de catálogo.

Nota: Cada vez que añada un valor para catgroup_id a la entidad de tienda, el número de referencia cambia para que coincida con un grupo de catálogo existente.

Parte 12: Asociación de impuestos al catálogo

1. Asocie impuestos a los productos y servicios del catálogo para una tienda específica. Debe asociar un código de cálculo a las entradas de catálogo,

añadiendo esta información a la tabla CATENCALCD. Para obtener más información, consulte “Creación de elementos de impuestos en WebSphere Commerce” en la página 158.

Parte 13: Asociación de métodos de envío al catálogo

1. Para asociar métodos de envío a los productos y servicios del catálogo, deberá asociar un código de cálculo de envío a las entradas de catálogo. Añada esta información a la tabla CATENCALCD. Para obtener más información, consulte “Creación de elementos de envío en WebSphere Commerce” en la página 139.

Parte 14: Asociación de un centro de despacho de pedidos al catálogo

1. Para enviar productos a los clientes, debe asociar el catálogo a un centro de despacho de pedidos. Un centro de despacho gestiona las existencias y el envío de productos para una tienda. Añada esta información a la tabla FFMCENTER. Para obtener más información, consulte “Creación de elementos de despacho de pedidos en WebSphere Commerce” en la página 114.

Parte 15: Creación de precios para las entradas de catálogo

1. Cree los precios para las entradas de catálogo. El cálculo de precios representa el rango de precios para una entrada de catálogo y los criterios que deben satisfacerse para utilizar cada precio. Para crear un catálogo operativo, debe añadir información sobre ofertas a la base de datos. Añada esta información a las tablas TRADEPOSCN, TDPSCNCNTR, MGPTRDPSCN, OFFER y OFFERPRICE. Para obtener más información, consulte “Creación de elementos de cálculo de precio en WebSphere Commerce” en la página 94. Alternativamente, puede crear o actualizar los precios para una entrada de catálogo utilizando las herramientas de Gestión de productos de WebSphere Commerce Accelerator.

Parte 16: Carga del archivo XML

1. Después de crear los datos, cargue el archivo XML en la base de datos utilizando Loader package o mediante la función Publicar de los Servicios de tienda. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225.






Nota: También puede utilizar las herramientas de Gestión de productos de WebSphere Commerce Accelerator para crear elementos de catálogo para su catálogo maestro. Para obtener información más detallada sobre las herramientas de Gestión de productos, consulte la ayuda en línea de WebSphere Commerce.

Visualización de los elementos de catálogo de la tienda

Para asociar un catálogo, grupos de catálogo y entradas de catálogo a una tienda, así como para asignar plantillas JSP para visualizar las entradas de catálogo y los grupos de catálogo, debe crear estas relaciones en la base de datos. Cree estas relaciones en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package.







El archivo `storecatalog.xml` de la tienda de ejemplo ToolTech se encuentra en su archivo archivador de tienda. Para ver el archivo `storecatalog.xml`, descomprima el archivador de tienda utilizando un programa ZIP. El archivo `storecatalog.xml` se encuentra en el directorio de datos siguiente:

-  `unidad:\WebSphere\CommerceServer\samples\estores`

-  *unidad*:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-   /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

El archivo store-catalog.dtd se encuentra en el directorio siguiente:

-  *unidad*:\WebSphere\CommerceServer\xml\sar
-  *unidad*:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-   /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

Antes de crear las relaciones entre tienda y catálogo, debe asegurarse de que ha creado los elementos de tienda. Complete las tareas siguientes, cada una de las cuales crea entradas en el archivo storecatalog.xml:

1. Para poder visualizar los grupos de catálogo (categorías) en la tienda, debe asignar plantillas JSP a los grupos de catálogo. Puede asignar una plantilla específica de página de visualización a un grupo de catálogo o una plantilla por omisión para visualizar todos los grupos de catálogo. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, asigne plantillas de grupo de catálogo añadiendo información a la tabla DISPCGPREL. Complete esta tarea para cada plantilla que desee asignar a los grupos de catálogo:

```
<dispcgprel
catgroup_id="@catgroup_id_1"
devicefmt_id="-1"
dispcgprel_id="@dispcgprel_id_1"
mbrgrp_id="0"
pagename="/ToolTech/CategoryDisplay.jsp"
storeent_id="@storeent_id_1"
rank="0"/>
```

donde

- catgroup_id es el número de referencia del grupo de catálogo para el que se utilizará este nombre de página. El valor 0 indica que este nombre de página se utilizará para todos los grupos de catálogo.
- devicefmt_id es el número de referencia del tipo de dispositivo en el que se visualizará la página. El valor -1 indica que esta página de plantilla la utilizará un navegador HTTP.
- dispcgprel_id es el número de referencia de esta entrada.
- mbrgrp_id es el número de referencia del grupo de miembros para el que se utilizará esta página de plantilla. El valor 0 indica que esta página de plantilla se utilizará para todos los grupos de miembros.
- pagename es el nombre y la vía de acceso a la página de la plantilla de visualización.
- rank es un número de secuencia que se utiliza cuando más de una página satisface los criterios de selección.

Nota: Cada vez que asigna una plantilla JSP a un grupo de catálogo, `catentry_id` cambia la secuencia para que coincida con una entrada de catálogo existente.

2. Para poder visualizar las entradas de catálogo (productos, artículos, paquetes, paquetes compuestos y kits dinámicos) en la tienda, debe asignar plantillas JSP a las entradas de catálogo. Puede asignar una plantilla por omisión para visualizar todas las entradas de catálogo o una plantilla por omisión para visualizar cada tipo de entrada de catálogo; por ejemplo, una plantilla para los productos y otra para los artículo, o una plantilla específica para una entrada de catálogo específica. Utilizando el siguiente ejemplo de la tienda ToolTech como muestra, asigne plantillas añadiendo información a la tabla DISPENTREL. Complete esta tarea para cada plantilla que desee asignar a las entradas de catálogo:

```
<dispentrel
auctionstate="0"
catentry_id="0"
catenttype_id="ProductBean"
devicefmt_id="-1"
dispentrel_id="@dispentrel_id_1"
mbrgrp="0"
pagename="/ToolTech/ProductDisplay.jsp"
storeent_id="@storeent_id_1"
rank="0"/>
```

donde

- `auctionstate` indica que esta página de plantilla muestra una entrada de catálogo que está en una subasta:
 - 0=No es una plantilla de subasta.
 - 1=Plantilla de subasta.
- `catentry_id` es el número de referencia de la entrada de catálogo para la que se utilizará este nombre de página. El valor 0 indica que este nombre de página se utilizará para todas las entradas de catálogo.
- `catenttype_id` es el tipo de entrada de catálogo para el que se utilizará esta página para su visualización:
 - ProductBean=Muestra un producto.
 - ItemBean=Muestra un artículo.
 - PackageBean=Muestra un paquete.
 - BundleBean=Muestra un paquete compuesto.
 - DynamicKitBean=Muestra un kit dinámico.
- `devicefmt_id` es el número de referencia del tipo de dispositivo en el que se visualizará la página. El valor -1 indica que esta página de plantilla la utilizará un navegador HTTP.
- `dispentrel_id` es el número de referencia de la entrada de catálogo.
- `mbrgrp` es el número de referencia del grupo de miembros para el que se utilizará esta página de plantilla. El valor 0 indica que esta página de plantilla se utilizará para todos los grupos de miembros.
- `pagename` es el nombre y la vía de acceso a la página de la plantilla de visualización.
- `storeent_id` es el número de referencia de la tienda para la que se utilizará esta página.
- `rank` es un número de secuencia que se utiliza cuando más de una página satisface los criterios de selección.

Nota: Cada vez que asigna una plantilla JSP a una entrada de catálogo, `catentry_id` cambia la secuencia para que coincida con una entrada de catálogo existente.

Creación de un catálogo de navegación

Una tienda de WebSphere Commerce permite dos tipos de catálogos: maestro y de navegación. Los catálogos de navegación no es necesario que satisfagan las restricciones estructurales que sí deben satisfacer los catálogos maestros. Estos catálogos están diseñados para proporcionar una estructura de visualización flexible que le permita crear catálogos de navegación que se adapten a los requisitos de la tienda.

En particular, los catálogos de navegación no es necesario que satisfagan las restricciones siguientes, que sí deben satisfacer los catálogos maestros:



- Un catálogo maestro debe ser un árbol apropiado, lo que significa que no hay ciclos y no se puede utilizar la estructura siguiente: La categoría padre A tiene una subcategoría B. Es importante que B y cualquiera de las subcategorías de B no sean la categoría padre de A.
- Un producto no puede pertenecer a más de una categoría.





Las tareas siguientes crean un catálogo de navegación modificando el catálogo de la tienda de ejemplo NewFashion. El catálogo resultante no puede seguir clasificándose como catálogo maestro puesto que los pasos siguientes introducen ciclos de categorías y la clasificación de algunos productos en varias categorías. Un catálogo de navegación clásico se crea añadiendo información a las tablas de relaciones de categoría: `CATGRPREL`, que contiene las relaciones entre subcategorías y `CATGPENREL`, que contiene las relaciones entre categorías y productos. Aunque estos ejemplos incluyen NewFashion, puede seguir estos pasos básicos con su propio catálogo maestro, realizando los ajustes necesarios para que coincidan con la información, la estructura y los diseños de su catálogo.

Creación de ciclos de categoría

El catálogo de NewFashion contiene cuatro categorías superiores: **Moda para caballero**, **Moda para señora**, **Novedades** y **Promociones**. Este ejemplo le muestra cómo copiar la categoría **Novedades** en **Moda para caballero** y copiar **Promociones** en **Moda para señora** y, a continuación cambiar el nombre por **Novedades**.

Para cambiar el catálogo maestro de la tienda de ejemplo NewFashion y convertirlo en un catálogo de navegación utilizando ciclos de categoría, realice lo siguiente:

1. Publique el archivador de tienda de NewFashion para crear la tienda de ejemplo NewFashion. NewFashion está disponible en inglés de EE.UU. y en uno de los nueve idiomas nacionales que se entregan con WebSphere Commerce. Elija uno de los archivos `NewFashion_en_US_entorno_nacional.sar` para la publicación.
2. Abra el archivo `catalog.xml` en un editor. El archivo se encuentra en el siguiente directorio de WebSphere Commerce:
 -  `unidad:\WebSphere\CommerceServer\samplstores\NewFashion\entorno_nacional\data`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores\NewFashion\entorno_nacional\data`

-  /usr/WebSphere/CommerceServer/samplestores/NewFashion/entorno_nacional/data
 -   /opt/WebSphere/Commerce/samplestores/NewFashion/entorno_nacional/data
 -  /qibm/proddata/WebCommerce/samplestores/NewFashion/entorno_nacional/data
3. Localice la sección de datos CATGRPREL del archivo catalog.xml. Cree una nueva relación de categoría superior entre **Moda para caballeros** y **Novedades**. Actualmente, ambas categorías están en el nivel superior. Para una relación de navegación, cree una sección nueva que coloque **Novedades** como subcategoría de **Moda para caballeros** al mismo tiempo que conserva la estructura original. Bajo la sección CATGRPREL, añada el fragmento siguiente:

```
<catgrprel
catgroup_id_parent="@catgroup_id_11"
catgroup_id_child="@catgroup_id_21"
catalog_id="@catalog_id_1"
sequence="7"
/>
```

donde

- catgroup_id_parent es el número de referencia interno del grupo de catálogo de la categoría padre definido por la tienda de ejemplo NewFashion. En este ejemplo, @catgroup_id_11 es la categoría **Moda para caballeros**.
 - catgroup_id_child es el número de referencia interno del grupo de catálogo de la categoría hijo definido por la tienda de ejemplo NewFashion. En este ejemplo, @catgroup_id_21 es la categoría **Novedades**.
 - catalog_id es el número de referencia interno del catálogo definido por la tienda de ejemplo NewFashion.
 - sequence es el número que determina el orden de visualización del contenido del grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, la categoría **Novedades** se visualizará en último lugar, después de las seis primeras subcategorías **Moda para caballeros**.
4. Repita el paso anterior, esta vez creando una nueva relación de categoría superior entre **Moda para señoras** y **Promociones de la página de presentación**. Actualmente, ambas categorías están en el nivel superior. Para una relación de navegación, cree una sección nueva que coloque **Promociones de la página de presentación** como subcategoría de **Moda para señoras** al mismo tiempo que conserva la estructura original. Bajo la sección CATGRPREL, añada el fragmento siguiente:

```
<catgrprel
catgroup_id_parent="@catgroup_id_20"
catgroup_id_child="@catgroup_id_22"
catalog_id="@catalog_id_1"
sequence="9"
/>
```

donde







- catgroup_id_parent es el número de referencia interno del grupo de catálogo de la categoría padre definido por la tienda de ejemplo NewFashion. En este ejemplo, @catgroup_id_20 es la categoría **Moda para señoras**.
- catgroup_id_child es el número de referencia interno del grupo de catálogo de la categoría hijo definido por la tienda de ejemplo NewFashion. En este ejemplo, @catgroup_id_22 es la categoría **Promociones de la página de presentación**.

- `catalog_id` es el número de referencia interno del catálogo definido por la tienda de ejemplo NewFashion.
 - `sequence` es el número que determina el orden de visualización del contenido del grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, la categoría **Promociones de la página de presentación** se visualizará en último lugar.
5. Ahora que **Promociones de la página de presentación** es una subcategoría en **Moda para señoras**, el nombre de categoría es incorrecto. Cambie el nombre de categoría por **Novedades**, para que coincida con la nueva subcategoría de **Moda para caballeros**.
 6. Guarde el archivo `catalog.xml`.
 7. Para ver los cambios, realice una de las acciones siguientes: publique el archivador de tienda modificado de NewFashion mediante los Servicios de tienda o cargue el archivo `catalog.xml` con Loader package siguiendo las instrucciones del apartado “Carga de grupos de elementos de base de datos” en la página 287.

Adición de un producto a una segunda categoría

Este ejemplo le muestra cómo copiar productos de una categoría a otra al mismo tiempo que se conserva la estructura original. La categoría **Promociones de la página de presentación** contiene el producto **Camisón de verano**, que también podría pertenecer a la subcategoría **Pijamas** de la categoría superior **Moda para señoras**. Estas instrucciones le mostrarán cómo copiar el producto **Camisón de verano** y sus códigos de artículo en la categoría **Pijamas**.

Para cambiar el catálogo maestro de la tienda de ejemplo NewFashion y convertirlo en un catálogo de navegación añadiendo un producto a una segunda categoría, realice lo siguiente:

1. Publique el archivador de tienda de NewFashion para crear la tienda de ejemplo NewFashion. NewFashion está disponible en inglés de EE.UU. y en uno de los nueve idiomas nacionales que se entregan con WebSphere Commerce. Elija uno de los archivos `NewFashion_en_US_entorno_nacional.sar` para la publicación.
2. Abra el archivo `catalog.xml` en un editor. El archivo se encuentra en el siguiente directorio de WebSphere Commerce:
 -  `unidad:\WebSphere\CommerceServer\samplstores\NewFashion\entorno_nacional\data`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores\NewFashion\entorno_nacional\data`
 -  `/usr/WebSphere/CommerceServer/samplstores/NewFashion/entorno_nacional/data`
 -   `/opt/WebSphere/CommerceServer/samplstores/NewFashion/entorno_nacional/data`
 -  `/qibm/proddata/WebCommerce/samplstores/NewFashion/entorno_nacional/data`
3. Localice la sección de datos CATGPENREL del archivo `catalog.xml`. Cree una entrada de producto nueva para **Camisón de verano**, que originalmente era un producto bajo la categoría **Promociones de la página de presentación**. Bajo la sección CATGPENREL, añada el fragmento siguiente para incluir el producto:

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@product_id_2692"
sequence="2"
/>
```

donde

- `catgroup_id` es el número de referencia interno de grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, `@catgroup_id_18` es la categoría **Pijamas**.
 - `catalog_id` es el número de referencia interno del catálogo definido por la tienda de ejemplo NewFashion.
 - `catentry_id` es el número de referencia interno de entrada de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, `@catentry_id_2692` es el producto **Camisón de verano**.
 - `sequence` es el número que determina el orden de visualización del contenido del grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, el producto **Camisón de verano** se visualizará en último lugar.
4. Después de añadir la entrada de producto **Camisón de verano**, añade las entradas de código de artículo para el producto bajo la sección CATGPENREL, tal como se han definido en la tienda de ejemplo NewFashion. Actualmente, el producto **Camisón de verano** contiene diez códigos de artículo definidos. Bajo la sección CATGPENREL, añade los fragmentos siguientes para incluir los códigos de artículo:

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2695"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2696"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2697"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2698"
sequence="2"
/>
```

```
<catgpenrel
catgroup_id="@catgroup_id_18"
catalog_id="@catalog_id_1"
catentry_id="@catentry_id_2699"
sequence="2"
/>
```

```
<catgpenrel
```

```
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2700"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2701"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2702"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2703"  
sequence="2"  
</>
```

```
<catgpenrel  
catgroup_id="@catgroup_id_18"  
catalog_id="@catalog_id_1"  
catentry_id="@catentry_id_2704"  
sequence="2"  
</>
```

donde




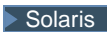


- `catgroup_id` es el número de referencia interno de grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, `@catgroup_id_18` es la categoría **Pijamas**.
- `catalog_id` es el número de referencia interno del catálogo definido por la tienda de ejemplo NewFashion.
- `catentry_id` es el número de referencia interno de entrada de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, `@catentry_id_2695` a `@catentry_id_2704` representan los diez códigos de artículo que se han definido para el producto **Camisión de verano**.
- `sequence` es el número que determina el orden de visualización del contenido del grupo de catálogo definido por la tienda de ejemplo NewFashion. En este ejemplo, los códigos de artículo de **Camisión de verano** se visualizarán en último lugar.

5. Guarde el archivo `catalog.xml`.
6. Para ver los cambios, realice una de las acciones siguientes: publique el archivador de tienda modificado de NewFashion mediante los Servicios de tienda o cargue el archivo `catalog.xml` con Loader package siguiendo las instrucciones del apartado “Carga de grupos de elementos de base de datos” en la página 287.

Gestión de elementos de catálogo en WebSphere Commerce

Con el tiempo, tendrá que actualizar la información de los elementos de base de datos en el catálogo maestro. El mantenimiento del catálogo es un proceso permanente, ya que de forma continuada necesitará añadir y eliminar mercancías, crear y asociar categorías o grupos de catálogos y actualizar información de productos como, por ejemplo, descripciones y precios.

Puede cambiar los elementos de catálogo editando los datos XML de WebSphere Commerce utilizando las entradas de base de datos existentes y los archivos `catalog.xml` de la tienda. Como referencia, utilice los archivos XML de la tienda de ejemplo de WebSphere Commerce que se encuentran en el siguiente directorio de datos:

-  `unidad:\WebSphere\CommerceServer\samplstores`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores`
-  `/usr/WebSphere/CommerceServer/samplstores`
-   `/opt/WebSphere/CommerceServer/samplstores`
-  `/qibm/proddata/WebCommerce/samplstores`

Nota: Estos ejemplos proceden de la tienda de ejemplo NewFashion e identifican los elementos XML que deben modificarse para cambiar la información de los elementos de catálogo.

Grupos de catálogo

Los grupos de catálogo se crean en un catálogo de WebSphere Commerce utilizando las tablas de base de datos `CATGROUP` y `CATGRPDESC`. En el archivo `catalog.xml`, un grupo de catálogo típico se parece al extracto siguiente:

```
<catgroup
catgroup_id="@catgroup_id_1"
member_id="%MEMBER_ID"
identifier="Accesorios"
markfordelete="0"
/>
```

`catgroup_id` es el número de referencia interno del grupo de catálogo. A cada grupo de catálogo se le asigna un número de referencia interno en WebSphere Commerce, que identifica el grupo cuando se añaden entradas de catálogo. `identifier` es un nombre externo para el grupo de catálogo. Ambos elementos son exclusivos en los elementos de base de datos y no se pueden duplicar.

Los nombres y las descripciones pertenecen al archivo `catalog.xml` específico de entorno nacional; se necesita un archivo para cada entorno nacional soportado por la tienda. Un grupo de catálogo típico que contenga información traducible se parece al extracto siguiente:

```
<catgrpdesc
language_id="%es_ES"
catgroup_id="@catgroup_id_1"
name="Accesorios"
shortdescription="Accesorios"
longdescription="Accesorios"
published="1"
/>
```

language_id identifica el idioma en que está la información del catálogo. Este identificador debe cambiar y coincidir con cada uno de los idiomas soportados por la tienda. El cliente ve el elemento name, al igual que los elementos shortdescription y longdescription, que pueden contener descripciones breves y detalladas del grupo de catálogo.

Cuando cree un nuevo grupo de catálogo, siga la estructura anterior para la información.

Notas:

1. Aunque los elementos identifier y name sean idénticos en el ejemplo anterior, el contenido puede variar. Por ejemplo, quizá quiera renombrar el grupo de catálogo a **Accesorios complementarios**. En ese caso, no es necesario cambiar la información en identifier, sólo name.
2. Al suprimir grupos de catálogo, asegúrese de actualizar las apariciones de catgroup_id. Por ejemplo, si también quiere suprimir las entradas de catálogo bajo el grupo de catálogo, elimine todas las entradas XML. Sin embargo, si piensa mantener las entradas de catálogo, deberá cambiar catgroup_id al grupo correcto.

Entradas de catálogo

Las entradas de catálogo se crean en un catálogo de WebSphere Commerce utilizando la información de las tablas de base de datos CATENTRY y CATENTDESC. Una entrada de catálogo puede ser un producto, un artículo, un paquete, un paquete compuesto o un kit dinámico. En el archivo catalog.xml, una entrada de catálogo típica se parece al extracto siguiente:

```
<catentry
  catentry_id="@product_id_102"
  baseitem_id="@baseitem_id_102"
  member_id="&MEMBER_ID"
  catenttype_id="ProductBean"
  partnumber="product-sku-nf-102"
  mfpartnumber="product-sku-nf-102"
  mfname="NewFashion"
  markfordelete="0"
  buyable="1"
/>
```

catentry_id es el número de referencia interno de la entrada de catálogo de productos. baseitem_id es el artículo base con el que esta entrada de catálogo está relacionada, a efectos de inventario. partnumber es el número de referencia que identifica el número de pieza de la entrada de catálogo. mfpartnumber es el número de pieza que utiliza el fabricante para identificar la entrada de catálogo. Estos elementos son exclusivos en los elementos de base de datos y no se pueden duplicar.

catenttype_id identifica el tipo de entrada de catálogo: ItemBean, ProductBean, PackageBean, BundleBean o DynamicKitBean.

Los nombres y las descripciones pertenecen al archivo catalog.xml específico de entorno nacional; se necesita un archivo para cada entorno nacional soportado por la tienda. Las imágenes de los productos también se incluyen en este archivo. Un grupo de catálogo típico que contenga información traducible se parece al extracto siguiente:

```
<catentdesc
  catentry_id="@product_id_102"
  language_id="&es_ES"
```



```

name="Cinturón"
shortdescription="Cinturón clásico"
longdescription="Este cinturón clásico es un complemento ideal para sus vaqueros favoritos o para ir elegante al trabajo. En piel, de 3,5 cm de ancho y con hebilla metálica."
thumbnail="images/mens_accessories_belt_sm.gif"
fullimage="images/mens_accessories_belt.gif"
available="1"
published="1"
/>

```

language_id identifica el idioma en que está la información del catálogo. Este identificador debe cambiar y coincidir con cada uno de los idiomas soportados por la tienda. El cliente ve el elemento name, al igual que los elementos shortdescription y longdescription, que pueden contener descripciones breves y detalladas de la entrada de catálogo.

Cuando cree una nueva entrada de catálogo, siga la estructura anterior para la información.

Notas:

1. Al suprimir entradas de catálogo, asegúrese de actualizar cada aparición de los elementos exclusivos. Por ejemplo, si también quiere suprimir las entradas de catálogo bajo el grupo de catálogo, elimine todas las entradas XML. Sin embargo, si piensa mantener las entradas de catálogo, deberá cambiar catgroup_id al grupo correcto.
2. Los productos deben crearse antes que otros tipos de entradas de catálogo.

Si no desea cambiar manualmente los archivos XML, elija uno de los métodos siguientes: las herramientas de Gestión de productos o los programas de utilidad de Catalog Manager.

Herramientas de Gestión de productos

Las herramientas de Gestión de productos de WebSphere Commerce Accelerator le permiten gestionar los productos del catálogo maestro de la tienda utilizando los diferentes asistentes y cuadernos. Puede actualizar el contenido del catálogo o crear nuevos datos de catálogo; por ejemplo, puede:

- Crear, actualizar y eliminar productos y detalles de productos mediante el asistente o el cuaderno. Los productos actúan como plantillas para los códigos de artículo, los elementos individuales que finalmente se venden al cliente. Los detalles del producto incluyen el código del producto (que identifica de forma exclusiva el producto), el nombre del producto y la descripción, las opciones de comercialización (por ejemplo, mostrar un producto a los clientes o indicar que forma parte de una promoción especial), las imágenes del producto, las especificaciones de impuestos y envío, los descuentos asignados a los productos y la información del fabricante.
- Generar, actualizar y eliminar los códigos de artículo. Los códigos de artículo representan cada uno de los artículos que pueden solicitarse de la mercancía que está en venta. Todos los códigos de artículo que están relacionados con un producto determinado presentan el mismo conjunto de atributos y se distinguen por los valores de sus atributos. Las modificaciones o adiciones que se pueden realizar en los códigos de artículo incluyen la misma información que la de los productos, excepto en lo relacionado con los pedidos.
- Crear, actualizar y eliminar categorías (o grupos de catálogos), que son un grupo de objetos con propiedades similares que se utilizan para organizar los productos o servicios que ofrece la tienda. Puede gestionar la jerarquía de

categorías de su catálogo maestro creando, modificando y eliminando categorías y los detalles de las categorías como, por ejemplo, el código, el nombre y la descripción de la categoría, incluida la categoría padre y las imágenes.

- Asociar productos y códigos de artículo con categorías, seleccionando la categoría padre o trasladando los productos y códigos de artículo de una categoría a otra.
- Crear atributos y valores de atributos para los productos. Cada combinación de atributos y valores de atributos posibles es igual a un nuevo código de artículo. Deberá predefinir previamente los valores de los atributos antes de asignarlos a los códigos de artículo. Después de crear atributos y sus valores, puede crear o actualizar información como, por ejemplo, el nombre, la descripción, (el texto, números enteros o números decimales) y la secuencia en la que aparecerán los atributos y los valores de atributos.
- Crear, actualizar, suprimir y asociar la fijación de precios del catálogo con los productos. Puede definir un precio para un producto o código de artículo, en una o varias monedas, junto con un conjunto de condiciones como, por ejemplo, establecer un precio para una cantidad unitaria o para cantidades generales, que deben satisfacerse para poder utilizar el pedido.

Puede consultar el apartado de Gestión de productos de la ayuda en línea si desea obtener instrucciones detalladas acerca de cada tarea.

Notas:

1. Las herramientas de Gestión de productos se recomiendan solamente para cambios ligeros. Para las actualizaciones importantes del catálogo, por ejemplo, para añadir o suprimir los productos de temporada o para preparar una venta de liquidación total, utilice Loader package.
2. Las modificaciones que se realicen en los datos del catálogo no se pueden mostrar en la tienda a menos que inhabilite el almacenamiento en antememoria o suprima las páginas JSP que actualmente hay en la antememoria. Para obtener más información, consulte el mandato CacheDelete en la ayuda en línea de WebSphere Commerce. El mandato CacheDelete inicia la limpieza remota de la antememoria de páginas dinámicas y permite gestionar la antememoria sin que sea necesario acceder directamente al sistema de archivos. Antes de utilizar este mandato compruebe que la invalidación de página automática esté habilitada. Tenga en cuenta que para utilizar este mandato debe haber iniciado la sesión como administrador de sitio o como desarrollador de tiendas.

Catalog Manager

También puede efectuar el mantenimiento del catálogo utilizando Loader package o Web Editor, dos programas de utilidad que están disponibles en Catalog Manager. Loader package resulta ideal para importar a la base de datos grandes cantidades de información de productos existentes. En WebSphere Commerce, esta es la herramienta principal para crear y gestionar información sobre catálogos. Este paquete consta principalmente de utilidades de mandatos para preparar y cargar datos en la base de datos de WebSphere Commerce. Loader package también permite extraer datos de una base de datos como un documento XML, transformar los datos XML en formatos XML alternativos y transformar datos entre un formato de variables delimitadas por caracteres y un formato de datos XML.

Web Editor utiliza una interfaz de navegador Web, en la que puede crear, editar o suprimir datos del catálogo. Los formularios de entrada de catálogo para ver y actualizar información son la base de Web Editor. En el caso más sencillo, los formularios se corresponden con tablas de la base de datos de WebSphere

Commerce. El administrador puede optar por utilizar los formularios que se proporcionan o por personalizarlos. Consulte la publicación *IBM WebSphere Commerce Versión 5.4, Guía del usuario de Catalog Manager* si desea obtener más información.

catálogo y los criterios, como por ejemplo la cantidad que va a comprar, que el cliente debe satisfacer para pagar dicho precio. Por ejemplo, a menudo, la mercancía o los servicios tienen precios distintos para los niños, los estudiantes, los adultos o los jubilados. En WebSphere Commerce, una oferta también se denomina propuesta de comercio y forma parte de un contenedor de propuestas de comercio.

Precio de oferta

El *precio de oferta* es el precio al que se ofrecen las entradas de catálogo de una tienda, mediante contratos o acuerdos comerciales. Una oferta puede tener uno o varios precios de oferta definidos en varias monedas.

Contenedor de propuestas de comercio

Una oferta forma parte de un *contenedor de propuestas de comercio*, que es propiedad de un miembro. Un contenedor de propuestas de comercio contiene las propuestas de comercio. Puede ponerse a disposición de todos los clientes o sólo de clientes de determinados grupos mediante acuerdos comerciales o contratos y los términos y condiciones de los contratos. Bajo un contrato, un contenedor de propuestas de comercio es un objeto de negocio de precio al que pueden hacer referencia varias políticas de negocio de precios y que pueden compartir una o todas las tiendas de un grupo de tiendas. Un contenedor de propuestas de comercio también se denomina *lista de precios*.

Términos y condiciones

Los *términos y condiciones* definen el comportamiento y las propiedades de un acuerdo de comercio. Muchos términos y condiciones hacen referencia a políticas de negocio ya que éstas definen varios aspectos del funcionamiento de una tienda.

Tipos de términos y condiciones de fijación de precios

Los *términos y condiciones de fijación de precios* definen qué productos están disponibles bajo un contrato y qué precios pagará el cliente por los productos. En un contrato se necesita como mínimo uno de los siguientes términos de fijación de precios:

- Una *lista de precios personalizada* especifica que tanto la lista de productos en venta como sus precios están personalizados para su venta en un contrato. Los artículos no están limitados a una sección del catálogo de tienda, pueden pertenecer a cualquier parte del catálogo de tienda.
- Un *catálogo completo con términos de ajuste* ofrece en venta pero con un porcentaje todos los productos que hay disponibles en el catálogo de una tienda. Si no se especifica ningún ajuste, los artículos se venden al precio base.
- Una *lista de precios con términos de ajuste* ofrece en venta pero con un ajuste de porcentaje todos los productos que hay disponibles en una lista de precios. Si no se especifica ningún ajuste, los artículos se venden al precio base.
- Una *lista de precios con términos de ajuste selectivos* es similar a la lista de precios con ajuste, excepto que el ajuste no se aplica a toda la lista de precios. El ajuste se realiza en un subconjunto de la lista de precios. El subconjunto de la lista de precios puede ser una política de negocio de conjunto de productos o un conjunto de productos personalizado.

Acuerdo de comercio

Un *acuerdo de comercio* puede ser un contrato, un RFQ, una cuenta de negocio o una subasta. Un acuerdo de negocio es un acuerdo que se negocia entre un vendedor y un comprador bajo el cual el comprador puede comprar determinados

artículos bajo los términos y condiciones especificados y las políticas de negocio que estipula el contrato. Por ejemplo, permite al cliente comprar productos de una tienda al precio especificado durante un periodo de tiempo especificado, y bajo los términos y condiciones de fijación de precios. En WebSphere Commerce, todos los clientes deben comprar en una tienda bajo un contrato. Una tienda puede difundir uno o varios contratos y se puede designar uno de ellos como el contrato por omisión. Un contrato por omisión contiene un conjunto de términos y condiciones asociados a un conjunto de políticas por omisión de una tienda. Un acuerdo de comercio puede contener cero o más participantes con roles diferentes.

Participante

Un *participante* puede formar parte de un acuerdo de comercio o de términos y condiciones. Un participante es un miembro y puede tratarse de un grupo de miembros, de una organización, etc. Si en un contrato se especifica un participante cuyo rol es el de comprador, para poder comprar bajo ese contrato el comprador debe ser miembro del participante que realiza las compras. Los términos y condiciones del contrato también pueden contener cero o varios participantes.

Rol de participante

Un participante puede tener los siguientes *roles de participantes*:

- Creador
- Vendedor
- Comprador
- Proveedor
- Aprobador
- Titular de la cuenta
- Persona de contacto del comprador
- Persona de contacto del vendedor
- Representante legal
- Administrador.

Contrato

Un *contrato* contiene el precio de oferta del producto. En WebSphere Commerce, todos los clientes deben comprar bajo un contrato. Un contrato permite al cliente adquirir productos de una tienda al precio especificado durante un periodo de tiempo especificado, y bajo los términos y condiciones, y políticas de negocio, estipulados en el contrato. Una tienda puede poseer varios contratos o no poseer ninguno pero, como mínimo, debe poseer un contrato por omisión.

Política de negocio

Las *políticas de negocio* son conjuntos de normas seguidas por una tienda o grupo de tiendas que definen los procesos de negocio, las prácticas del sector y el ámbito y las características de las ofertas de una tienda o grupo de tiendas. Las políticas de negocio se imponen con una combinación de uno o varios mandatos de política de negocio que implementan las normas de la política de negocio, una referencia a un objeto de negocio sobre el que actúan las normas y un conjunto de propiedades que configuran el funcionamiento de los mandatos de política de negocio.

Política de fijación de precios

Una *política de fijación de precios* contiene una referencia a una lista de precios y se puede asociar con varios mandatos de la política de negocio que se implementarán

en las listas de precios. La política se puede definir para una tienda o grupo de tiendas. Si la política se registra para un grupo de tiendas, entonces todas las tiendas de dicho grupo podrán utilizar la política.

Envío de entrada de catálogo

La información de *envío de entrada de catálogo* incluye información acerca de cómo se empaqueta el producto para su envío. Cada entrada de catálogo puede tener definida diferentes tipos de información de envío. Por ejemplo, la altura, el peso y la longitud del producto una vez empaquetado.

Otros elementos asociados al cálculo de precio

Los elementos siguientes son elementos asociados al cálculo de precio:

- Un *miembro* que es el propietario del contenedor de propuestas de comercio. Un contenedor de propuestas de comercio sólo tiene un propietario.
- Una *entidad de tienda* representa una tienda en la base de datos de WebSphere Commerce Server.
- Un *catálogo* contiene entradas de catálogo a las que se hará referencia en un contrato. El catálogo contiene toda la información jerárquica y de navegación para el catálogo en línea y es un conjunto de grupos de catálogo y de entradas de catálogo que se pueden visualizar y adquirir en una tienda en línea.
- Un *grupo de catálogo*, o categoría, son agrupaciones genéricas de las entradas de catálogo, que se crean para facilitar la navegación y partición del catálogo. Un grupo de catálogo pertenece a un catálogo y puede contener más de un grupo de catálogo o entradas de catálogo. Puede asociar grupos de catálogo a más de un catálogo.
- Las *entradas de catálogo* representan mercancía que se puede solicitar en un catálogo en línea. Las entradas de catálogo pertenecen a grupos de catálogo. Una oferta siempre está asociada a una entrada de catálogo.



Para obtener información más detallada sobre la estructura de los elementos del cálculo de precio en WebSphere Commerce Server, consulte los modelos de datos y de objetos de cálculo de precio en la ayuda en línea de WebSphere Commerce.

Creación de elementos de cálculo de precio en WebSphere Commerce

Tiene dos opciones para crear los elementos de cálculo de precio:

- Crear precios utilizando las herramientas de Gestión de productos de WebSphere Commerce Accelerator. El uso de las herramientas de WebSphere Commerce Accelerator es más conveniente en la creación de precios de catálogos muy pequeños.
- Crear precios en un archivo XML, que se puede cargar mediante WebSphere Commerce Loader package, o como parte de un archivador de tienda que se puede publicar a través de los Servicios de tienda. Este método es más adecuado para crear grandes cantidades de datos.







Para obtener más información sobre la creación de precios utilizando las herramientas de Gestión de productos de WebSphere Commerce Accelerator, consulte la ayuda en línea de WebSphere Commerce. Para obtener más información sobre la creación de precios en un archivo XML, consulte “Creación de elementos de cálculo de precio en un archivo XML” en la página 95.

Creación de elementos de cálculo de precio en un archivo XML

Cree los elementos de cálculo de precio en formato de archivos XML que se puedan cargar en la base de datos utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, "Publicación de la tienda" en la página 225.

1. Revise los archivos XML que se utilizan para crear elementos de cálculo de precio para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.







Los archivos de archivador de tienda se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\samplestores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye dos archivos `offering.xml`, que incluyen la información sobre el cálculo de precios. Para ver los archivos `offering.xml` en el archivador de tienda, descomprímalo utilizando un programa ZIP. Los archivos `offering.xml` se encuentran en el directorio de datos. El archivo `offering.xml` específico de un idioma está situado en un subdirectorio específico del entorno nacional del directorio de datos.

2. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
3. Cree un archivo `offering.xml`, copiando uno de los archivos `offering.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a `offering.xml`. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Cree un contenedor de propuestas de comercio. Para ofrecer los precios de los artículos de su tienda, primero debe crear un contenedor de propuestas de comercio. Para crearlo, añada información a la tabla TRADEPOSCN.

- a. Utilice el ejemplo siguiente como guía y cree un contenedor de propuestas de comercio en su archivo XML, en la tabla TRADEPOSCN:

```
<tradeposcn
tradeposcn_id="@tradeposcn_id_101"
member_id="@MEMBER_ID"
```

```
markfordelete="0"
name="ToolTech"
precedence="0"

/>
```

donde

- tradeposcn_id es una clave exclusiva generada
 - member_id es el propietario del contenedor de propuestas de comercio.
 - markfordelete indica lo siguiente:
 - 0 = puede utilizarse el contenedor de propuestas de comercio
 - 1 = el contenedor de propuestas de comercio se ha marcado para suprimirlo (consulte el programa de utilidad DBClean) y no se debe utilizar
 - name es un nombre nemotécnico para el contenedor de propuestas de comercio, exclusivo de un propietario particular.
 - precedence se utiliza cuando se habilita más de un contenedor de propuestas de comercio en un momento determinado; en ese caso, se utiliza la propuesta cuyo valor de PRECEDENCE sea mayor.
5. Asocie el catálogo maestro a un contenedor de propuestas de comercio, añadiendo información a la tabla CATGRPTPC. Cuando asocia el catálogo maestro a un contenedor de propuestas de comercio, todas las entradas del catálogo maestro deben tener un precio estándar. Para obtener más información sobre la creación de catálogos maestros, consulte “Visualización de los elementos de catálogo de la tienda” en la página 77.
- a. Utilizando el ejemplo siguiente como muestra, asocie el catálogo maestro al contenedor de propuestas de comercio añadiendo información a la tabla CATGRPTPC:

```
<catgrptpc
catalog_id="@catalog_id_1"
tradeposcn_id="@tradeposcn_id_101"
/>
```

donde

- catalog_id es el catálogo maestro.
 - tradeposcn_id es el contenedor de propuestas de comercio.
6. Cree ofertas y precios de oferta para las entradas de catálogo añadiendo información en las tablas OFFER y OFFERPRICE
- a. Utilizando el ejemplo siguiente como muestra, cree una oferta para una entrada de catálogo añadiendo información a la tabla OFFER. Recuerde que para crear precios, debe haber creado entradas de catálogo. Para obtener más información sobre la creación de entradas de catálogo, consulte “Visualización de los elementos de catálogo de la tienda” en la página 77.

```
offer
offer_id="@offer_id_138"
startdate="2000-06-19 00:00:00.000000"
catentry_id="@product_id_102"
precedence="0"
published="1"
identifier="1"
flags="1"
tradeposcn_id="@tradeposcn_id_101"
/>
```

donde

- offer_id es una clave exclusiva generada.
 - startdate es el inicio del rango de fecha y hora durante el cual estará en vigor esta oferta.
 - catentry_id es la entrada de catálogo que se ofrece a la venta.
 - precedence: cuando hay más de una oferta en vigor en un momento determinado, se utiliza la que tiene el valor de PRECEDENCE más elevado.
 - published puede ser
 - 0 = no publicado (inhabilitado temporalmente)
 - 1 = publicado
 - 2 = marcado para suprimirlo (y no publicado).
 - identifier es un número que identifica de forma exclusiva a esta oferta junto con su entrada de catálogo especificada y contenedor de propuestas de comercio.
 - flags son
 - 1 = shiptoAddressRequired - si es 1, OrderPrepare devolverá un error si un artículo de pedido hace referencia a esta oferta pero no tiene una dirección de envío.
 - tradeposcn_id es el contenedor de propuestas de comercio del que forma parte esta oferta.
- b. Utilizando el ejemplo siguiente como muestra, cree un precio de oferta para una entrada de catálogo añadiendo información a la tabla OFFERPRICE. El precio de oferta es el precio real al que una entrada de catálogo se pone a la venta. Recuerde que para crear precios, debe haber creado entradas de catálogo. Para obtener más información sobre la creación de entradas de catálogo, consulte “Visualización de los elementos de catálogo de la tienda” en la página 77.

```
<offerprice
offer_id="@offer_id_138"
currency="EUR"
price="590.00"
/>
```

donde

- offer_id es la oferta asociada a este precio.
- currency es la moneda en que se ofrece el precio.
- price es el precio para la cantidad nominal (vea CATENTSHIP.NOMINALQUANTITY) del producto al que hace referencia la oferta.

Nota: Para mostrar varias monedas en su tienda, cree una entrada XML diferente en la tabla OFFERPRICE para cada moneda. Por ejemplo, para mostrar la moneda en dólares de Canadá, utilice currency="CAD" en una entrada XML nueva. El valor del precio cambiará para reflejar el precio en dólares de Canadá. O puede utilizar una conversión que permita al cliente ver los diferentes tipos impositivos según la moneda que seleccionen. Para obtener más información, consulte “Creación de elementos de moneda utilizando un archivo XML” en la página 127.

- c. Repita los pasos a y b para todas las entradas de su catálogo.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Capítulo 10. Elementos de contrato

En WebSphere Commerce, todos los clientes de una tienda deben comprar bajo un contrato. Un contrato permite a los clientes adquirir productos de una tienda al precio especificado durante el periodo de tiempo especificado, y bajo condiciones específicas. Al examinar el catálogo de una tienda, los clientes sólo verán los productos que cubran los contratos que tengan estipulados con la tienda.

Si desea que los clientes que no tengan ningún contrato con la tienda (por ejemplo, los compradores invitados) también puedan comprar, o que los clientes puedan comprar productos que no estén cubiertos por los contratos que tengan, su tienda necesitará un *contrato por omisión*.

Importante

WebSphere Commerce Professional Edition solamente da soporte al contrato por omisión de tienda.

Los contratos que no sean el contrato por omisión de tienda solamente están soportados por WebSphere Commerce Business Edition.

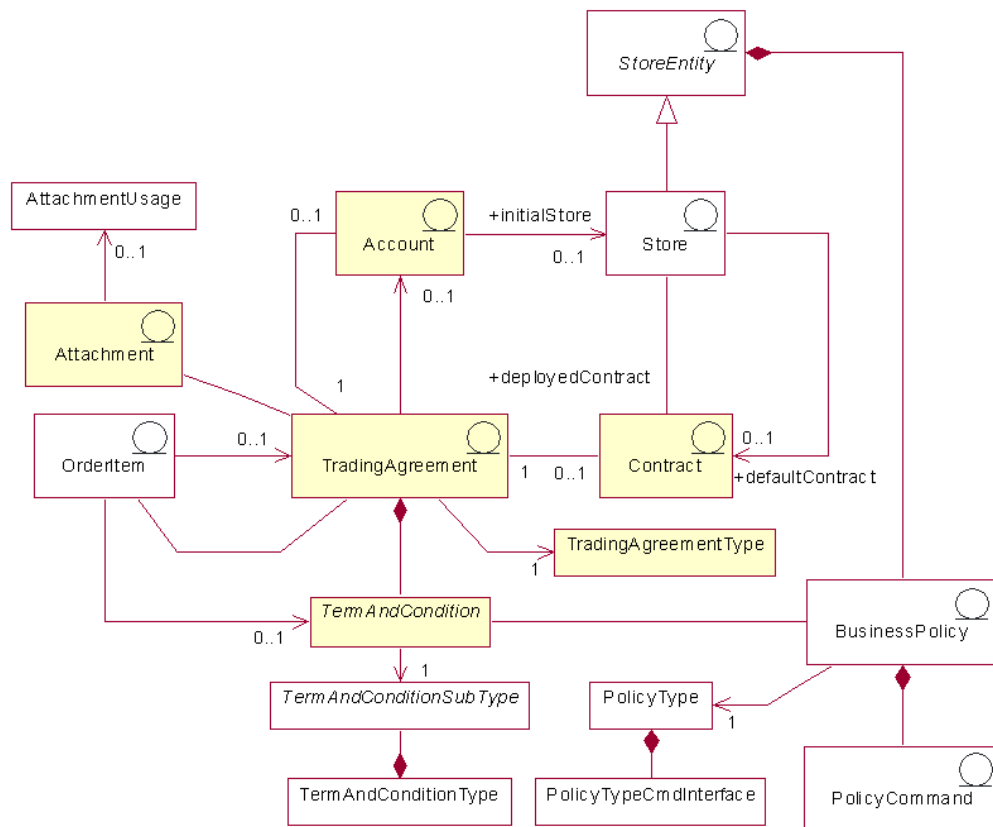
Para que todos los clientes puedan comprar en una tienda creada con WebSphere Commerce, ésta debe incluir lo siguiente:

- Políticas de negocio
- Contrato por omisión

El contrato por omisión hace referencia a las políticas de negocio, permitiendo que todos los clientes puedan comprar en una tienda.

Descripción de los contratos en WebSphere Commerce

El diagrama siguiente ilustra la estructura de los contratos en WebSphere Commerce:



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forman parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte el apartado “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Cuenta (cuenta de negocio)

Una cuenta de negocio representa la relación entre una organización compradora y una organización vendedora. Una cuenta de negocio puede utilizarse para organizar otros acuerdos de comercio y para especificar los términos y condiciones relativos a la relación entre el comprador y el vendedor, como por ejemplo, la personalización de las facturas, la verificación de la orden de compra o el mantenimiento de la línea de crédito del comprador con el vendedor.

Los contratos están asociados a las cuentas de negocio ya que representan un acuerdo entre el comprador y el vendedor. La excepción a esto es el contrato por omisión de la tienda, que no puede asociarse a una cuenta de negocio. Una cuenta de negocio puede tener muchos contratos asociados a ella.

Una cuenta de negocio es un tipo de acuerdo de comercio. Para obtener una descripción de los acuerdos de comercio, consulte el apartado “Acuerdo de comercio”.

Importante: los contratos de negocio sólo están soportados por WebSphere Commerce Business Edition.

Contrato

Hay dos tipos de contratos activos asociados a las tiendas: los contratos por omisión y los contratos desplegados. Los contratos desplegados autorizan a organizaciones compradoras o compradores individuales y se pueden crear utilizando WebSphere Commerce Accelerator después de crear la tienda. Un contrato desplegado se asocia a una cuenta de negocio. Un contrato por omisión define el comportamiento por omisión con los compradores que no tienen ningún otro contrato con la tienda. Un contrato por omisión solamente puede crearse utilizando archivos XML y sólo puede definirse un contrato por omisión para una tienda. Para obtener más información sobre los contratos, consulte la información en línea. Para obtener información sobre la creación de un elemento de contrato por omisión, consulte el apartado “Creación de un elemento de contrato por omisión en WebSphere Commerce” en la página 103.

Un contrato es un tipo de acuerdo de comercio. Para obtener una descripción de los acuerdos de comercio, consulte “Acuerdo de comercio”.

Acuerdo de comercio

WebSphere Commerce proporciona varios mecanismos de comercio que rigen las interacciones entre compradores y vendedores. Las diferentes ediciones de WebSphere Commerce dan soporte a los mecanismos de comercio siguientes:

- Subastas (soportadas tanto por Business Edition como por Professional Edition)
- Cuentas de negocio (soportadas solamente por Business Edition)
- Contratos (vea las limitaciones descritas anteriormente en este capítulo)
- Solicitud de presupuesto (RFQ) (soportada solamente por Business Edition)

Todos estos mecanismos de comercio tienen propiedades comunes. Por ejemplo, todos los mecanismos de comercio tienen participantes y todos tienen normas que rigen el comportamiento del mecanismo de comercio. Las normas que rigen el comportamiento de los mecanismos de comercio se conocen como *términos y condiciones* en WebSphere Commerce.

Un acuerdo de comercio representa una instancia de un mecanismo de comercio y registra las propiedades de esa instancia de un mecanismo de comercio. Cada contrato, cuenta de negocio y RFQ en WebSphere Commerce es un acuerdo de comercio. Hay un acuerdo de comercio que rige todas las subastas en WebSphere Commerce.

Un acuerdo de comercio consta de un perfil almacenado en la tabla TRADING; de participantes almacenados en la tabla PARTICIPANT; de términos y condiciones almacenados en la tabla TERMCOND; y de adjuntos opcionales almacenados como URI (Identificador de recursos universal) en la tabla ATTACHMENT. Puesto que un acuerdo de comercio puede tener múltiples adjuntos, éstos se relacionan con el acuerdo de comercio a través de la tabla TRDATTACH. Tenga en cuenta que las RFQ no dan soporte a los adjuntos.

Además del acuerdo de comercio general, cada tipo de acuerdo de comercio almacena información adicional específica del tipo de acuerdo de comercio en su propia tabla: CONTRACT almacena información específica de contratos; RFQ almacena información específica de RFQ; y ACCOUNT almacena información específica de cuentas.

Términos y condiciones

Los términos y las condiciones definen el comportamiento y las propiedades de un acuerdo de comercio.

En los contratos, los términos y condiciones definen cómo se implementa un contrato para una organización compradora. Definen qué se está vendiendo bajo el contrato, el precio de los artículos que se están vendiendo, cómo se envían los artículos, cómo se pagan los pedidos, cómo se manejan las devoluciones de artículos, cómo se aprueban los pedidos y desde dónde se envían los pedidos.

Muchos términos y condiciones hacen referencia a políticas de negocio ya que éstas definen muchos aspectos del funcionamiento de una tienda. Los términos y condiciones pueden proporcionar parámetros para las políticas de negocio a las que hacen referencia. El suministro de parámetros a las políticas de negocio le permite modificar el comportamiento de esas políticas para cada contrato.

Políticas de negocio

Las políticas de negocio son conjuntos de normas seguidas por una tienda o un grupo de tiendas. Las políticas de negocio definen los procesos de negocio, las prácticas del sector y el ámbito y las características de las ofertas de una tienda o grupo de tiendas. Son la plantilla de origen y referencia central para todas las prácticas permitidas y soportadas en una tienda o un grupo de tiendas.

En WebSphere Commerce, las políticas de negocio se imponen con una combinación de uno o más mandatos de política de negocio que implementan las normas de la política de negocio, una referencia a un objeto de negocio sobre el que actúan las normas y un conjunto de propiedades que configuran el funcionamiento de los mandatos de política de negocio. Los términos y condiciones pueden proporcionar parámetros para las políticas de negocio a las que hacen referencia. Esto permite modificar el comportamiento de la política de negocio según el término y la condición que hagan referencia a la política de negocio.

Adjunto

Un adjunto proporciona información adicional acerca de un acuerdo de negocio que no queda cubierto por los demás elementos del acuerdo de negocio. Un ejemplo es un archivo que proporciona información adicional acerca de las solicitudes de RFQ y cualquier observación general acerca de RFQ. Un acuerdo de negocio puede tener varios adjuntos. Los adjuntos se almacenan fuera de WebSphere Commerce y el acuerdo de negocio almacena los URI (Universal Resource Identifiers) asociados a los adjuntos. A continuación se muestran algunos ejemplos de URI:

- <http://www.miempresa.com/information/documento1.txt>
- [archivo:///home/usuariojoan/misdocs/documento1](file:///home/usuariojoan/misdocs/documento1)
- <ftp://ftp.miempresa.com/information/adjunto.txt>

Se puede asignar un uso a todos los adjuntos que indique la finalidad del adjunto. El uso de un adjunto es una propiedad opcional de un adjunto.

Artículo de pedido

Un artículo de pedido es un producto que se incluye en un pedido. En un mismo pedido, se pueden comprar diferentes artículos de pedido bajo distintos acuerdos de comercio de contrato. En función del diseño de la tienda, los compradores pueden seleccionar el acuerdo de comercio de contrato bajo el que comprarán, al principio del flujo de compra o cuando añaden un artículo al pedido. Cuando se compran artículos bajo acuerdos de comercio de contrato diferentes, se aplican las normas siguientes:

- Los acuerdos de comercio de contrato para todos los artículos de un pedido deben tener en común, como mínimo, un método de pago. Si el contrato para un artículo no tiene en común un método de pago, el comprador no puede añadir dicho artículo al pedido. Para pagar el pedido sólo se pueden utilizar los métodos de pago comunes a todos los artículos de un pedido.
- Todos los artículos de un pedido deben proceder de acuerdos de comercio de contrato que pertenezcan a la misma cuenta de negocio o al contrato por omisión de la tienda.



Para obtener información más detallada sobre la estructura de los elementos de contrato en WebSphere Commerce, consulte el modelo de datos de contrato en la ayuda en línea de WebSphere Commerce.

Creación de un elemento de contrato por omisión en WebSphere Commerce

El contrato por omisión define el comportamiento por omisión de una tienda. Al igual que con todos los contratos, puede establecer los productos disponibles, los métodos de pago, los métodos de envío y otro tipo de comportamiento de la tienda.

Los contratos por omisión de tienda que se proporcionan con las tiendas de ejemplo de WebSphere Commerce contienen términos y condiciones que especifican lo siguiente:







- Los clientes pueden comprar todos los productos disponibles en el catálogo maestro de la tienda a los precios estándar establecidos en dicho catálogo (sin descuentos ni recargos).
- Los costes de envío se deben pagar al vendedor (la tienda).
- Los clientes pueden devolver los artículos comprados sin recargos de penalización dentro de un periodo de tiempo especificado.
- Los clientes pueden recibir reembolsos utilizando el mismo método de pago usado para la compra original.

Además, la versión más general del contrato por omisión de una tienda omite los términos y condiciones que limitan los métodos de pago y envío que pueden utilizar los compradores. Al omitir estos términos los compradores pueden pagar sus compras utilizando cualquiera de los métodos de pago por omisión soportados por la tienda y utilizar cualquiera de los métodos de envío que haya disponibles en la tienda.

Las propiedades del contrato por omisión se definen en sus términos y condiciones. Algunos de los términos y condiciones hacen referencia a políticas de negocio. Para obtener más información sobre las políticas de negocio y los términos y condiciones, consulte la información en línea.







Para crear un elemento de contrato por omisión, efectúe lo siguiente:

1. Revise la información en línea sobre términos y condiciones, contratos, contratos por omisión y políticas de negocio.
2. Revise las políticas de negocio definidas en el archivo `wcs.bootstrap.xml`. Para obtener información sobre el archivo `wcs.bootstrap.xml`, consulte la información en línea.
3. Revise los archivos utilizados para crear elementos de contrato por omisión para las tiendas de ejemplo. Todos los archivos de las tiendas de ejemplo se encuentran en el archivador de tienda correspondiente. Cada tienda de ejemplo incluye un archivo `businesspolicy.xml` y un archivo `contract.xml`, que incluyen información adicional sobre políticas de negocio e información sobre el contrato por omisión. Los archivos de archivador de tienda se encuentran en el directorio siguiente:

-  `/usr/WebSphere/CommerceServer/samplestores`
-  `/qibm/proddata/WebCommerce/samplestores`
-  `/opt/WebSphere/CommerceServer/samplestores`
-  `/opt/WebSphere/CommerceServer/samplestores`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplestores`
-  `unidad:\WebSphere\CommerceServer\samplestores`

Notas:







- a. La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.
 - b. Para ver los archivos `businesspolicy.xml` y `contract.xml` del archivador de tienda, descomprímalos utilizando un programa ZIP. Los archivos están situados en el directorio `data`.
 - c. Los archivos de elementos de contrato de la tienda de ejemplo ToolTech que se proporciona con WebSphere Commerce Business Edition incluyen información para contratos que no son el contrato por omisión de la tienda.
4. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
 5. Cree un archivo `businesspolicy.xml`, copiando uno de los archivos `businesspolicy.xml` de los archivadores de tienda de ejemplo o bien creando un archivo nuevo. Encontrará las instrucciones sobre cómo crear un archivo nuevo en el apartado “Creación de archivos XML de políticas de negocio” en la página 105. Si desea crear diferentes políticas de negocio a partir de las descritas, consulte el archivo DTD correspondiente a `businesspolicy.xml`. Los archivos DTD se encuentran en el directorio siguiente:

-  `/usr/WebSphere/CommerceServer/xml/sar`
-  `/qibm/proddata/WebCommerce/xml/sar`
-  `/opt/WebSphere/CommercServer/xml/sar`
-  `/opt/WebSphere/CommercServer/xml/sar`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar`
-  `unidad:\WebSphere\CommerceServer\xml\sar`

6. Cargue el archivo `businesspolicy.xml` utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225. Si crea una tienda multicultural, deberá crear archivos

XML para cada entorno nacional al que la tienda dé soporte. El archivo específico de entorno nacional debe especificar toda la información de las descripciones, para que se pueda traducir fácilmente.

7. Cree un archivo `contract.xml`, copiando uno de los archivos `contract.xml` de los archivadores de tienda de ejemplo o bien creando un archivo nuevo. Encontrará las instrucciones sobre cómo crear un archivo nuevo en el apartado “Creación de un archivo XML de contrato por omisión” en la página 106. Si desea crear un contrato por omisión más complejo, revise el archivo `B2BTrading.dtd` que define la estructura de un archivo `contract.xml`. El archivo `B2BTrading.dtd` está en el directorio siguiente:

-  `/usr/WebSphere/CommerceServer/xml/trading`
-  `/qibm/proddata/WebCommerce/xml/trading`
-  `/opt/WebSphere/CommerceServer/xml/trading`
-  `/opt/WebSphere/CommerceServer/xml/trading`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\trading`
-  `unidad:\WebSphere\CommerceServer\xml\trading`

8. Publique el contrato utilizando el mandato `ContractImportApprovedVersion`. Para obtener más información, consulte el Capítulo 29, “Publicación de cuentas de negocio y contratos” en la página 293. En la ayuda en línea también hay información disponible acerca del mandato `ContractImportApprovedVersion`.

Los usuarios de WebSphere Commerce Business Edition pueden definir contratos para clientes específicos utilizando WebSphere Commerce Accelerator. Para obtener más información acerca de cómo crear contratos para clientes específicos, consulte la información en línea.

Creación de archivos XML de políticas de negocio

Aunque WebSphere Commerce proporciona varias políticas de negocio a las que pueden hacer referencia los términos y condiciones del contrato por omisión de la tienda, deberá definir, de todos modos, algunas políticas de negocio. Debe definir los costes de las devoluciones, las aprobaciones de las devoluciones y las políticas de negocio de cálculo de precio a los que harán referencia los términos del contrato por omisión de la tienda. Se proporcionan los mandatos para estas políticas de negocio y se pueden utilizar sin tener que modificarlos. Si desea crear sus propias políticas de negocio, consulte la publicación *IBM WebSphere Commerce, Guía del programador*.

Para crear políticas de negocio para la tienda, debe crear una política de negocio y asociar uno o más mandatos a dicha política de negocio. Para crear una política de negocio, añada información a la tabla `POLICY`. Para asociar un mandato a una política de negocio, añada información a la tabla `POLICYCMD`.

Para crear una política de negocio y asociar mandatos a ella, haga lo siguiente:

1. Cree una política de negocio en el archivo XML de políticas de negocio, añadiendo información a la tabla `POLICY`. Utilice el ejemplo siguiente como muestra:

```
<policy
policy_id="@policy_id_10"
policyname="MasterCatalogPriceList"
```

```

policytype_id="Price"
storeent_id="@storeent_id_1"
properties="name=InFashion&member_id=&MEMBER_ID"
/>

```

donde

- policy_id es el identificador numérico y exclusivo de la política de negocio.
 - policyname es un nombre exclusivo de la política de negocio.
 - policytype_id es el tipo de política que se está definiendo. Los valores de policytype_id válidos son:
 - InvoiceFormat
 - Payment
 - Price
 - ProductSet
 - ReturnApproval
 - ReturnCharge
 - ReturnPayment
 - ShippingCharge
 - ShippingPayment
 - storeent_id es la tienda o grupo de tiendas.
 - properties es una lista de parejas de nombre/valor que se envía al mandato de política de negocio.
2. Asocie un mandato a la política de negocio en el archivo XML de políticas de negocio, añadiendo información a la tabla POLICYCMD. Utilice el ejemplo siguiente como muestra:

```

<polycmd
policy_id="@policy_id_10"
businesscmdclass=
"com.ibm.com.commerce.price.commands.CalculateContractPricesCmdImpl"
/>

```

donde

- policy_id es el identificador numérico de la política de negocio con el que se ha asociado el mandato.
- businesscmdclass es el nombre de la clase Java que implementa la política de negocio.

Las particiones de línea en el atributo businesscmdclass sólo se muestran a efectos de presentación.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Creación de un archivo XML de contrato por omisión

Para crear un contrato por omisión, debe definir el contrato, el propietario del contrato, la descripción del contrato, los participantes del contrato y los términos y condiciones del contrato. La información sobre un contrato se almacena en cuatro tablas: CONTRACT, PARTICIPNT, TRADING y TERMCOND.

El contrato por omisión se asocia a la tienda mediante la tabla de base de datos STOREDEF. Para los usuarios de WebSphere Commerce Business Edition, los contratos distintos al contrato por omisión se asocian a una tienda mediante la tabla de base de datos STORECNTR.

Para crear un contrato por omisión, haga lo siguiente:

1. Defina el contrato por omisión en el archivo XML. El contrato por omisión se define al principio del archivo XML, tal como se muestra a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Trading SYSTEM "B2BTrading.dtd">
<Trading>
<Contract state="Active" origin="Manual"
  name="&STORE_IDENTIFIER; Default Contract" majorVersionNumber="1"
  minorVersionNumber="0" contractUsage="Default">
```

Las particiones de línea en el elemento Contract sólo aparecen a efectos de presentación.

2. Defina el propietario del contrato. Utilice el ejemplo siguiente como muestra:

```
<ContractOwner>
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</ContractOwner>
```

donde distinguishName es el nombre del usuario propietario del contrato con el formato de nombre distinguido LDAP. Por ejemplo, uid=erickoeck,ou=People,dc=ibm,dc=com.

3. Defina la descripción del contrato en el archivo XML de contratos. Utilice el ejemplo siguiente como muestra:

```
<ContractDescription title="Contrato por omisión de la tienda." languageId="-1">
</ContractDescription>
```

donde

- title es una descripción de texto del contrato.
- languageId es el idioma del título. Para languageId se han definido previamente los valores siguientes:
 - -1 (Inglés de EE.UU.)
 - -2 (Francés)
 - -3 (Alemán)
 - -4 (Italiano)
 - -5 (Español)
 - -6 (Portugués de Brasil)
 - -7 (Chino simplificado)
 - -8 (Chino tradicional)
 - -9 (Coreano)
 - -10 (Japonés)

Se pueden definir valores adicionales para languageId actualizando los elementos de idioma de la tienda. Para obtener más información sobre los elementos de tienda, consulte el Capítulo 14, “Elementos de idioma” en la página 123.

4. Defina los participantes del contrato en el archivo XML de contratos. Utilice el ejemplo siguiente como muestra:

```

<Participant role="Buyer">
</Participant>
<Participant role="Seller">
  <Member>
    <Organization distinguishName="&MEMBER_IDENTIFIER;" />
  </Member>
</Participant>

```

donde distinguishName es el nombre del vendedor del contrato con el formato de nombre distinguido LDAP. Por ejemplo, uid=erickoeck,ou=People,dc=ibm,dc=com. En muchos casos, será igual que el propietario del contrato.

Nota: No se especifican miembros en el rol de participante de comprador porque el contrato está disponible para todos los usuarios con el rol de comprador.

- Defina los términos y condiciones en el archivo XML de contratos. Los elementos y atributos XML son distintos para los diversos tipos de términos y condiciones. Utilice el archivo B2BTrading.dtd para saber qué elementos y atributos XML utilizar para cada tipo de término. Al definir términos y condiciones, comúnmente se utilizan los atributos siguientes:

policyName

El nombre de la política de negocio al que hacen referencia el término y condición. El nombre se almacena en POLICY.POLICYNAME.

policyType

El tipo de política de negocio al que hacen referencia el término y condición. Los valores válidos son:

- Price
- ProductSet
- InvoiceFormat
- Payment
- ReturnApproval
- ReturnCharge
- ReturnPayment
- ShippingCharge
- ShippingMode

storeIdentity

La tienda o grupo de tiendas para el término y condición.

distinguishName

El nombre del usuario que es el propietario de la tienda o grupo de tiendas. El nombre debe tener el formato de nombre distinguido LDAP. Por ejemplo, uid=wcsadmin,o=Root Organization.

Los siguientes términos y condiciones de ejemplo van precedidos de una descripción de lo que definen:

- Todos los compradores pueden adquirir todos los artículos del catálogo maestro de la tienda, a los precios establecidos en dicho catálogo maestro:

```

<TermCondition>
  <PriceTC>
    <PriceTCMasterCatalogWithOptionalAdjustment>
  </PriceTCMasterCatalogWithOptionalAdjustment>
  </PriceTC>
</TermCondition>

```

- Los compradores pagan al vendedor los costes de envío:

```
<TermCondition>
  <ShippingTC>
    <ShippingTCShippingCharge>
      <PolicyReference policyName="StandardShippingChargeBySeller"
        policyType="ShippingCharge" storeIdentity="&STORE_IDENTIFIER;">
        <Member>
          <User distinguishName="&MEMBER_IDENTIFIER;">
        </Member>
      </PolicyReference>
    </ShippingTCShippingCharge>
  </ShippingTC>
</TermCondition>
```

Las particiones de línea en el elemento PolicyReference sólo aparecen a efectos de presentación.

- Los compradores pueden devolver los productos sin sufrir ningún recargo de devolución. Los productos deben devolverse dentro del número de días definido en la política de negocio ApprovalByDays:

```
<TermCondition>
  <ReturnTC>
    <ReturnTCReturnCharge>
      <ReturnChargePolicyReference>
        <PolicyReference policyName="NoCharges"
          policyType="ReturnCharge"
          storeIdentity="&STORE_IDENTIFIER;">
          <Member>
            <Organization distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ReturnChargePolicyReference>
      <ReturnApprovalPolicyReference>
        <PolicyReference policyName="ApprovalByDays"
          policyType="ReturnApproval"
          storeIdentity="&STORE_IDENTIFIER;">
          <Member>
            <Organization distinguishName="&MEMBER_IDENTIFIER;">
          </Member>
        </PolicyReference>
      </ReturnApprovalPolicyReference>
    </ReturnTCReturnCharge>
  </ReturnTC>
</TermCondition>
```

Las particiones de línea en los elementos PolicyReference sólo aparecen a efectos de presentación.

Nota para usuarios de WebSphere Commerce Business Edition:

Omitir estos términos y condiciones en el contrato por omisión de la tienda indica que, por omisión, la tienda no acepta devoluciones. Sin embargo, otros contratos pueden permitir que los compradores hagan devoluciones, si se definen los términos y condiciones de las devoluciones.

Nota para usuarios de WebSphere Commerce Professional Edition:

Omitir estos términos y condiciones en el contrato por omisión de la tienda indica que la tienda no acepta devoluciones.

- Los reembolsos se pagan utilizando el mismo método de pago que ha utilizado el comprador para formalizar el pedido:

```
<TermCondition>
  <ReturnTC>
    <ReturnTCRefundPaymentMethod>
```

```
<PolicyReference policyName="UseOriginalPayment"
  policyType="ReturnPayment" storeIdentity="&STORE_IDENTIFIER;">
  <Member>
    <User distinguishName="&MEMBER_IDENTIFIER;">
    </Member>
  </PolicyReference>
</ReturnTCTRefundPaymentMethod>
</ReturnTC>
</TermCondition>
```

Las particiones de línea en el elemento PolicyReference sólo aparecen a efectos de presentación.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Capítulo 11. Elementos de despacho de pedidos

Las tiendas utilizan los centros de despacho de pedidos como almacenes de inventario y como centros de envío y recepción. Una tienda puede tener uno o muchos centros de despacho de pedidos asociados a ella. Un centro de despacho gestiona el inventario de productos y el envío de productos de una tienda. El despacho de pedidos incluye la selección, el empaquetado y el envío. La selección de productos puede efectuarse para una o más salidas de un centro de despacho de pedidos, el empaquetado consiste en poner estos productos en contenedores de envío y el envío es la acción de enviarlos a los clientes.

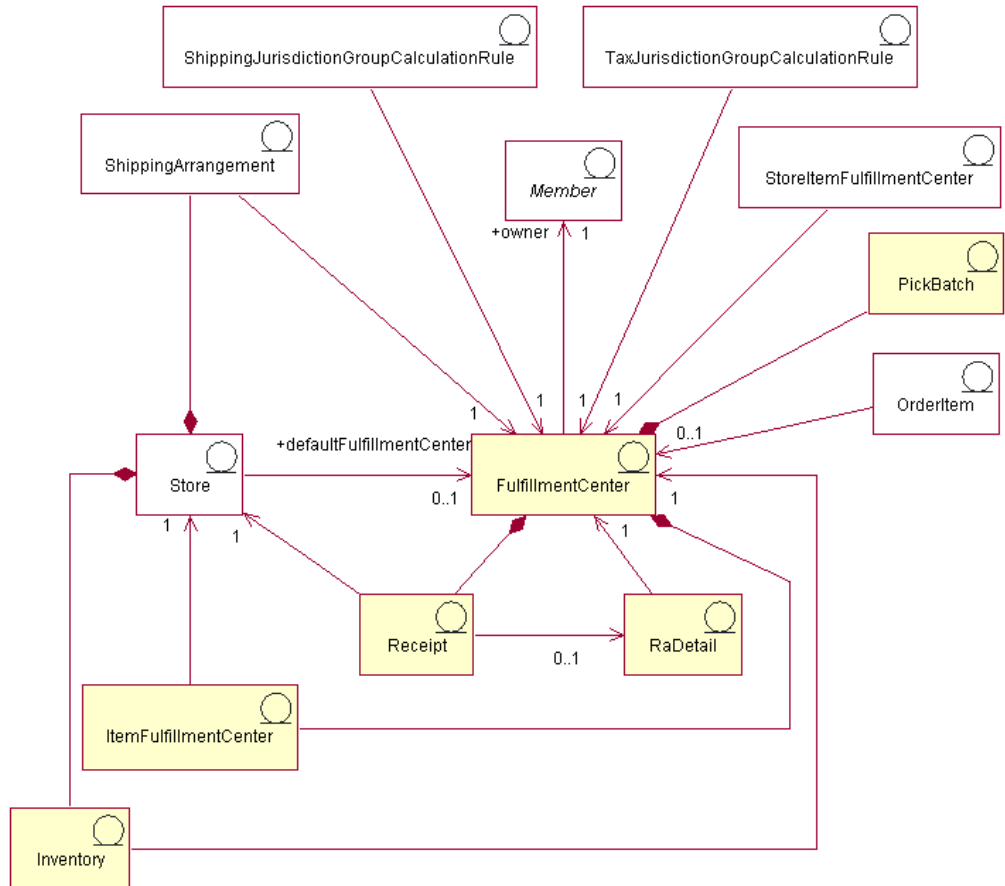
Los productos se configuran para el despacho de pedidos con el asistente Producto y el cuaderno Producto. La configuración de producto proporciona opciones para hacer un seguimiento del inventario, permitir pedidos en espera, forzar pedidos en espera, despachar el producto por separado y especificar que el producto no se debe devolver.

Normalmente hay varias personas trabajando a la vez en un centro de despacho de pedidos, cada una con una tarea o tareas diferentes a realizar. WebSphere Commerce Accelerator divide las tareas más comunes en roles y estos roles se asignan a usuarios. En WebSphere Commerce Accelerator, deberá seleccionar un centro de despacho de pedidos en el momento de realizar la conexión, si se le han asignado uno o más roles pertenecientes al despacho de pedidos.

Nota: Para obtener más información sobre el despacho de pedidos, los centros de despacho de pedidos y los roles, consulte la ayuda en línea de WebSphere Commerce.

Los elementos de despacho de pedidos en WebSphere Commerce

Para comprender los elementos de despacho de pedido, es necesario comprender las relaciones entre el despacho de pedidos y la tienda. Esto puede explicarse mediante el uso de un modelo de información. Las secciones siguientes describen las relaciones que tiene el inventario con una tienda y otros elementos.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Centro de despacho de pedidos

En el diagrama anterior, el *centro de despacho de pedidos* es el centro del proceso de despacho de pedidos. Un centro de despacho de pedidos tiene un propietario, definido en la tabla MEMBER. Cada tienda está asociada a un centro de despacho de pedidos y tiene un centro de despacho de pedidos por omisión. Un centro de despacho de pedidos puede tener varias tiendas asociadas a él. Tal como se indica en el diagrama, existen varias interacciones entre la tienda y el centro de despacho de pedidos. Para obtener más información sobre los elementos de tienda, consulte el apartado “Los elementos de tienda en WebSphere Commerce” en la página 43.

Recepciones

Los centros de despacho de pedidos reciben los artículos cada día, semana o mes. Cuando se recibe un artículo, se crea una *recepción* en la tabla RECEIPT que registra la información sobre la cantidad recibida, así como la tienda que posee el inventario. A medida que se procesan los pedidos, la tabla RECEIPT se actualiza para reflejar los niveles actuales de inventario disponible. Para obtener más información sobre cómo crear recibos, consulte el apartado “Creación de elementos de inventario en WebSphere Commerce” en la página 183.

RaDetail

RaDetail es la información detallada acerca de los artículos de un registro de inventario esperado. Esta información puede utilizarse para calcular cuándo se puede esperar recibir el inventario en un centro de despacho de pedidos y para proporcionar a los clientes las fechas de envío esperadas para los artículos de los pedidos en espera.

Inventario

Una tienda tiene *inventario* que está asociado con el centro de despacho de pedidos. El inventario incluye todo aquello que puede contabilizarse en un centro de despacho de pedidos. El inventario está asociado a una tienda y a un centro de despacho de pedidos. También se registra la información sobre el inventario que posee una tienda en el centro de despacho de pedidos, como por ejemplo, cantidades reservadas, cantidades en pedidos en espera y cantidades asignadas a los pedidos en espera. Esta información se almacena en la tabla ITEMFFMCTR. Para obtener más información sobre el inventario y los elementos de inventario, consulte el Capítulo 21, “Elementos de inventario” en la página 179.

Acuerdos de envío

La última relación entre la tienda y el centro de despacho de pedidos requiere *acuerdos de envío*. Los acuerdos de envío indican que un centro de despacho de pedidos puede enviar productos en nombre de una tienda utilizando una modalidad de envío. Cada tienda tiene un acuerdo de envío con un centro de despacho de pedidos y viceversa. Los acuerdos de envío se definen en la tabla SHPARRANGE. Para obtener más información sobre cómo crear acuerdos de envío, consulte el apartado “Creación de elementos de despacho de envío” en la página 149.

Otros elementos relacionados con el despacho de pedidos

Hay otras relaciones con un centro de despacho de pedidos que no están directamente relacionadas con una tienda. Un lote de requisición de artículos es un elemento que está asociado a un centro de despacho de pedidos. Un lote de requisición de artículos agrupa las salidas de los pedidos para su proceso como una unidad en un centro de despacho de pedidos y crea listas de requisición y listas de embalaje. Una vez los artículos se han recogido y se han empaquetado, puede efectuarse el envío de la salida del pedido, y el envío puede confirmarse. La información sobre lotes de requisición se almacenan en la tabla PICKBATCH. Un artículo de pedido también está asociado a un centro de despacho de pedidos. Un artículo es una instancia específica de un producto, definida por atributos. La información sobre cada artículo de un pedido se almacena en la tabla ORDERITEM. Para obtener más información sobre los elementos de los pedidos, consulte el Capítulo 22, “Elementos de pedido” en la página 185.

Al igual que otras entidades, un centro de despacho de pedidos tiene normas que rigen algunas de sus acciones. Cada centro de despacho de pedidos tiene normas para los impuestos y los costes de envío. Estas normas están definidas en las tablas TAXJCRULE y SHPJCRULE, respectivamente. Para obtener más información sobre los elementos de impuestos y de envío, consulte el Capítulo 18, “Elementos de envío” en la página 137 y el apartado “Los elementos de impuestos en WebSphere Commerce” en la página 155.



Para obtener información más detallada sobre la estructura de los elementos de despacho de pedidos en WebSphere Commerce Server, consulte los modelos de datos de despacho de pedidos en la ayuda en línea de WebSphere Commerce.





Creación de elementos de despacho de pedidos en WebSphere Commerce

Antes de que la tienda pueda enviar artículos a un cliente, debe definir el centro, o centros, de despacho de pedidos que suministrarán estos artículos. Esta información debe crearse en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225.

Para crear elementos de despacho de pedidos para su tienda utilizando un archivo XML, haga lo siguiente:

1. Revise los archivos XML utilizados para crear elementos de despacho de pedidos para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda se encuentran en el directorio siguiente:






-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo fulfillment.xml, que incluye la información de despacho de pedidos. Para ver el archivo fulfillment.xml del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo fulfillment.xml se encuentra en el directorio de datos.

2. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
3. Cree un archivo fulfillment.xml, copiando uno de los archivos fulfillment.xml de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a fulfillment.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar

-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

4. Defina el centro, o los centros, de despacho de pedidos que la tienda soporta:
 - a. Utilizando como guía el ejemplo siguiente, defina un centro de despacho de pedidos en el archivo XML, en la tabla FFMCENTER:

```
<ffmcenter
  ffmcenter_id="@ffmcenter_id_1"
  member_id="MEMBER_ID"
  name="ToolTech Home"
  defaultshipoffset="0"
  markfordelete="0"
/>
```

donde

- ffmcenter_id es una clave exclusiva generada
 - member_id es el propietario del centro de despacho de pedidos
 - name es una serie que, junto con el propietario, identifica de forma exclusiva el centro de despacho de pedidos.
 - defaultshipoffset es una estimación del número de segundos que tarda en enviarse un artículo desde este centro de despacho de pedidos. Este valor se puede modificar en la tabla STORITMFFC.
 - markfordelete indica si el centro de despacho de pedidos debe suprimirse: 0 = no suprimir. 1 = suprimir si ya no se usa. Para obtener más detalles, consulte la información sobre el programa de utilidad de Limpieza de base de datos en la ayuda en línea de WebSphere Commerce.
- b. Utilizando como guía el ejemplo siguiente, describa el centro de despacho de pedidos en el archivo XML, en la tabla FFMCENTDS. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional.

```
<ffmcentds
  ffmcenter_id="@ffmcenter_id_1"
  description="El centro de despacho de pedidos que proporciona
              productos a ToolTech."
  language_id="@es_ES"
  displayname="Centro de despacho de pedidos de ToolTech"
  staddress_id="@staddress_id_es_ES_1"
/>
```

donde

- ffmcenter_id es una clave exclusiva generada
- description es una descripción del centro de despacho de pedidos adecuada para mostrarla a un cliente.
- language_id es el idioma en que se visualiza esta información.
- displayname es una descripción del centro de despacho de pedidos adecuada para mostrarla a un cliente.
- staddress_id_loc es la ubicación física del centro de despacho de pedidos.

- c. Repita los pasos a y b para todos los centros de despacho de pedidos que la tienda soporte.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, “Creación de los datos” en la página 327.

Creación de elementos de despacho de pedidos

Después de haber definido el centro o los centros de despacho de pedidos que proporcionarán artículos para la tienda, deberá asociar un centro de despacho de pedidos a cada producto. Es decir, deberá identificar qué centro de despacho de pedidos suministrará cada uno de los productos. Para crear esta relación, añada información a la tabla INVENTORY. Esta información debe crearse en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225.







Nota:

1. Para poder asociar una tienda con un centro de despacho de pedidos, deberá crear elementos de tienda. Para obtener más información sobre la creación de elementos de tienda, consulte “Creación de elementos de datos de tienda en un archivo XML” en la página 45. También debe crear los elementos de catálogo antes de crear los elementos de despacho de pedidos de la tienda. Para obtener más información, consulte “Visualización de los elementos de catálogo de la tienda” en la página 77.
2. Cree elementos de despacho de pedidos de tienda sólo si implementa el despacho de pedidos no ATP. La tabla INVENTORY no la utiliza una tienda que incluye las funciones ATP.

Para crear la relación de despacho de pedidos de la tienda utilizando un archivo XML, haga lo siguiente:

1. Revise los archivos XML que se utilizan para crear elementos de despacho de pedidos de la tienda para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda se encuentran en el directorio siguiente:







-  unidad:\WebSphere\CommerceServer\samplstores
-  unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo `storefulfill.xml`, que incluye la información de despacho de pedidos de la tienda. Para ver el archivo

storefulfill.xml del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo storefulfill.xml se encuentra en el directorio de datos.

2. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
3. Cree un archivo storefulfill.xml, copiando uno de los archivos storefulfill.xml de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a storefulfill.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommercServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Utilizando como guía el ejemplo siguiente, cree una relación de tienda y centro de despacho de pedidos en el archivo XML, añadiendo información en la tabla INVENTORY.

```
<inventory
catentry_id="@catentry_id_1470"
quantity="100"
ffmcenter_id="@ffmcenter_id_1"
store_id="@storeent_id_1"
quantitymeasure="C62"
inventoryflags="0"
/>
```

donde

- catentry_id es la entrada de catálogo que suministrará este centro de despacho de pedidos.
 - quantity es la cantidad, en las unidades indicadas por QUANTITYMEASURE, disponible en este centro de despacho de pedidos.
 - ffmcenter_id es el centro de despacho de pedidos que suministrará el inventario.
 - store_id es la tienda para la que se suministra el inventario.
 - quantitymeasure es la unidad de medida para QUANTITY.
 - inventoryflags son los distintivos de bit que indican cómo se utiliza QUANTITY:
 - 1 = noUpdate. El mandato de tarea por omisión UpdateInventory no actualiza QUANTITY.
 - 2 = noCheck. Los mandatos de tarea por omisión CheckInventory y UpdateInventory no comprueban QUANTITY.
5. Repita el paso 3 para cada entrada de catálogo de su tienda.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Capítulo 12. Elementos de campaña

Las campañas sirven para organizar las actividades de marketing. Las campañas las crea normalmente un Director de marketing o un Director de comercialización. Generalmente están asociadas con un conjunto determinado de objetivos. Por ejemplo, una campaña de "Vuelta al cole" puede tener el objetivo de incrementar las ventas de ropa infantil durante la campaña.

Las campañas en WebSphere Commerce

En WebSphere Commerce, las campañas contienen cualquier número de iniciativas de campaña, que definen una condición. Las iniciativas de campaña generan contenido destinado a los clientes, cuando la condición definida se evalúa como verdadera. El resultado es que una campaña es el elemento de marketing de alto nivel que organiza las iniciativas.

Las iniciativas de campaña se asocian con una campaña que contenga un conjunto de iniciativas. Como ejemplo de esta relación, si una tienda de artículos de oficina tenía una campaña de "Vuelta al cole", las iniciativas serán responsables de las acciones de nivel inferior, por ejemplo anunciar un descuento en los lápices o sugerir papel rayado a cualquier cliente que se haya registrado y haya indicado que es un estudiante.

Las iniciativas de campaña son capaces de visualizar tres tipos de contenido dinámico:

- Iniciativa de ventas por sugerencia
- Recomendación basada en un filtro colaborador
- Anuncios publicitarios informativos

El contenido de venta por sugerencia está diseñado para proporcionar recomendaciones de productos y categorías según unas normas; estas sugerencias van destinadas a un público de clientes específico, basándose en el perfil del cliente y el comportamiento de otros clientes. Las iniciativas que visualizan este tipo de contenido están destinadas a utilizarse para crear oportunidades de venta cruzada y de venta ascendente.

Las recomendaciones basadas en el filtro colaborador también están diseñadas para crear recomendaciones de productos, pero utilizan un algoritmo de recomendación diferente, que presenta artículos basándose en el comportamiento general de los clientes, en lugar de hacerlo en normas predefinidas.

Los anuncios publicitarios informativos están diseñados para dirigir el contenido de un anuncio a un grupo de clientes específico, basándose en los mismos criterios que los utilizados para la venta por sugerencia, pero su fin es que el cliente tenga más información sobre las actividades de la tienda en línea, resaltar las ofertas especiales y aumentar el conocimiento de los tipos de tarjeta.

Las iniciativas pueden incorporarse en cualquier página del sitio. Cuando se diseña el sitio, se colocan espacios reservados especiales, llamados Zonas de e-marketing, en el sitio. Cuando se presentan a un cliente, estos espacios reservados se sustituyen por el contenido específico deseado. Las ubicaciones de destino se asignan planificando la visualización de iniciativas en zonas de e-marketing de las

ubicaciones deseadas. Para obtener más información sobre cómo añadir zonas de e-marketing a su tienda, consulte el Capítulo 32, “Adición de zonas de e-marketing a la tienda” en la página 317.

Las iniciativas de campaña contienen una condición que determina cuándo y a quién se muestran. Esta condición se define cuando se crea la iniciativa y se puede cambiar durante el tiempo de vida de la iniciativa para ajustar la visibilidad de la iniciativa y el contenido visualizado.

Las iniciativas de campaña generan estadísticas sobre su uso. Estas estadísticas pueden verlas los comerciantes, los directores de marketing y los directores de comercialización utilizando WebSphere Commerce Accelerator. Las estadísticas ilustran el índice de pulsaciones de una iniciativa para cada zona de e-marketing donde se implementa. Estas estadísticas proporcionan información sobre la eficacia de la iniciativa, así como porcentajes comparativos de éxito entre las diversas ubicaciones en las que se visualiza.

Creación de elementos de campaña en WebSphere Commerce

Las campañas y las iniciativas de campaña las crea normalmente un Director de marketing o un Director de comercialización, mediante los asistentes de campaña o de iniciativa de campaña de WebSphere Commerce Accelerator. Para obtener más información, consulte la ayuda en línea de WebSphere Commerce.

Para obtener más información sobre cómo añadir zonas de e-marketing a su tienda, consulte el Capítulo 32, “Adición de zonas de e-marketing a la tienda” en la página 317.

Capítulo 13. Elementos de pago

WebSphere Commerce da soporte a IBM Payment Manager. Para crear elementos de pago para la tienda, especifique si la tienda utiliza Payment Manager y, si es así, el tipo de casete de pago y los tipos de tarjeta que acepta la tienda.

Especifique esta información realizando lo siguiente:

- Cree datos de pago con el formato de un archivo XML (`paymentinfo.xml`) que se carga durante la publicación de la tienda mediante los Servicios de tienda. Esto configurará Payment Manager con el comerciante y los tipos de tarjetas especificados para la tienda que se está publicando. Para obtener más información, consulte el apartado "Creación de elementos de pago utilizando un archivo XML".

Nota: `paymentinfo.xml` no llena de datos las tablas de la base de datos de WebSphere Commerce Server, sino que configura Payment Manager. `paymentinfo.xml` sólo es aplicable si se están utilizando como método de pago tarjetas de crédito fuera de línea. Para configurar otros métodos de pago, consulte la información que viene a continuación.







- Complete la configuración de Payment Manager para la tienda mediante la Consola de administración o la interfaz de usuario de Payment Manager. Si utiliza la Consola de administración, aparecerán los elementos de menú en el menú de Payment Manager. Si utiliza la interfaz de usuario de Payment Manager, los elementos de menú aparecerán bajo Administración en el marco de navegación. Para obtener más información, consulte el tema "*Configurar Payment Manager para la tienda*" en la ayuda en línea de WebSphere Commerce.

Creación de elementos de pago utilizando un archivo XML

Para crear elementos de pago para la tienda utilizando un archivo XML, realice lo siguiente:

1. Revise los archivos XML utilizados para crear elementos de pago para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:







-  NT unidad:\WebSphere\CommerceServer\samplestores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplestores
-  AIX /usr/WebSphere/CommerceServer/samplestores
-  Solaris /opt/WebSphere/CommerceServer/samplestores
-  Linux /opt/WebSphere/CommerceServer/samplestores
-  400 /qibm/proddata/WebCommerce/samplestores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo `paymentinfo.xml`, que contiene la información de pago. Para ver el archivo `paymentinfo.xml` del archivador de

tienda, descomprímalo utilizando un programa ZIP. Los archivos paymentinfo.xml se encuentran en el directorio de datos.

2. Cree un archivo paymentinfo.xml, copiando uno de los archivos paymentinfo.xml de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a paymentinfo.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

3. Habilite o inhabilite Payment Manager.

- a. Utilizando como guía el ejemplo siguiente, habilite o inhabilite Payment Manager en el archivo XML y especifique los tipos de casete de pago, monedas y tipos de tarjeta que acepta la tienda.

```
<paymentinfo>
  <PaymentManager enable="yes"/>
  <Cassette type="OfflineCard">
    <Account currency="USD">
      <Brand type="MasterCard"/>
      <Brand type="VISA"/>
      <Brand type="American Express"/>
    </Account>
    <Account currency="EUR">
      <Brand type="MasterCard"/>
      <Brand type="VISA"/>
      <Brand type="American Express"/>
    </Account>
  </Cassette>
</paymentinfo>
```

donde:

- enable indica si Payment Manager está habilitado o inhabilitado.
- Cassette type es el tipo de casete soportado.
- Account currency son las monedas a las que da soporte la tienda. Este valor es necesario si está utilizando el tipo de casete OfflineCard. La moneda se debe identificar mediante un código de tres letras que se ajuste al estándar ISO 4217. Por ejemplo, "USD" para dólares de EE.UU.
- Brand type es el tipo de tarjeta de crédito soportado por Account currency.

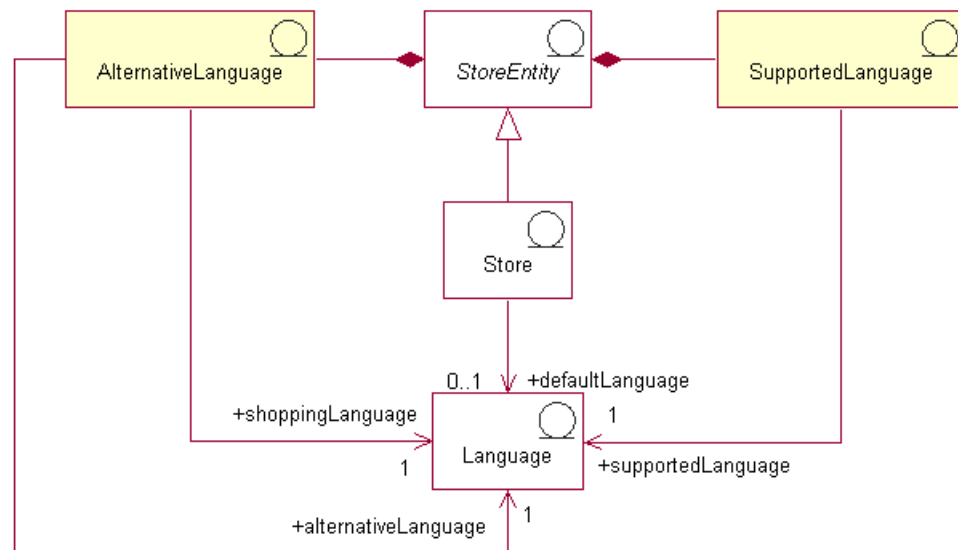
Capítulo 14. Elementos de idioma

En WebSphere Commerce, un sitio puede definir muchos idiomas que se pueden utilizar en el mismo. La tabla LANGUAGE define los diez idiomas soportados, que incluyen alemán, chino tradicional y chino simplificado, japonés, coreano, italiano, francés, español, portugués de Brasil e inglés. Los sitios pueden definir idiomas adicionales o dialectos de las lenguas existentes para personalizar el modo en que se presenta la información a los clientes de culturas diferentes o cuyos datos estadísticos sean diferentes.

Los elementos de idioma en WebSphere Commerce

Para comprender los elementos de idioma, es necesario conocer las relaciones entre los idiomas y la tienda. Esto puede explicarse mediante el uso de un modelo de información, que se muestra a continuación. La sección siguiente describe las relaciones y asociaciones que tiene el idioma con una tienda y otros elementos.

El diagrama siguiente muestra el modelo de información de los elementos de idioma.



En WebSphere Commerce, los idiomas se clasifican en cuatro tipos. Los idiomas por omisión, los idiomas soportados, los idiomas alternativos y los idiomas de compra. Cada uno de estos tipos de idioma lleva a cabo un rol distinto en la tienda. Todos los idiomas se almacenan en la tabla LANGUAGE.

Idioma por omisión

Cada tienda tiene asociado un *idioma por omisión*. Este es el idioma que la tienda ha elegido para utilizar como idioma principal y será el idioma que se mostrará en primer lugar a los clientes que seleccionan de forma explícita un idioma de compras. El idioma por omisión de una tienda está soportado de forma implícita por la tienda, es decir, la tienda debe poder visualizar la información en el idioma por omisión o en uno de los idiomas alternativos, si los hay definidos en la tabla

LANGPAIR. Cuando la información no está disponible en uno de los idiomas soportados o idiomas alternativos, la información se mostrará en el idioma por omisión.

Idiomas soportados

La tabla STORELANG indica los idiomas que soporta cada tienda. Una tienda debe poder mostrar información en sus *idiomas soportados* o en sus idiomas alternativos, si están definidos en la tabla LANGPAIR. Una tienda también debe soportar todos los idiomas soportados por su grupo de tiendas.

Idioma alternativo

Cuando no hay información disponible en uno de los idiomas soportados, la tienda intenta mostrar la información en un *idioma alternativo*, si está disponible. Una tienda puede especificar el orden en que se han de intentar sus idiomas alternativos. Los idiomas alternativos de una tienda incluyen los idiomas alternativos de su grupo de tiendas. Los idiomas alternativos pueden ser útiles cuando hay información que solamente está disponible en un idioma, pero debería estar disponible para los clientes que compran en un idioma diferente pero afín. Esto puede ser el caso cuando, por ejemplo, todavía no se ha traducido toda la información en todos los idiomas soportados o cuando, por ejemplo, se da soporte a dos dialectos muy similares de la misma lengua, a veces con la misma información.



Para obtener información más detallada sobre la estructura de los elementos de idioma en WebSphere Commerce Server, consulte los modelos de datos y de objeto de idioma en la ayuda en línea de WebSphere Commerce.

Creación de elementos de idioma en WebSphere Commerce

Puede definir los idiomas a los que da soporte su tienda de una de las dos maneras siguientes:

- Utilizando las herramientas de los Servicios de tienda
- En un archivo XML que cargará Loader package o la herramienta de publicación de los Servicios de tienda

Nota: Las herramientas de los Servicios de tienda trabajan con archivos XML con datos preinsertados, en forma de archivador de tienda.

Para obtener más información sobre la definición de idiomas soportados por la tienda utilizando los Servicios de tienda, consulte la ayuda en línea de WebSphere Commerce. Para obtener más información sobre la definición de los idiomas soportados por la tienda en un archivo XML, consulte “Creación de elementos de datos de tienda en un archivo XML” en la página 45.

Capítulo 15. Elementos de moneda

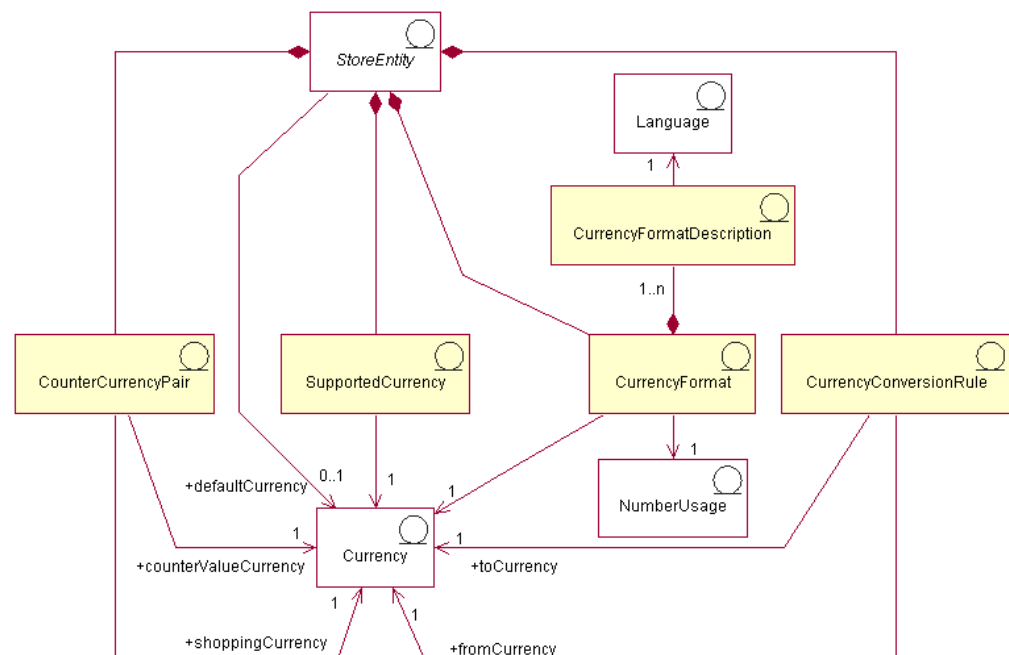
Puede mostrar los precios en su sitio Web en una sola moneda o puede utilizar varias monedas siguiendo las instrucciones que se proporcionan para el euro. Para un sitio con múltiples tiendas, puede utilizar monedas diferentes para las tiendas o puede asignar monedas al grupo de tiendas. En función de la naturaleza del sitio que esté creando, puede especificar qué monedas desea utilizar y cómo se visualizarán.

En WebSphere Commerce, puede permitir que los clientes seleccionen una moneda de compra. La moneda de compra es la moneda con la que los clientes pagan los productos en una tienda específica. Todos los importes monetarios de las páginas de la tienda se visualizan en esta moneda. Cuando los clientes cambian la moneda de compra, los precios de los artículos que han añadido a los carros de la compra y los totales de pedido se convierten, recalculan y visualizan automáticamente en la nueva moneda de compra.

Los clientes pueden comprar en muchas monedas, incluido el euro. El euro se convirtió en moneda legal para la Unión Europea el 1 de enero de 1999 y se utiliza actualmente en los mercados financieros. Las tasas de conversión entre el euro y las monedas de todos los países participantes son fijas. Los billetes y las monedas para la moneda única de la Unión Europea están disponibles desde el 1 de enero de 2002. Seis meses después, las monedas nacionales existentes serán retiradas de la circulación de forma definitiva. Durante el periodo de transición entre 1999 y 2001, los comerciantes aceptaban tanto las monedas nacionales como el euro.

Los elementos de moneda en WebSphere Commerce

El diagrama siguiente muestra la estructura de los elementos de moneda en WebSphere Commerce Server:





Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

En el diagrama anterior, la moneda está en el centro del modelo de información. Cada tienda, o grupo de tiendas, tiene una moneda por omisión.

Formato de moneda

Una entidad de tienda puede tener muchas *normas de formato de moneda*. Si una tienda no tiene una norma de formato para una moneda específica, utiliza la norma de formato de su grupo de tiendas. Los formatos de tienda se establecen en la tabla CURFORMAT.

Utilización de números

Cada norma de formato de moneda está asociada con una *utilización de números*. Los números como, por ejemplo, los importes monetarios y las cantidades se pueden formatear y redondear de forma diferente según la utilización asociada. Las tiendas pueden especificar distintas normas de redondeo y formateo de los números que se visualizan, según la forma en que se utilicen; por ejemplo, una tienda puede redondear los precios por unidad a cuatro cifras decimales especificando la utilización de precio unitario, y redondear otros importes a dos cifras decimales especificando la utilización por omisión. La utilización de números se almacena en la tabla NUMBRUSG.

Descripción del formato de moneda

Una norma de formato de moneda puede tener muchas *descripciones de formato de moneda*. Una descripción de formato de moneda describe cómo se da formato (para su visualización) a un importe monetario en una moneda e idioma determinados. Cada descripción está asociada a un idioma en la tabla LANGUAGE. Para obtener más información sobre los elementos de idioma consulte el Capítulo 14, “Elementos de idioma” en la página 123. Las descripciones sobre los formatos de moneda se almacenan en la tabla CURFMTDESC.

Moneda soportada

Una entidad de tienda puede tener muchas *monedas soportadas*. Una moneda soportada es aquella en que se aceptan los pagos.

Norma de conversión de moneda

Todas las monedas tienen normas que rigen la conversión a y de otras monedas. Cada *norma de conversión de moneda* puede utilizarse para convertir un precio (almacenado en la base de datos en una moneda específica) al importe que se le cobrará a los clientes en una moneda de compra soportada.

Moneda de contravalor

Las *monedas de contravalor* son importes monetarios que se muestran junto con una moneda soportada. No se pueden utilizar para comprar sino que se utilizan para fines informativos. Si los clientes deciden comprar en euros, podrán visualizar los importes monetarios en euros y en otra moneda en la tienda. Los importes en la

moneda de compra se convierten a todas las monedas de contravalor para esa moneda de compra. Las monedas de contravalor están emparejadas con una moneda soportada, como por ejemplo la peseta española y el euro. Las parejas de moneda de contravalor están almacenadas en la tabla CURCVLIST.



Para obtener información más detallada sobre la estructura de los elementos de moneda en WebSphere Commerce Server, consulte el modelo de datos de moneda en la ayuda en línea de WebSphere Commerce.

Creación de elementos de moneda en WebSphere Commerce

Las herramientas de los Servicios de tienda de WebSphere Commerce le permiten añadir monedas soportadas a la tienda y seleccionar una moneda por omisión para la tienda. Para obtener más información sobre qué elementos puede editar con las herramientas de Servicios de tienda, consulte el tema "*Cambiar los elementos de base de datos de la tienda*" de la ayuda en línea de WebSphere Commerce.

Nota: Las herramientas de los Servicios de tienda trabajan con archivos XML con datos preinsertados, en forma de archivador de tienda.

También puede añadir a la tienda las monedas soportadas y una moneda por omisión mediante un archivo XML que se puede cargar en la base de datos utilizando Loader package. Este método también permite crear otros tipos de elementos de moneda, incluida la definición de las tasas de conversión de monedas y las monedas de contravalor.

Para obtener información sobre cómo editar los elementos de moneda de un archivador de tienda existente, consulte la ayuda en línea de WebSphere Commerce. Para obtener información sobre cómo crear elementos de moneda nuevos con formato de archivo, consulte "Creación de elementos de moneda utilizando un archivo XML".






Creación de elementos de moneda utilizando un archivo XML

Cree los elementos de moneda de su tienda en formato de archivo XML que se pueda cargar en la base de datos mediante Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, "Publicación de la tienda" en la página 225.

Para crear elementos de moneda para su tienda utilizando un archivo XML, haga lo siguiente:

1. Revise los archivos XML que se utilizan para crear elementos de moneda para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda se encuentran en el directorio siguiente:







-  unidad:\WebSphere\CommerceServer\samplstores
-  unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores

-  /qibm/proddata/WebCommerce/samplestores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo `currency.xml`, que contiene la información de moneda. Para ver los archivos `currency.xml` en el archivador de tienda, descomprímalo utilizando un programa ZIP. Los archivos `currency.xml` se encuentran en el directorio de datos.

2. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
3. Cree un archivo `currency.xml`, copiando uno de los archivos `currency.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponde a `currency.xml`. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommercServer/xml/sar
-  Linux /opt/WebSphere/CommercServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Defina las monedas que soporta la tienda.
 - a. Utilizando el ejemplo siguiente como muestra, defina las monedas soportadas por la tienda en el archivo XML para la tabla CURLIST:


```
<curlist currstr="EUR" storeent_id="@storeent_id_1" />
```

donde:

- `currstr` es el código de moneda ISO 4217 de tres caracteres que representa la moneda soportada. Este código debe aparecer en la columna SETCCURR de la tabla SETCURRE. Una tienda debe poder aceptar el pago en todas las monedas soportadas.
- `storeent_id` es la entidad de tienda.

- b. Repita este paso para cada moneda soportada por la tienda.



La moneda por omisión para una tienda se define en la tabla STOREENT. Para obtener más información, consulte “Creación de elementos de datos de tienda en un archivo XML” en la página 45.

5. (Opcional) Los precios y la moneda que aparecen en la tienda dependen del modo en que defina los precios. Puede definir los precios para cada moneda que se utilice en la tienda o puede definirlos solamente para la moneda por omisión. Para obtener más información sobre cómo definir los precios, consulte el apartado “Creación de elementos de cálculo de precio en WebSphere Commerce” en la página 94.

Si cuando define los precios solamente define los precios para la moneda por omisión y, sin embargo, desea mostrar los precios de la tienda en otras monedas soportadas, deberá añadir las tasas de conversión de la tienda. Utilice esta tasa de conversión para realizar la conversión desde la moneda por omisión a la moneda soportada.

- a. Determine la moneda desde la que realizará la conversión, por ejemplo, el dólar americano (USD) y la moneda o monedas a las que va a convertir los precios, por ejemplo, el Yen (JPY). Para determinar los códigos de moneda ISO para cada moneda, consulte los códigos ISO 4217 para las monedas internacionales.
- b. Utilice el ejemplo siguiente como guía y añada la información de conversión a la tabla CURCONVERT:

```
<curconvert
storeent_id="@storeent_id_1"
fromcurr="USD"
tocurr="JPY"
factor="105.10"
multiplyordivide="M"
bidirectional="Y"
updatable="Y"
curconvert_id="@curconvert_id_1" />
```

donde:

- storeent_id es la entidad de tienda.
- fromcurr es la moneda desde la que va a realizar la conversión. Un importe en la moneda FROMCURRE normalmente forma parte de una norma o de otra información que se utiliza para determinar un precio, un descuento, un coste de envío o un importe similar asociado con un producto que se ha puesto en venta.
- tocurr es la moneda a la que va a realizar la conversión. Normalmente, TOCURRE es la moneda en la que el cliente piensa pagar. Normalmente, los importes en esta moneda forman parte de un artículo de pedido como, por ejemplo, un precio unitario, un coste de envío o un importe de impuestos.
- factor es el factor de conversión.
- multiplyordivide es como se indica a continuación: para realizar la conversión de FROMCURRE a TOCURRE:
 - M = Multiplicar por FACTOR
 - D = Dividir por FACTOR

Para las normas bidireccionales, se permite la conversión de TOCURRE a FROMCURRE utilizando la operación inversa.

- bidirectional indica que la norma es bidireccional o unidireccional:
 - Y = bidireccional
 - N = unidireccional
 - updatable es un distintivo que utiliza una interfaz de usuario que gestiona las normas de conversión de moneda. Valores válidos:
 - N = la tasa de conversión es irrevocable y no debe cambiarse nunca
 - Y = la tasa de conversión se puede cambiar
 - curconvert_id es una clave exclusiva generada.
- c. Repita los pasos a y b para todas las monedas en que desee mostrar los precios.



Incluso si ha definido precios para todas las monedas soportadas en su información de cálculo de precios, es posible que desee definir las tasas de conversión de moneda para las monedas que soporta la tienda.

6. (Opcional) si desea mostrar precios en la moneda de compra y en la moneda de contravalor (por ejemplo, para mostrar precios en pesetas y en euros), debe añadir la información a la tabla CURCVLIST.

a. Utilice el ejemplo siguiente como guía y añada la información de conversión a la tabla CURCVLIST:

```
<curcvlist  
storeent_id="@storeent_id_1"  
currstr="NLG"  
countervaluecurr="EUR"  
displayseq="1" />
```

donde:

- storeent_id es la entidad de tienda.
- currstr es el código de moneda ISO 4217 de tres caracteres que representa la moneda. Este código debe aparecer en la columna SETCCURR de la tabla SETCURRE y es la moneda a la que realizará la conversión. Un importe en la moneda FROMCURRE normalmente forma parte de una norma o de otra información que se utiliza para determinar un precio, un descuento, un coste de envío o un importe similar asociado con un producto que se ha puesto en venta.
- countervaluecurr es el código de moneda ISO 4217 de tres caracteres que representa la moneda de contravalor. Este código debe aparecer en la columna SETCCURR de la tabla SETCURRE.
- displayseq es el número que indica el orden de presentación de la moneda de contravalor. Las monedas de contravalor se muestran en orden ascendente según la secuencia de visualización del contravalor que se ha especificado en la columna DISPLAYSEQ de la tabla CURCVLIST.



Para obtener más información sobre cómo utilizar @ y & consulte el Apéndice B, "Creación de los datos" en la página 327.

Otras tareas de moneda

Para obtener más información sobre la moneda en general y otras tareas de moneda, entre ellas:

- Añadir nuevas monedas que WebSphere Commerce no soporta actualmente
- Cambiar los formatos de las monedas existentes
- Añadir nuevas normas de formato para monedas nuevas o para una tienda específica

consulte la ayuda en línea de WebSphere Commerce.

Capítulo 16. Elementos de unidad de medida

Los productos pueden venderse, y se puede hacer un seguimiento del inventario, utilizando diversas unidades de medida, como por ejemplo, kilos, pulgadas, litros, etc. Usando estas unidades, se pueden solicitar productos en cantidades mínimas y por múltiplos de cantidades específicas.

Los mandatos de controlador utilizan el parámetro UOM (unidad de medida) para especificar la unidad de medida. Si no se especifica un parámetro UOM, la cantidad especificada del cliente se multiplica por la cantidad nominal de la entrada de catálogo de la tabla de base de datos CATENTSHIP. El resultado se conoce como la cantidad solicitada.

La cantidad solicitada se redondea por exceso al siguiente múltiplo de cantidad más alto para la entrada de catálogo. Por ejemplo, si el múltiplo es 2 kilos y la cantidad solicitada es 4,1 kilos, el resultado del redondeo será 6 kilos. La cantidad redondeada se utiliza al comprobar el inventario, que tiene su propia unidad de medida. Si la unidad de medida del inventario y la unidad de medida de entrada de catálogo son diferentes, debe efectuarse una conversión entre las dos unidades.

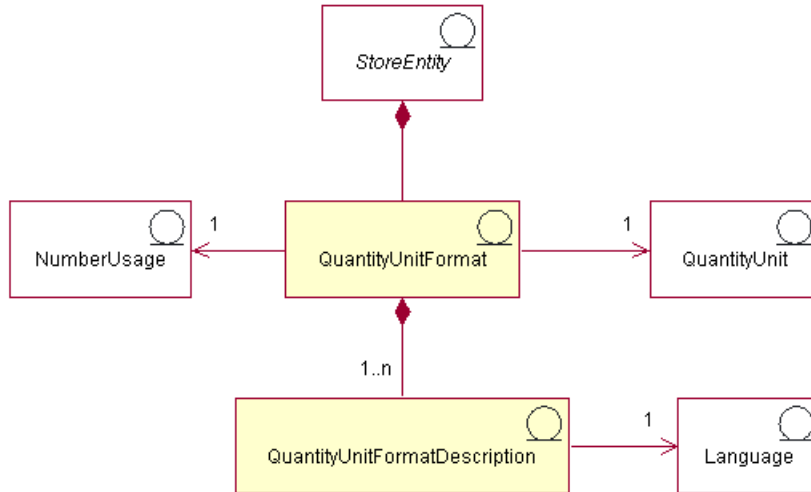
Cuando se habilita el inventario de disponibilidad asegurada (ATP) (consulte la columna ALLOCATIONGOODFOR de la tabla STORE), la unidad de medida de inventario se define en la columna QUANTITYMEASURE de la tabla BASEITEM. De lo contrario, se define en la columna QUANTITYMEASURE de la tabla INVENTORY.

La cantidad redondeada dividida por la cantidad nominal de la entrada de catálogo se conoce como cantidad normalizada. La cantidad normalizada se almacena en el artículo de pedido o el artículo de interés, en función del mandato que se esté ejecutando. Por ejemplo, si la cantidad redondeada es 6 kilos y la cantidad nominal es 2 kilos, la cantidad normalizada es 3.

Cuando se busca una oferta de una entrada de catálogo, la cantidad solicitada puede tener importancia para seleccionar la oferta que proporciona el mejor precio y, por consiguiente, determinar qué oferta se utilizará. Por ejemplo, si la cantidad redondeada es 6 kg y hay dos ofertas, una que especifica un precio de 4,00 euros para la cantidad nominal de 2 kilos y una cantidad mínima de 10 kilos y otra que especifica un precio de 4,50 euros para la cantidad nominal de 2 kilos y una cantidad mínima de 2 kilos, sólo se podrá utilizar la segunda oferta.

Las unidades de medida en WebSphere Commerce

El diagrama siguiente ilustra la estructura de las unidades de medida en WebSphere Commerce Server:



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Unidad de cantidad y formato de la unidad de cantidad

Una *unidad de cantidad* es la unidad de medida que se utiliza en la tienda, por ejemplo, kilogramos, libras, metros, pulgadas, litros, etc. El formato de la unidad de cantidad indica cómo se formatea esta unidad de cantidad, por ejemplo, cuántas cifras decimales se utilizan al mostrar la unidad de cantidad.

Cada *formato de unidad de cantidad* forma parte de una sola entidad de tienda, pero cada entidad de tienda puede tener varios formatos de unidad de cantidad.

Cada formato de unidad de cantidad tiene una una unidad de cantidad y utilización de números, pero puede tener una o más descripciones de formato de unidad de cantidad, dependiendo del número de idiomas que la tienda soporte.

Descripción del formato de unidad de cantidad

La *descripción del formato de unidad de cantidad* describe cómo dar formato (a efectos de visualización) a una cantidad en una unidad de cantidad y un idioma específicos.

Utilización de números

La *utilización de números* define el modo en que se utiliza un número en una aplicación. Por ejemplo, si utiliza códigos de utilización de números en el código de WebSphere Commerce, podrá elegir el modo en que desea que se formatee o redondee dicho número (moneda o cantidad). Estos códigos (definidos en la tabla NUMBRUSG) permiten formatear el número de acuerdo con las normas especificadas para dicho tipo de utilización de números en las tablas CURFORMAT, CURFMTDESC, QTYFORMAT y QTYFMTDESC. Esto permite a las tiendas formatear los números de varias formas, a fin de satisfacer los requisitos de diversas situaciones.



Para obtener información más detallada sobre la estructura de los elementos de unidad de medida en WebSphere Commerce Server, consulte el modelo de datos de la unidad de cantidad en la ayuda en línea de WebSphere Commerce.

Creación de unidades de medida en WebSphere Commerce

Cuando se crea una instancia, las unidades de medida están ya insertadas en la base de datos de WebSphere Commerce Server. Para obtener más información, consulte el Capítulo 5, “Elementos de sitio” en la página 39.

También puede definir nuevas unidades de medida en WebSphere Commerce para utilizarlas en la tienda o suprimir unidades de medida que no desee utilizar en la tienda.

Para definir nuevas unidades de medida para utilizarlas en la tienda, añada información a las siguientes tablas de base de datos:

- QTYUNIT
- QTUNITDSC
- QTYFORMAT
- QTYFMTDESC
- QTYUNITMAP
- QTYCONVERT

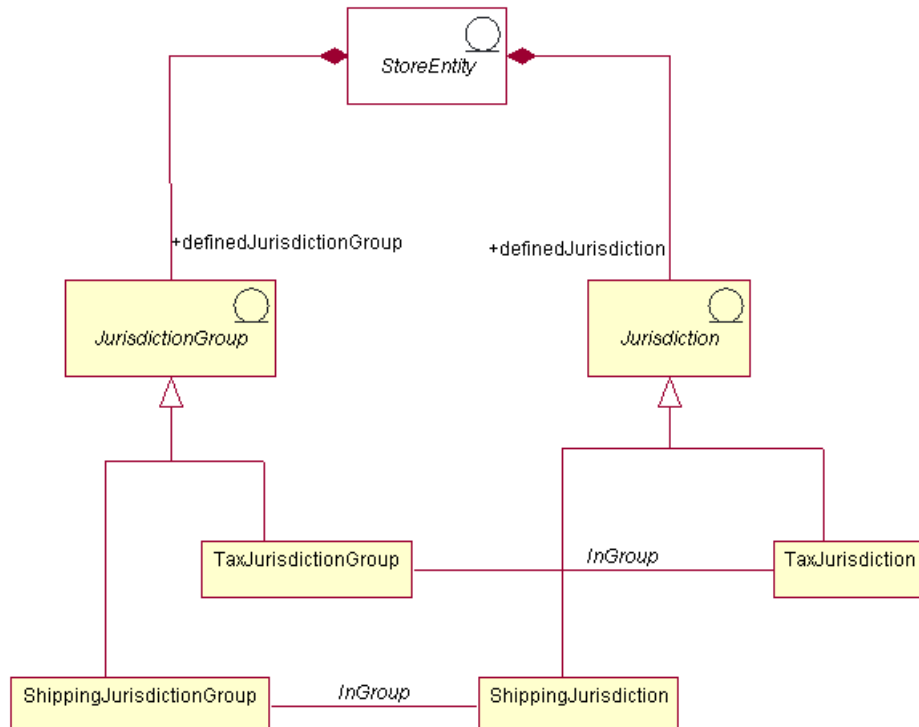
Capítulo 17. Elementos de jurisdicción

Las *jurisdicciones* son regiones o zonas geográficas que representan un país o una región, una provincia o un territorio, o un rango de códigos postales, en los que se venden mercancías. Las jurisdicciones se pueden agrupar para formar *grupos de jurisdicciones*.

Los grupos de jurisdicciones se utilizan en el cálculo de los costes de envío y los impuestos de los pedidos. Es decir, un grupo de jurisdicciones se puede utilizar para calificar las normas de cálculo de costes de envío e impuestos que se utilizan. Estas normas de cálculo calificadas se aplican a los artículos de un pedido solamente si el artículo se va a enviar a una dirección dentro de una de las jurisdicciones de un grupo de jurisdicciones que está asociado con la norma de cálculo. Como resultado, los costes de envío y los impuestos se pueden calcular de forma diferente según las direcciones de envío de los diferentes artículos del pedido.

Los elementos de jurisdicción en WebSphere Commerce

El siguiente diagrama ilustra la relación entre las jurisdicciones y los grupos de jurisdicciones en WebSphere Commerce Server.





Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

En WebSphere Commerce una jurisdicción o un grupo de jurisdicciones es una parte de la tienda y es exclusiva de la tienda para la que se ha creado. Por ejemplo, si crea tres jurisdicciones para una tienda y, posteriormente, suprime la tienda, también se suprimen las jurisdicciones. No las puede utilizar ninguna tienda existente ni ninguna tienda que cree en el futuro.

Sin embargo, si crea jurisdicciones para un grupo de tiendas, las jurisdicciones no se suprimirán cuando se supriman las tiendas de dicho grupo. Las jurisdicciones estarán disponibles para las nuevas tiendas creadas en dicho grupo de tiendas.

WebSphere Commerce da soporte a dos tipos de jurisdicciones: jurisdicciones de envío y jurisdicciones fiscales. Las jurisdicciones de envío se pueden agrupar para formar grupos de jurisdicciones que califican las normas de cálculo de costes de envío. Del mismo modo, las jurisdicciones fiscales se pueden agrupar para formar grupos de jurisdicciones fiscales que calificarán las normas de cálculo de impuestos.



Para obtener información más detallada sobre la estructura de los elementos de jurisdicción en WebSphere Commerce Server, consulte el modelo de datos de jurisdicción en la ayuda en línea de WebSphere Commerce.

Creación de elementos de jurisdicción en WebSphere Commerce

Debe crear elementos de jurisdicción para la tienda para poder aplicar los impuestos y los costes de envío. Para obtener más información sobre la creación de jurisdicciones, consulte “Creación de elementos de impuestos en WebSphere Commerce” en la página 158 o “Creación de elementos de envío en WebSphere Commerce” en la página 139.

Una vez haya creado jurisdicciones para la tienda, puede editarlas o crear de nuevas, utilizando los cuadernos Impuestos y Envíos de los Servicios de tienda.

Nota: Los Servicios de tienda crean automáticamente un grupo de jurisdicciones para cada jurisdicción que crean. Los Servicios de tienda crean jurisdicciones para tiendas pero no para grupos de tiendas.

Nota: Los Servicios de tienda solamente trabajan con los datos no publicados con formato de archivador de tienda.

Capítulo 18. Elementos de envío

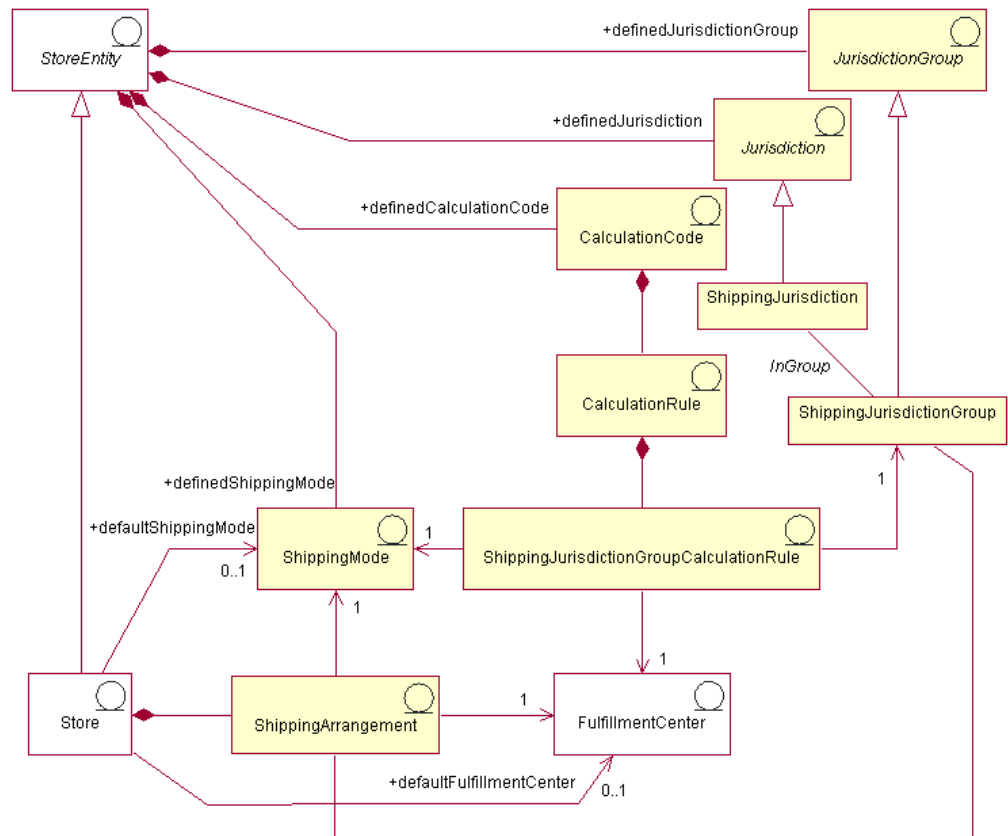
El envío es el modo en que una tienda lleva a cabo físicamente la entrega de mercancías a los clientes. En la mayoría de los casos, las mercancías se envían desde un centro de despacho de pedidos, una agencia independiente que es responsable de almacenar las mercancías de la tienda.

Para ofrecer servicios de envío, y cobrar por dichos servicios, una tienda que se haya creado con WebSphere Commerce deberá incluir lo siguiente:

- una modalidad de envío como mínimo
- un código de cálculo de costes de envío como mínimo
- jurisdicciones y grupos de jurisdicciones

Los elementos de envío en WebSphere Commerce

El diagrama siguiente ilustra la estructura de envío en WebSphere Commerce Server.





Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Modalidades de envío

La *modalidad de envío* es una forma de enviar las mercancías. Más específicamente, una modalidad de envío es la combinación de una empresa de transportes (que es una empresa que proporciona servicios de envío desde un centro de despacho de pedidos a un cliente) y el servicio de envío ofrecido por dicha empresa de transportes. Por ejemplo, Compañía de transportes ABC, Servicio 12 horas y Compañía de transportes ABC, Entrega urgente son modalidades de envío.

Una modalidad de envío pertenece a una entidad de tienda. Si se suprime la entidad de tienda, también se suprimirán las modalidades de envío definidas en dicha entidad de tienda. No es necesario que una tienda tenga una modalidad de envío por omisión pero es recomendable.

Acuerdos de envío

Un *acuerdo de envío* es un acuerdo entre la tienda y el centro de despacho de pedidos, que indica que un centro de despacho de pedidos enviará mercancías para una tienda determinada utilizando las modalidades de envío especificadas. Se pueden imponer determinadas restricciones en un acuerdo de envío, incluyendo el periodo de tiempo durante el cual el acuerdo de envío está en vigor y las jurisdicciones de envío.

Si un acuerdo de envío está asociado con una modalidad de envío, solamente se aplica a dicha modalidad de envío. De lo contrario, el acuerdo de envío se aplica a todas las modalidades de envío disponibles. Un acuerdo de envío forma parte de una tienda y se suprimirá si se suprime la tienda.

Códigos de cálculo

Los *códigos de cálculo* se utilizan para calcular los costes de envío, es decir, un código de cálculo de costes de envío indica cómo se calculan los costes de envío para los artículos de pedido. Para calcular los costes de envío de un artículo de pedido, debe asignar códigos de cálculo de costes de envío a una entrada de catálogo o a un grupo de entradas de catálogo.

Un código de cálculo forma parte de una entidad de tienda. Un código de cálculo sólo se puede asociar a una entidad de tienda, pero una entidad de tienda puede tener varios códigos de cálculo. Si se suprime la entidad de tienda, también se suprimirán los códigos de cálculo asociados a dicha entidad de tienda.



Para obtener más información sobre el uso de códigos de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

Normas de cálculo

Cada código de cálculo tiene un conjunto de *normas de cálculo*. Los costes de envío de un artículo de pedido pueden variar en función de la modalidad de envío, del centro de despacho de pedidos y de las jurisdicciones de envío.

ShippingJurisdictionGroupCalculationRules son los objetos de relación que asocian las normas de cálculo de envío con las jurisdicciones, los centros de despacho de pedidos y las modalidades de envío para determinar qué normas de cálculo deben utilizarse para cada artículo de pedido.

Si se suprime la norma de cálculo o cualquiera de los objetos a los que hace referencia ShippingJurisdictionGroupCalculationRules, también se suprimirá ShippingJurisdictionGroupCalculation.



Para obtener más información sobre el uso de normas de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

Jurisdicciones y grupos de jurisdicciones

Las *jurisdicciones* son regiones o zonas geográficas que representan un país o una región, una provincia o un territorio, o un rango de códigos postales, en los que se venden mercancías. Las jurisdicciones se agrupan para formar *grupos de jurisdicciones*.

WebSphere Commerce soporta dos tipos de jurisdicciones: jurisdicciones de envío y jurisdicciones fiscales. Cada una de estas jurisdicciones forma parte del grupo correspondiente, por ejemplo, las jurisdicciones de envío están en el grupo de jurisdicciones de envío y las jurisdicciones fiscales están en el grupo de jurisdicciones fiscales.

Los grupos de jurisdicciones se asocian a normas de cálculo. La norma de cálculo utiliza el grupo de jurisdicciones como parte del cálculo para determinar el importe de los costes de envío.

Las jurisdicciones y los grupos de jurisdicciones forman parte de una entidad de tienda. Si se suprime una entidad de tienda, también se suprimirán las jurisdicciones y los grupos de jurisdicciones asociados a dicha entidad de tienda.

Una dirección de envío se puede resolver en varias jurisdicciones de envío. Por ejemplo, una dirección de envío en Nueva York, Estados Unidos, se aplicará a las siguientes jurisdicciones de envío: "Nueva York, Estados Unidos", "Estados Unidos" y el "Mundo". Cuando una dirección de envío se aplica a varias jurisdicciones de envío, se podrán aplicar varias normas de cálculo de envío. En estos casos, se utiliza la prioridad de la ShippingJurisdictionGroupCalculationRules asociada para determinar qué norma o normas se utilizarán.



Para obtener información más detallada sobre la estructura de los elementos de envío en WebSphere Commerce Server, consulte los modelos de datos y de objeto de impuestos en la ayuda en línea de WebSphere Commerce.

Creación de elementos de envío en WebSphere Commerce

Las herramientas de Servicios de tienda de WebSphere Commerce le permiten crear y editar en un archivador de tienda determinados elementos de envío (por ejemplo modalidades de envío y jurisdicciones), pero no todos los elementos de envío. Para obtener más información sobre qué elementos puede editar con las herramientas de Servicios de tienda, consulte el tema "*Cambiar los elementos de base de datos de la tienda*" de la ayuda en línea de WebSphere Commerce.

Nota: Las herramientas de los Servicios de tienda trabajan con archivos XML con datos preinsertados, en forma de archivador de tienda.

También puede crear los elementos de envío en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package. De este modo, tiene las dos opciones siguientes para crear elementos de envío:

- Edite los elementos de envío existentes de una de las tiendas de ejemplo que se incluyen con WebSphere Commerce, o de un archivador de tienda existente.
- Cree nuevos elementos de envío con formato de archivo XML

Para obtener información sobre cómo editar los elementos de envío de un archivador de tienda existente, consulte la ayuda en línea de WebSphere Commerce. Para obtener información sobre cómo crear elementos de envío nuevos en forma de archivo XML, consulte “Creación de elementos de envío utilizando un archivo XML”.







Creación de elementos de envío utilizando un archivo XML

Cree los elementos de envío en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225. Si crea una tienda multicultural, deberá crear archivos XML para cada entorno nacional al que la tienda dé soporte. El archivo específico de entorno nacional debe contener toda la información de las descripciones, para que se pueda traducir fácilmente.

Las tiendas de ejemplo, de las que se han tomado muchos de los ejemplos de estas tareas, utilizan un archivo `shipping.xml` para toda la información que no necesita traducirse y otro archivo `shipping.xml` para cada entorno nacional soportado por la tienda para la información que necesita traducirse. Los archivos específicos de entorno nacional contienen toda la información de descripción, para que ésta pueda traducirse fácilmente.







Para crear elementos de envío para la tienda utilizando un archivo XML, realice lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, envío) asociados al producto o servicio que un cliente ha seleccionado para adquirir.
2. Revise los archivos XML utilizados para crear elementos de envío para las tiendas de ejemplo. Todos los archivos de las tiendas de ejemplo se encuentran en el archivador de tienda correspondiente. Cada tienda de ejemplo incluye dos o más archivos `shipping.xml`, que incluyen la información de envío. Los archivos de archivador de tienda están ubicados en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Nota: La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Para ver los archivos shipping.xml del archivador de tienda, descompríalos utilizando un programa ZIP. Los archivos shipping.xml se encuentran en el directorio de datos. El archivo shipping.xml específico de un idioma está situado en el subdirectorío específico del entorno nacional del directorio de datos.

3. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
4. Cree un archivo shipping.xml, copiando uno de los archivos shipping.xml de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a shipping.xml. Los archivos DTD se encuentran en el directorio siguiente:
 -  NT unidad:\WebSphere\CommerceServer\xml\sar
 -  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
 -  AIX /usr/WebSphere/CommerceServer/xml/sar
 -  Solaris /opt/WebSphere/CommercServer/xml/sar
 -  Linux /opt/WebSphere/CommercServer/xml/sar
 -  400 /qibm/proddata/WebCommerce/xml/sar
5. Defina las jurisdicciones y el grupo de jurisdicciones a las que está enviando mercancías o servicios. Todas las jurisdicciones deben pertenecer a un grupo de jurisdicciones.

- a. Utilizando el ejemplo siguiente como guía, defina un grupo de jurisdicciones en el archivo XML de la tabla JURSTGROUP:

```
<jurstgroup
jurstgroup_id="@jurstgroup_id_1"
description="Grupo de jurisdicciones1 para envío"
subclass="1"
storeent_id="@storeent_id_1"
code="Mundial"/>
```

donde

- jurstgroup_id es una clave exclusiva generada
 - description es una breve descripción del grupo de jurisdicciones, adecuada para visualizarse en una interfaz de usuario para la gestión de grupos de jurisdicciones.
 - subclass es la subclase del grupo de jurisdicciones tal como se indica a continuación:
 - 1 = ShippingJurisdictionGroup
 - 2 = TaxJurisdictionGroup
 - storeent_id es la entidad de tienda asociada a este grupo de jurisdicciones.
 - code que, junto con la entidad de tienda y la subclase, identifica de forma exclusiva este grupo de jurisdicciones.
- b. Utilizando el ejemplo siguiente como guía, defina una jurisdicción en el archivo XML de la tabla JURST.

```
< jurst
jurst_id="@jurst_id_1"
```

```
storeent_id="@storeent_id_1"
code="Mundial"
subclass="1"/>
```

donde

- jurst_id es una clave exclusiva generada
- storeent_id es la entidad de tienda asociada a este grupo de jurisdicciones.
- code que, junto con la entidad de tienda y la subclase, identifica de forma exclusiva este grupo de jurisdicciones.
- subclass es la subclase de jurisdicción tal como se indica a continuación:
 - 1 = ShippingJurisdiction
 - 2 = TaxJurisdiction

- c. Utilizando el ejemplo siguiente como guía, asocie la jurisdicción que ha creado en el paso b al grupo de jurisdicciones que ha definido en el paso a, añadiendo información a la tabla JURSTGRPTEL.

```
<jurstgprel
jurst_id="@jurst_id_1"
jurstgroup_id="@jurstgroup_id_1"
subclass="1"/>
```

donde

- jurst_id es la jurisdicción
- jurstgroup_id es el grupo de jurisdicciones
- subclass es la subclase de la jurisdicción y del grupo de jurisdicciones. Estas deben coincidir:
 - 1 = ShippingJurisdiction[Group]
 - 2 = TaxJurisdiction[Group]

- d. Repita los pasos a a c para todas las jurisdicciones y todos los grupos de jurisdicciones que la tienda soporte.

6. Defina las modalidades de envío que utilizará la tienda.

- a. Utilizando el ejemplo siguiente como guía, defina una modalidad de envío en el archivo XML para la tabla SHIPMODE:

```
<shipmode
shipmode_id="@shipmode_id_1"
field1
storeent_id="@storeent_id_1"
code="1 semana por correo normal"
carrier="Empresa de transportes XYZ"/
```

donde:

- shipmode_id es una clave exclusiva generada.
- field1 es un campo que se puede personalizar.
- storeent_id es la entidad de tienda asociada con esta modalidad de envío.
- code es el código asignado por el comerciante, exclusivo para la entidad de tienda.
- carrier es el nombre o identificador de la empresa de transportes.

- b. Utilizando el ejemplo siguiente como guía, añade información acerca de la modalidad de envío a la tabla SHPMODEDSC. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional:

```
< shpmodedsc
description="Correo internacional"
field1="5 euros por pedido más 1 euro por cada artículo"
field2="5 días laborables"
shipmode_id="@shipmode_id_1"
language_id="&es_ES;" />
```

donde:

- description es una breve descripción de la Modalidad de envío, adecuada para mostrarla a un cliente para que la seleccione.
- field1 y field2 son campos disponibles que se pueden personalizar.
- shipmode_id es una clave exclusiva generada.
- language_id es el idioma utilizado.

- c. Repita los pasos a y b para todas las modalidades de envío de la tienda.

7. Defina los códigos de cálculo que deberá utilizar la tienda.

- a. Utilizando los ejemplos siguientes como guía, defina el código de cálculo en el archivo XML para la tabla CALCODE.

```
< calcode
calcode_id="@calcode_id_1"
code="Código de envío 1- por/pedido"
calusage_id="-2"
storeent_id="@storeent_id_1"
groupby="0"
published="1"
sequence="+0.00E+000"
calmethod_id="-23"
calmethod_id_app="-24"
calmethod_id_qfy="-22"
flags="0" />
```

donde:

- calcode_id es una clave exclusiva generada.
- code es una serie de caracteres que identifica de forma exclusiva este Código de cálculo, dado un uso de cálculo y una entidad de tienda determinados.
- calusage_id indica la clase de cálculo para la que se utiliza este código de cálculo. Por ejemplo, el código de cálculo puede utilizarse para calcular uno de los siguientes importes monetarios:
 - descuentos (-1)
 - costes de envío (-2)
 - impuestos sobre la venta (-3)
 - impuestos sobre el envío (-4)
 - cupones (-5)
- storeent_id es la entidad de tienda asociada a este código de cálculo.

- `groupby` son distintivos de bit que indican al método `CalculationCodeCombineMethod` cómo deben agruparse los artículos de pedido al realizar cálculos. 0 = Sin agrupación. Coloque todos los artículos de pedido aplicables en un solo grupo. Consulte *Tabla CALCODE: detalles* en la ayuda en línea de WebSphere Commerce para obtener más información.
 - `published` especifica si el código de cálculo se publica o no:
 - 0 = No publicado (inhabilitado temporalmente)
 - 1 = Publicado
 - 2 = Marcado para suprimirse (y no publicado)
 - `sequence`: Los códigos de cálculo se calculan y se aplican en secuencia, del más bajo al más alto. Si dos códigos de cálculo tienen el mismo número de secuencia, se calcularán primero los códigos de cálculo cuyo `calcode_id` sea menor.
 - `calmethod_id` es el método `CalculationCodeCalculateMethod` que define cómo calcular un importe monetario para este código de cálculo. `calmethod_id="-23"`, el método `CalculationCodeCalculateMethod` para envío, es el único método de cálculo de envío proporcionado con WebSphere Commerce.
 - `calmethod_id_app` es el método `CalculationCodeApplyMethod` que almacena el importe calculado para los artículos de pedido asociados. `calmethod_id_app="-24"`, el método `CalculationCodeApplyMethod` para envío, es el único método de aplicación de envío proporcionado con WebSphere Commerce.
 - `calmethod_id_qfy` es el método `CalculationCodeQualifyMethod` que define qué artículos de pedido se asocian a este código de cálculo. `calmethod_id_qfy="-22"`, el método `CalculationCodeQualifyMethod` para envío, es el único método de calificación de envío proporcionado con WebSphere Commerce.
 - `flags` especifica si se debe invocar el método `CalculationCodeQualifyMethod` de este código de cálculo.
 - 0 = no restringido. El método no se invocará
 - 1 = restringido. El método se invocará.
- b. Utilizando el ejemplo siguiente como guía, añada la información de descripción de código de cálculo al archivo XML para la tabla `CALCODEDSC`. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional.
- ```
<calcodedsc
 calcode_id="@calcode_id_3"
 description="5 euros por pedido"
 language_id("&es_ES"
 longdescription= "Este código de cálculo de envío cobra 5 euros por envío."
/>
```
- donde
- `calcode_id` es el código de cálculo al que se aplica esta información.
  - `description` es una descripción breve del código de cálculo.
  - `language_id` es el idioma para el que se aplica esta información.
  - `longdescription` es la descripción detallada del código de cálculo.
- c. Repita los pasos a y b para cada código de cálculo utilizado en la tienda.
8. Defina las normas de cálculo para la tienda.

- a. Utilizando el ejemplo siguiente como guía, configure la norma de cálculo en el archivo XML para la tabla CALRULE:

```
<calrule
calrule_id="@calrule_id_1"
calcode_id="@calcode_id_1"
startdate="1900-01-01 00:00:00.000000"
enddate="2100-01-01 00:00:00.000000"
sequence="+1.0000000000000000E+000"
combination="2"
calmethod_id="-27"
calmethod_id_qfy="-26"
flags="1"
identifier="1" />
```

- `calrule_id` es un identificador exclusivo generado.
- `calcode_id` es el código de cálculo del que forma parte esta norma de cálculo.
- `startdate` es el momento en el que entra en vigor esta norma de cálculo.
- `enddate` es la hora en que esta norma de cálculo deja de estar en vigor.
- `sequence` es el orden en el que se procesará esta norma de cálculo. Las normas de cálculo para el mismo código de cálculo se procesan en secuencia del valor más bajo al más alto.
- `combination` especifica el distintivo de bit para indicar el proceso especial que deberá realizar la implementación por omisión de `CalculationRuleCombineMethod`. Consulte la tabla CALRULE en la ayuda en línea de WebSphere Commerce para obtener más información.
- `calmethod_id` es el método `CalculationRuleCalculateMethod` que calcula un resultado monetario para un conjunto de artículos de pedido.
- `calmethod_id_qfy` es el método `CalculationRuleQualifyMethod` que determina qué artículo de pedido de un conjunto de artículos de pedido debe enviarse al método `CalculationRuleCalculateMethod`.
- `flags`: los utiliza el método `CalculationRuleCombineMethod` para determinar cómo se puede combinar esta norma de cálculo con otras normas de cálculo. Consulte la tabla CALRULE para obtener más información.
- `identifier` identifica la norma de cálculo, en combinación con el código de cálculo de la misma.






Para obtener más información, consulte la tabla CALRULE en la ayuda en línea de WebSphere Commerce.


- b. Repita el paso a para cada norma de cálculo utilizada en la tienda. Tenga en cuenta que cada código de cálculo puede tener varias normas de cálculo. Por ejemplo, `calcode_id="@calcode_id_1"` puede estar asociado a varios `calrule_id`.
9. Defina escalas de cálculo para la tienda.
- Una escala de cálculo es el conjunto de rangos que se aplicarán al cálculo. Por ejemplo, para los costes de envío puede tener un conjunto de rangos de peso que correspondan cada uno a un coste determinado. Es decir, un producto que pese entre 0 y 5 kg puede tener un coste de envío de 10 euros. Y un producto que pese entre 5 y 10 kg puede tener un coste de envío de 15 euros. Estos rangos crean una escala.

- a. Utilizando el ejemplo siguiente como guía, configure la escala de cálculo en el archivo XML para la tabla CALSCALE:

```
<calscale
 calscale_id="@calscale_id_1"
 code="Código de escala 1 por pedido EUR"
 storeent_id="@storeent_id_1"
 calusage_id="-2"
 setcurr="EUR"
 calmethod_id="-28"/>
```

donde

- `calscale_id` es un identificador exclusivo generado.
- `code` es una serie de caracteres que identifica de forma exclusiva esta escala de cálculo, dado un uso de cálculo y una entidad de tienda determinados.
- `storeent_id` es la entidad de tienda de la que forma parte esta escala de cálculo.
- `calusage_id` indica la clase de cálculo para la que se utiliza esta escala de cálculo. Por ejemplo, la escala de cálculo puede utilizarse para calcular uno de los importes monetarios siguientes:
  - descuentos (-1)
  - costes de envío (-2)
  - impuestos sobre la venta (-3)
  - impuestos sobre el envío (-4)
  - cupones (-5)
- `setcurr`, si se especifica, indica la moneda para los valores de inicio de rango de los objetos de rango de cálculo para esta escala de cálculo. El método `CalculationScaleLookupMethod` debe devolver un "número de búsqueda" en esta moneda.
- `calmethod_id` es el método `CalculationScaleLookupMethod` que, dado un conjunto de artículos de pedido, determina un valor de búsqueda, un valor monetario base, un multiplicador de resultado y un conjunto de pesos matemáticos que la escala de cálculo puede utilizar para calcular el importe monetario. Para determinar qué método `CalculationScaleLookupMethod` se debe utilizar, realice lo siguiente:
  - Consulte la tabla CALMETHOD en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna SUBCLASS. Pulse el enlace para la tabla CALMETHOD: detalles. Esta tabla lista los tipos de métodos de cálculo disponibles. El método `MonetaryCalculationScaleLookupMethod` es 9.
  - Abra el archivo de rutina de carga `wcs.bootstrap_xx_XX.xml`, donde `xx_XX` es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:
    -  `NT` unidad:\WebSphere\CommerceServer\schema\xml
    -  `2000` unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml
    -  `AIX` /usr/WebSphere/CommerceServer/schema/xml
    -  `Solaris` /opt/WebSphere/CommerceServer/schema/xml
    -  `Linux` /opt/WebSphere/CommerceServer/schema/xml

-  /qibm/proddata/WebCommerce/schema/xml
- Localice la sección que lista los métodos de cálculo disponibles (CALMETHOD).
- Localice los métodos de cálculo con el valor calusage\_ID para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
- Localice los métodos de cálculo, que tienen una subclase de 7; hay varios. Seleccione uno que satisfaga sus necesidades.

Para obtener más información, consulte la tabla CALSCALE en la ayuda en línea de WebSphere Commerce.







- b. Repita el paso a para cada escala de cálculo utilizada en la tienda. Por ejemplo, para el envío, InFashion crea una escala de coste por pedido y una escala de coste por artículo.
10. Defina rangos de cálculo para las escalas de cálculo.

- a. Utilizando el ejemplo siguiente como guía, configure el rango de cálculo en el archivo XML para la tabla CALRANGE.

```
<calrange
calrange_id="@calrange_id_1"
calscale_id="@calscale_id_1"
calmethod_id="-33"
rangestart="0.00000"
cumulative="0"/>
```

donde

- calrange\_id es un identificador exclusivo generado.
- calscale\_id es la escala de cálculo de la que forma parte este rango de cálculo.
- calmethod\_id es el método CalculationRangeMethod que determina un importe monetario a partir de CalculationRangeLookupResult. Por ejemplo, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd o PercentageCalculationRangeCmd. Para determinar el método CalculationRangeMethod, realice lo siguiente:
  - Consulte la tabla CALMETHOD en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna SUBCLASS. Pulse el enlace para la tabla CALMETHOD: detalles. Esta tabla lista los tipos de métodos de cálculo disponibles. El método CalculationRangeMethod es 10.
  - Abra el archivo de rutina de carga wcs.bootstrap\_xx\_XX.xml, donde xx\_XX es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\schemaschema.xml
-  unidad:\Archivos de programa\WebSphere\CommerceServer\schemaschema.xml
-  /usr/WebSphere/CommerceServer/schemaschema.xml
-  /opt/WebSphere/CommerceServer/schemaschema.xml
-  /opt/WebSphere/CommerceServer/schemaschema.xml
-  /qibm/proddata/WebCommerce/schema/xml

- Localice la sección que lista los métodos de cálculo disponibles (CALMETHOD).
- Localice los métodos de cálculo con el valor calusage\_ID para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
- Localice los métodos de cálculo que tienen una subclase de 9; hay varios. Seleccione uno que satisfaga sus necesidades.
- cumulative son los valores válidos:
  - 0 = sólo se utiliza el rango de cálculo coincidente que tiene el valor RANGESTART más alto.
  - 1 = se utilizan todos los rangos de cálculo coincidentes. Los importes monetarios calculados se suman para llegar al resultado final.

Para obtener más información, consulte la tabla CALRANGE en la ayuda en línea de WebSphere Commerce.

- b. Repita el paso a para cada rango de cálculo asociado a la escala de cálculo utilizada en la tienda.
11. Defina los valores de búsqueda de cálculo para las escalas de cálculo. Los valores de búsqueda de cálculo son los valores asociados a la escala de cálculo. Por ejemplo, una escala de cálculo incluye los rangos de peso y los precios asociados siguientes para el envío:
- De 0 a 5 kg cuesta 10 euros
  - De 5 a 10 kg cuesta 15 euros

Los valores de búsqueda son 10 euros y 15 euros.

- a. Utilizando como guía los ejemplos siguientes, configure valores de búsqueda de cálculo en el archivo XML para la tabla CALRLOOKUP. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional, es decir, un archivo por cada entorno nacional que soporte la tienda. Por ejemplo, si la tienda realiza envíos a los clientes de España y EE.UU., deberá añadir los valores de búsqueda de euro en un archivo XML y los valores de búsqueda de dólar americano en otro archivo XML.

```
<calrlookup
calrlookup_id="@calrlookup_id_1"
setccurr="EUR"
calrange_id="@calrange_id_1"
value="500"/>
```

donde

- calrlookup\_id es un identificador exclusivo generado.
- calrange\_id es el rango de cálculo del que forma parte este resultado de búsqueda de rango de cálculo.
- value es el valor del resultado de búsqueda de rango de cálculo, utilizado por el método del rango de cálculo para determinar un resultado monetario.

Para obtener más información, consulte la tabla CALRLOOKUP en la ayuda en línea de WebSphere Commerce.

- b. Repita el paso a para cada valor de búsqueda asociado a la escala de cálculo utilizada en la tienda.
12. Asocie la norma de cálculo y la escala de cálculo

- a. Utilizando los ejemplos siguientes como guía, asocie la escala de cálculo a la norma de cálculo en el archivo XML para la tabla CRULESCALE.

```
< crulescale
 calrule_id="@calrule_id_1"
 calscale_id="@calscale_id_1" />
```

donde

- calrule\_id es la norma de cálculo.
- calscale\_id es la escala de cálculo.

- b. Repita el paso a para cada escala de cálculo y asociación de norma.



Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, "Creación de los datos" en la página 327.

## Creación de elementos de despacho de envío




Para que los elementos de envío funcionen correctamente en la tienda, deberá asociar los grupos de jurisdicciones de envío con las normas de cálculo y los centros de despacho de pedidos con las modalidades de envío utilizadas en la tienda.

Deberá crear los elementos de despacho para poder asociar los elementos de envío a un centro de despacho de pedidos. Para obtener más información sobre cómo crear elementos de despacho, consulte "Creación de elementos de despacho de pedidos en WebSphere Commerce" en la página 114.







Después de haber creado los elementos de despacho de pedidos, asocie a éstos los elementos de envío añadiendo información en las tablas SHPJCRULE y SHPARRANGE. Efectúe lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, envío) asociados al producto o servicio que un cliente ha seleccionado para adquirir.
2. Revise los archivos XML utilizados para crear elementos de despacho de envío para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente. Cada tienda de ejemplo incluye un archivo `shipfulfill.xml`, que incluye la información de despacho de envío. Para ver el archivo `shipfulfill.xml` del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo `shipfulfill.xml` está ubicado en el directorio de datos.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

3. Revise la información del Apéndice B, "Creación de los datos" en la página 327.
4. Cree un archivo `shipfulfill.xml`, copiando uno de los archivos `shipfulfill.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a `shipfulfill.xml`. Los archivos DTD se encuentran en el directorio siguiente:
  -  `unidad:\WebSphere\CommerceServer\xml\sar`
  -  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar`
  -  `/usr/WebSphere/CommerceServer/xml/sar`
  -  `/opt/WebSphere/CommerceServer/xml/sar`
  -  `/opt/WebSphere/CommerceServer/xml/sar`
  -  `/qibm/proddata/WebCommerce/xml/sar`
5. Asocie normas de cálculo a un grupo de jurisdicciones de envío añadiendo información en la tabla `SHPJCRULE`. Utilice como guía el ejemplo siguiente. Si está creando una tienda multicultural, cree también un archivo XML para cada entorno nacional que la tienda soporte.

```
<shpjcrule
calrule_id="@calrule_id_1"
ffmcenter_id="@ffmcenter_id_1"
jurstgroup_id="@jurstgroup_id_1"
precedence="0"
shipmode_id="@shipmode_id_1"
shpjcrule_id="@shpjcrule_id_1"
```

donde

- `calrule_id` es la norma de cálculo utilizada.
  - `ffmcenter_id` es el centro de despacho de pedidos. Si es `NULL`, esta asociación se aplica a todos los centros de despacho de pedidos.
  - `jurstgroup_id` es el grupo de jurisdicciones de envío. Si es `NULL`, esta asociación se aplica a todos los grupos de jurisdicciones de envío.
  - `precedence` se aplica cuando una dirección de envío está en más de uno de los grupos de jurisdicciones de envío especificados para el mismo centro de despacho de pedidos y la misma modalidad de envío. Sólo es apta la norma de cálculo que tiene el valor de `SHPJCRULE.PRECEDENCE` más alto.
  - `shipmode_id` es la modalidad de envío.
  - `shpjcrule_id` es un identificador exclusivo generado.
6. Repita el paso 3 para cada grupo de jurisdicciones, centro de despacho de pedidos y asociación de normas de la tienda.
  7. Asocie la modalidad de envío y un centro de despacho de pedidos a la tienda, añadiendo información en la tabla `SHPARRANGE`. Utilice el ejemplo siguiente como guía:

```
<shparrange
shparrange_id="@shparrange_id_2"
store_id="@storeent_id_1"
ffmcenter_id="@ffmcenter_id_1"
shipmode_id="@shipmode_id_2"
startdate="1970-06-22 23:00:00.000000"
```



```
enddate= "2008-06-22 23:00:00.000000"
precedence= "0"
flags="0"
</>
```

donde

- shparrange\_id es un identificador exclusivo generado.
- store\_id es la tienda.
- ffmcenter\_id es el centro de despacho de pedidos.
- shipmode\_id es la modalidad de envío. NULL indica que este acuerdo de envío puede utilizarse independientemente de la modalidad de envío.
- startdate es el momento en que este acuerdo de envío empieza a estar en vigor.
- enddate es el momento en que este acuerdo de envío deja de estar en vigor.
- precedence se aplica cuando en un momento determinado están en vigor más de un acuerdo de envío (para la misma tienda y modalidad de envío); se utiliza el que tiene la prioridad (PRECEDENCE) más alta.
- flags contiene distintivos de bit:
  - 1 = restringido - Este acuerdo de envío sólo se aplica a artículos de pedido cuya dirección de envío coincide con uno de los grupos de jurisdicciones de envío asociados (mediante la tabla SHPARJURGP) con este acuerdo de envío.

8. Repita el paso 5 para todas las modalidades de envío utilizadas en la tienda.



---

Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, "Creación de los datos" en la página 327.

---

## Creación de elementos de tienda-catálogo-envío

Para asociar modalidades de envío con la tienda, deberá asociar un código de cálculo con las entradas de catálogo de la tienda para cada contrato que la tienda incluya.






Para poder crear elementos de tienda-catálogo-envío deberá crear elementos de tienda y de catálogo. Para obtener más información sobre cómo crear elementos de tienda, consulte "Creación de elementos de datos de tienda en un archivo XML" en la página 45. Para obtener más información sobre cómo crear elementos de catálogo, consulte "Visualización de los elementos de catálogo de la tienda" en la página 77.

Para crear elementos de tienda-catálogo-envío, realice lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, envío) asociados al producto o servicio que un cliente ha seleccionado para adquirir.
2. Revise los archivos XML utilizados para crear elementos de despacho de envío para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:







-  unidad:\WebSphere\CommerceServer\samplestores

-  unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo store-catalog-shipping.xml, que incluye la información de despacho de envío. Para ver el archivo store-catalog-shipping.xml del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo store-catalog-shipping.xml está ubicado en el directorio de datos.

3. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
4. Cree un archivo store-catalog-shipping.xml, copiando uno de los archivos store-catalog-shipping.xml de los archivadores de tienda de ejemplo o bien creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a store-catalog-shipping.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\xml\sar
-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. Cree la relación de tienda-catálogo-envío añadiendo información a la tabla CATENCALCD. Utilice el ejemplo siguiente como guía:

```
<catencalcd
 calcode_id="@calcode_id_1"
 catencalcd_id="@catencalcd_id_1"
 store_id="@storeent_id_1"
/>
```

donde

- calcode\_id es el código de cálculo.
- catencalcd\_id es un identificador exclusivo generado.
- store\_id es la tienda.




---

Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, “Creación de los datos” en la página 327.







---

## Creación de una modalidad de envío por omisión

Para establecer una modalidad de envío por omisión para la tienda, deberá añadir información en la tabla STOREDEF. Para añadir información a la tabla STOREDEF, realice lo siguiente:

1. Revise los archivos XML utilizados para crear elementos por omisión de tienda para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.







Los archivos de archivador de tienda están ubicados en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo store-defaults.xml, que incluye la información de envío por omisión. Para ver el archivo store-defaults.xml del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo store-defaults.xml está ubicado en el directorio de datos.

2. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
3. Cree un archivo store-defaults.xml, copiando uno de los archivos store-defaults.xml de los archivadores de tienda de ejemplo o bien creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a store-defaults.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

4. Utilizando como guía el ejemplo siguiente, en el archivo XML, especifique la modalidad de envío por omisión para la tienda añadiendo información en la tabla STOREDEF:

```
<storedef
 store_id="@storeent_id_1"
 shipmode_id="@shipmode_id_1"
/>
```

donde

- store\_id es la tienda.
- shipmode\_id es la modalidad de envío por omisión para la tienda.



Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, “Creación de los datos” en la página 327.

---

## Capítulo 19. Elementos de impuestos

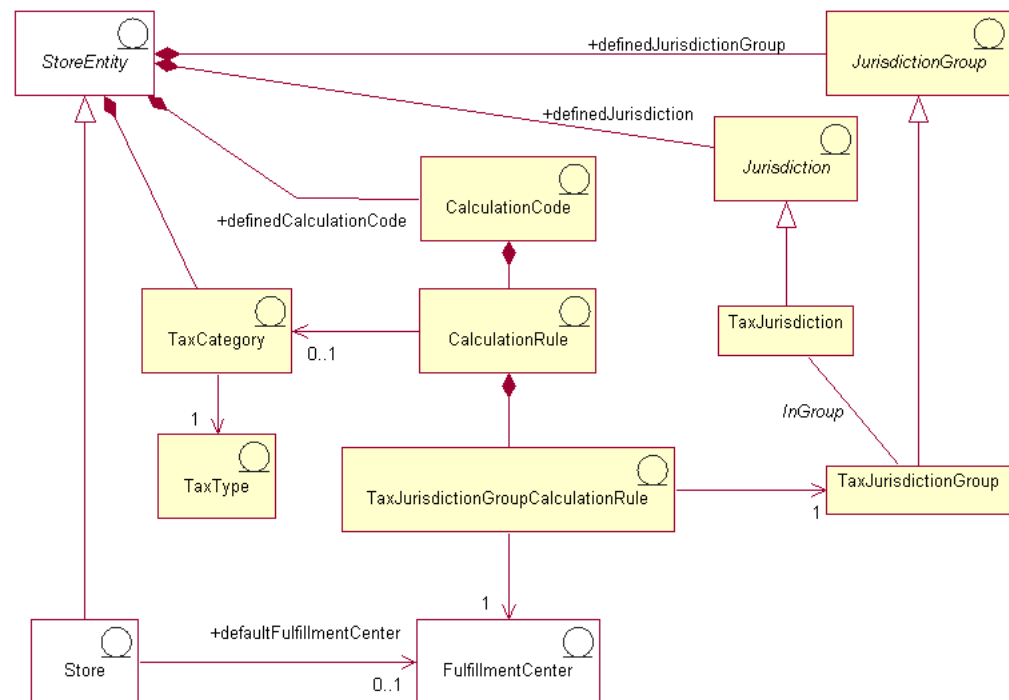
Para cargar o recaudar impuestos sobre las mercancías y los servicios que la tienda proporciona, una tienda creada con WebSphere Commerce debe incluir lo siguiente:

- Categorías de impuestos
- Códigos de cálculo
- Jurisdicciones y grupos de jurisdicciones

La combinación de las categorías de impuestos, los códigos de cálculo y las jurisdicciones y grupos de jurisdicciones crea las tarifas de impuestos para la tienda.

### Los elementos de impuestos en WebSphere Commerce

El diagrama siguiente ilustra la estructura de fijación de impuestos en WebSphere Commerce Server.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

## Categoría de impuestos

Las *categorías de impuestos* corresponden a las diferentes clases de impuestos que una tienda puede tener que recaudar, por ejemplo federales, estatales, provinciales y municipales.

Una categoría de impuestos forma parte de una entidad de tienda, aunque una entidad de tienda pueda tener varias categorías de impuestos. Si se suprime la entidad de tienda, también se suprimirán las categorías de impuestos asociadas a dicha entidad de tienda.

### Tipo de impuestos

Una tienda recauda normalmente dos tipos de impuestos: impuestos sobre las ventas o el uso, e impuestos sobre el envío. Cada categoría de impuestos tiene un *tipo de impuesto*. Aunque cada categoría de impuestos puede ser sólo de un tipo de impuesto, (por ejemplo, la categoría de impuestos federal es un tipo de impuestos sobre la venta), varias categorías de impuestos diferentes pueden pertenecer al mismo tipo de impuesto (por ejemplo, el tipo de impuestos sobre la venta se aplica a las categorías federal, provincial y municipal).

## Código de cálculo

Los *códigos de cálculo* se utilizan para calcular el importe de los impuestos, es decir, un código de cálculo de impuestos indica cómo se calculan los impuestos para los artículos de pedido. Para calcular el impuesto sobre un artículo del pedido, debe asignar códigos de cálculo de impuestos sobre la venta y de impuestos sobre el envío a una entrada de catálogo o a un grupo de entradas de catálogo. Sólo se puede aplicar un código de cálculo de impuestos de cada tipo de impuesto a una entrada de catálogo determinada o a un grupo de entradas de catálogo. Normalmente, los impuestos sobre las ventas o el uso se calculan sobre el precio neto y los impuestos sobre el envío se calculan sobre los costes de envío.

Un código de cálculo forma parte de una entidad de tienda. Un código de cálculo sólo se puede asociar a una entidad de tienda, pero una entidad de tienda puede tener varios códigos de cálculo. Si se suprime la entidad de tienda, también se suprimirán los códigos de cálculo asociados a dicha entidad de tienda.



Para obtener más información sobre el uso de códigos de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

---

### Normas de cálculo

Todo código de cálculo tiene como mínimo una *norma de cálculo*, que define los cálculos de cada categoría de impuesto y especifica las condiciones bajo las que se realizarán los cálculos. Cada norma de cálculo de impuestos está asociada a una categoría de impuesto, un grupo de jurisdicciones y un centro de despacho de pedidos, que juntos definen las condiciones bajo las cuales se utilizará la norma de cálculo. Por ejemplo, se puede seleccionar una norma diferente para calcular un importe para una categoría de impuestos determinada dependiendo de la dirección de envío y del centro de despacho de pedidos especificados en el pedido.

Cada norma de cálculo pertenece exactamente a un código de cálculo.

Un código de cálculo de impuestos determinado puede tener varias normas de cálculo, una para cada combinación de categoría de impuestos, grupo de jurisdicciones fiscales y centro de despacho de pedidos asociado a la tienda. Todas las normas de cálculo de impuestos sobre la venta y de impuestos sobre el envío se

pueden asociar con varias TaxJurisdictionGroupCalculationRules (TaxRules). Por ejemplo en el diagrama siguiente, la norma de cálculo 10001 se puede aplicar a los grupos de jurisdicciones 1234 y 1235.

| TAXJCRULE_ID | CALRULE_ID | FFMCENTER_ID | JURSTGROUP_ID | PRECEDENCE |
|--------------|------------|--------------|---------------|------------|
| 10001        | 10001      | NULL         | 1234          | 0          |
| 10002        | 10001      | NULL         | 1235          | 0          |

Cada TaxRule define las condiciones bajo las cuales se aplicará la norma de cálculo. Por ejemplo, puede definir una norma de cálculo para cada grupo de jurisdicciones a los que la tienda realiza envíos. En el ejemplo siguiente, la norma de cálculo 10001 se puede aplicar a los grupos de jurisdicciones 1234 y 1235.

En el ejemplo siguiente, el código de cálculo de impuestos utiliza la norma de cálculo A para la categoría provincial de impuestos sobre la venta, cuando la jurisdicción fiscal es Alberta y la norma C cuando la jurisdicción fiscal es British Columbia.

| Jurisdicción fiscal      | Impuestos sobre la venta federales | Impuestos sobre la venta provinciales |
|--------------------------|------------------------------------|---------------------------------------|
| Alberta, Canadá          | Norma de cálculo B, que da Y%      | Norma de cálculo A, que da X%         |
| British Columbia, Canadá | Norma de cálculo B, que da Y%      | Norma de cálculo C, que da Z%         |

Cuando una dirección de envío coincide con más de un grupo de jurisdicciones fiscales, se utiliza la norma de cálculo con el valor de columna TAXJCRULE.PRECEDENCE asociada más alto.

La asociación de TaxJurisdictionGroupCalculationRules (TaxRule) con una norma de cálculo determina cuándo se puede aplicar la norma de cálculo. Una norma de cálculo de impuestos sobre la venta o de impuestos sobre el envío es aplicable cuando se satisfacen cualquiera de las condiciones determinadas por TaxRules. En el ejemplo siguiente, la norma de cálculo 10001 es aplicable cuando el envío se realiza al grupo de jurisdicciones 1001 o cuando el envío se realiza desde el centro de despacho de pedidos 1001 o si está realizando el envío al grupo de jurisdicciones 1001.

| CALRULE_ID | FFMCENTER_ID | JURSTGROUP_ID |
|------------|--------------|---------------|
| 10001      | NULL         | 1001          |
| 10001      | 1001         | 1001          |

Toda TaxJurisdictionGroupCalculationRule está asociada con un grupo de jurisdicciones como mínimo. Las normas de cálculo por sí mismas no están asociadas directamente con los grupos de jurisdicciones.



Para obtener más información sobre el uso de normas de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

## Jurisdicciones y grupos de jurisdicciones

Las *jurisdicciones* son regiones o zonas geográficas que representan un país o una región, una provincia o un territorio, o un rango de códigos postales, en los que se venden mercancías. Las jurisdicciones se agrupan para formar *grupos de jurisdicciones*.

WebSphere Commerce soporta dos tipos de jurisdicciones: jurisdicciones de envío y jurisdicciones fiscales. Cada una de estas jurisdicciones forma parte del grupo correspondiente, por ejemplo, las jurisdicciones de envío están en el grupo de jurisdicciones de envío y las jurisdicciones fiscales están en el grupo de jurisdicciones fiscales.

Las jurisdicciones y los grupos de jurisdicciones determinan qué normas de cálculo se utilizan para calcular los impuestos.

Las jurisdicciones y los grupos de jurisdicciones forman parte de una entidad de tienda. Cada jurisdicción y grupo de jurisdicciones forma parte de una entidad de tienda, sin embargo una entidad de tienda puede tener varias jurisdicciones y grupos de jurisdicciones. Si se suprime la entidad de tienda, también se suprimirán las jurisdicciones y los grupos de jurisdicciones asociados a dicha entidad de tienda.



---

Para obtener información más detallada sobre la estructura de los elementos de impuestos en WebSphere Commerce Server, consulte los modelos de datos y de objeto de impuestos en la ayuda en línea de WebSphere Commerce.

---

## Creación de elementos de impuestos en WebSphere Commerce

Las herramientas de Servicios de tienda de WebSphere Commerce le permiten crear y editar determinados elementos de impuestos (por ejemplo categorías de impuestos y jurisdicciones fiscales) en un archivador de tienda, pero no todos los elementos de impuestos. Para obtener más información sobre qué elementos puede editar con las herramientas de Servicios de tienda, consulte el tema "*Cambiar los elementos de base de datos de la tienda*" de la ayuda en línea de WebSphere Commerce.

**Nota:** Las herramientas de los Servicios de tienda trabajan con archivos XML con datos preinsertados, en forma de archivador de tienda.

También puede crear los elementos de impuestos en formato de archivos XML que se pueden cargar en la base de datos utilizando Loader package. De este modo, tiene las dos opciones siguientes para crear elementos de envío:

- editar los elementos de impuestos existentes de una de las tiendas de ejemplo proporcionadas con WebSphere Commerce o de un archivador de tienda existente
- crear elementos de impuestos nuevos en forma de archivo XML que se puede publicar como parte de un archivador de tienda o cargar utilizando Loader package.

Para obtener información sobre cómo editar los elementos de impuestos de un archivador de tienda existente, consulte la ayuda en línea de WebSphere Commerce. Para obtener información sobre cómo crear elementos de impuestos nuevos en forma de archivo XML, consulte "Creación de elementos de impuestos utilizando un archivo XML" en la página 159.



## Creación de elementos de impuestos utilizando un archivo XML







Cree los elementos de impuestos en formato de archivos XML que se puedan cargar en la base de datos utilizando Loader package. Para obtener más información sobre Loader package, consulte la Parte 7, “Publicación de la tienda” en la página 225. Si crea una tienda multicultural, deberá crear archivos XML para cada entorno nacional al que la tienda dé soporte. El archivo específico de entorno nacional debe especificar toda la información de las descripciones, para que se pueda traducir fácilmente.

Las tiendas de ejemplo, de las que se toman muchos de los ejemplos de estas tareas, utilizan un archivo `tax.xml` para toda la información que no es necesario traducir y otro archivo `tax.xml` para cada entorno nacional que la tienda soporta, para la información que es necesario traducir. Los archivos específicos de entorno nacional contienen toda la información de las descripciones.

Para crear elementos de impuestos para la tienda utilizando un archivo XML, realice lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, impuestos) asociados al producto o servicio que un cliente ha seleccionado para adquirir.
2. Revise los archivos XML utilizados para crear elementos de impuestos para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:






-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye dos archivos `tax.xml`, que contienen la información de impuestos. Para ver los archivos `tax.xml` del archivador de tienda, descompríalos utilizando un programa ZIP. Los archivos `tax.xml` están ubicados en el directorio de datos. El archivo `tax.xml` específico de un idioma está situado en el subdirectorío específico del entorno nacional del directorio de datos.

3. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
4. Cree un archivo `tax.xml`, copiando uno de los archivos `tax.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a `tax.xml`. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar

-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. Defina las jurisdicciones y el grupo de jurisdicciones a los que está enviando mercancías y servicios. Asigne las jurisdicciones fiscales a los grupos de jurisdicciones según sus normas de cálculo de categorías de impuestos.

a. Utilizando el ejemplo siguiente como guía, defina un grupo de jurisdicciones en el archivo XML de la tabla JURSTGROUP:

```
<jurstgroup
jurstgroup_id="@jurstgroup_id_2"
description="Grupo de jurisdicciones fiscales 1"
subclass="2"
storeent_id="@storeent_id_1"
code="Mundial"/>
```

donde

- `jurstgroup_id` es una clave exclusiva generada
- `description` es una breve descripción del grupo de jurisdicciones, adecuada para visualizarse en una interfaz de usuario que gestione grupos de jurisdicciones.
- `subclass` es la subclase del grupo de jurisdicciones tal como se indica a continuación:
  - 1 = ShippingJurisdictionGroup
  - 2 = TaxJurisdictionGroup
- `storeent_id` es la entidad de tienda asociada a este grupo de jurisdicciones.
- `code` que, junto con la entidad de tienda y la subclase, identifica de forma exclusiva este grupo de jurisdicciones.

b. Utilizando el ejemplo siguiente como guía, defina una jurisdicción en el archivo XML de la tabla JURST.

```
<jurst
jurst_id="@jurst_id_2"
storeent_id="@storeent_id_1"
code="Mundial"
subclass="2"/>
```

donde

- `jurst_id` es una clave exclusiva generada
- `storeent_id` es la entidad de tienda asociada a este grupo de jurisdicciones.
- `code` que, junto con la entidad de tienda y la subclase, identifica de forma exclusiva este grupo de jurisdicciones.
- `subclass` es la subclase de jurisdicción tal como se indica a continuación:
  - 1 = ShippingJurisdiction
  - 2 = TaxJurisdiction

- c. Utilizando el ejemplo siguiente como guía, asocie la jurisdicción que ha creado en el paso b al grupo de jurisdicciones que ha definido en el paso a, añadiendo información a la tabla JURSTGRPREL.

```
<jurstgprel
 jurst_id="@jurst_id_2"
 jurstgroup_id="@jurstgroup_id_1"
 subclass="2"/>
```

donde

- jurst\_id es la jurisdicción
- jurstgroup\_id es el grupo de jurisdicciones
- subclass es la subclase de la jurisdicción y del grupo de jurisdicciones. Estas deben coincidir:
  - 1 = ShippingJurisdiction[Group]
  - 2 = TaxJurisdiction[Group]

- d. Repita los pasos a a c para todas las jurisdicciones y todos los grupos de jurisdicciones que la tienda soporte.

6. Defina las categorías de impuestos que utilizará la tienda.

- a. Utilizando el ejemplo siguiente como guía, defina una categoría de impuestos en el archivo XML para la tabla TAXCGRY:

```
<taxcgry
 taxcgry_id="@taxcgry_id_1"
 taxtype_id="-3"
 storeent_id="@storeent_id_1"
 name="Impuestos sobre la venta"
 displayseq="0"
 displayusage="0"/>
```

donde:

- taxcgry\_id es una clave exclusiva generada.
- taxtype\_id="-3" es el tipo de impuestos para esta categoría de impuestos. WebSphere Commerce soporta dos tipos de impuestos:
  - impuestos sobre la venta o el uso (-3)
  - impuestos sobre el envío (-4)
- storeent\_id es la entidad de tienda asociada a esta categoría de impuestos.
- name es el nombre de la categoría de impuestos. Junto con la entidad de tienda, el nombre identifica esta categoría de impuestos de forma exclusiva.
- displayseq especifica la secuencia, del más bajo al más alto, de los importes de los impuestos cuando se visualizan, por ejemplo, en un pedido.
- displayusage especifica que esta categoría de impuestos en relación con el bean de datos de precio es la siguiente:
  - 0 = no se calcula
  - 1 = se calcula

El bean de datos de precio puede utilizarse para obtener importes de impuestos que deben mostrarse junto con el precio del producto.

- b. Repita el paso a para cada categoría de impuestos utilizada en la tienda.
- c. Utilice el ejemplo siguiente como guía y añada la información de descripción de categoría de impuestos a su archivo XML para la tabla

TAXCGRYDS. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional.

```
<taxcgryds
 taxcgry_id="@taxcgry_id_1"
 description="Impuestos sobre la venta"
 language_id("&es_ES"/>
```

donde

- `taxcgry_id` es la categoría de impuestos.
- `description` es una descripción breve de la categoría de impuestos, adecuada para mostrarla a los clientes.
- `language_id` es el idioma en que se visualiza esta información.

d. Repita el paso c para cada categoría de impuestos utilizada en la tienda.







7. Defina los códigos de cálculo que deberá utilizar la tienda.

a. Utilizando los ejemplos siguientes como guía, defina el código de cálculo en el archivo XML para la tabla CALCODE.

```
<calcode
 calcode_id="@calcode_id_3"
 code="Código de impuestos 1"
 calusage_id="-3"
 storeent_id="@storeent_id_1"
 groupby="0"
 published="1"
 sequence="0"
 calmethod_id="-43"
 calmethod_id_app="-44"
 calmethod_id_qfy="-42"
 displaylevel="0"
 flags="0"
 precedence="0"
/>
```

donde:

- `calcode_id` es una clave exclusiva generada.
- `code` es una serie de caracteres que identifica de forma exclusiva este código de cálculo, dado un uso de cálculo y una entidad de tienda determinados.
- `calusage_id` indica la clase de cálculo para la que se utiliza este código de cálculo. Por ejemplo, el código de cálculo puede utilizarse para calcular uno de los importes monetarios siguientes:
  - descuentos (-1)
  - costes de envío (-2)
  - impuestos sobre la venta (-3)
  - impuestos sobre el envío (-4)
  - cupones (-5)
- `storeent_id` es la entidad de tienda asociada a este código de cálculo.
- `groupby` son distintivos de bit que indican al método de combinación de código de cálculo cómo deben agruparse los artículos de pedido al realizar los cálculos. Cero especifica que no se agrupan (todos los artículos de pedido están en un solo grupo). Consulte *Tabla CALCODE: detalles* en la ayuda en línea de WebSphere Commerce para obtener más información.
- `published` especifica si el código de cálculo se publica o no:
  - 0 = no publicado (inhabilitado temporalmente)

- 1 = publicado
- 2 = marcado para supresión (y no publicado)
- sequence es el orden en el que se calcula el código de cálculo. Los códigos de cálculo se calculan y se aplican en secuencia del más bajo al más alto. Si dos códigos de cálculo tienen el mismo número de secuencia, se calcularán primero los códigos de cálculo cuyo calcode\_id sea menor.
- calmethod\_id: método de cálculo del código de cálculo que define cómo se calculan los importes de los impuestos para este código de cálculo. Para determinar qué método de cálculo de código de cálculo se debe utilizar, realice lo siguiente:
  - Consulte la tabla CALMETHOD en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna SUBCLASS. Pulse el enlace para la tabla CALMETHOD: detalles. Esta tabla lista los tipos de CALMETHOD disponibles. El método de cálculo del código de cálculo es 3.
  - Abra el archivo de rutina de carga wcs.bootstrap\_xx\_XX.xml, donde xx\_XX es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:
    -  NT unidad:\WebSphere\CommerceServer\schema\xml
    -  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml
    -  AIX /usr/WebSphere/CommerceServer/schema/xml
    -  Solaris /opt/WebSphere/CommerceServer/schema/xml
    -  Linux /opt/WebSphere/CommerceServer/schema/xml
    -  400 /qibm/proddata/WebCommerce/schema/xml
  - Localice la sección que lista los métodos de cálculo disponibles (CALMETHOD).
  - Localice los métodos de cálculo con el valor calusage\_ID para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
  - Localice el método de cálculo que tiene la subclase de 3. Este método de cálculo es -43.
- calmethod\_id\_app es el método CalculationCodeApplyMethod que almacena el importe calculado para los artículos de pedido asociados. Utilice el método descrito en calmethod\_id para determinar qué método de aplicación de código de cálculo se debe utilizar.
  - calmethod\_id\_app="-44" es el método CalculationCodeApplyMethod para Impuesto sobre la venta
- calmethod\_id\_qfy es el método CalculationCodeQualifyMethod que define qué artículos de pedido están asociados a este código de cálculo. Utilice el método descrito en calmethod\_id para determinar qué método de calificación de código de cálculo se debe utilizar.
  - calmethod\_id\_qfy="-42" es el método CalculationCodeQualifyMethod para Impuesto sobre la venta.
- display level determina si los importes calculados por este código de cálculo deben visualizarse con cada:
  - 0 = Artículo de pedido
  - 1 = Pedido

- 2 = Producto
- 3 = Artículo
- 4 = Contrato
- flags especifica si se debe invocar el método CalculationCodeQualifyMethod de este código de cálculo.
  - 0 = no restringido. El método no se invocará
  - 1 = restringido. El método se invocará.
- b. Utilizando el ejemplo siguiente como guía, añada la información de descripción de código de cálculo al archivo XML para la tabla CALCODEDSC. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional.

```
<calcodedsc
 calcode_id="@calcode_id_3"
 description="Vitaminas
 language_id="&es_ES"
 longdescription="En Ontario, a las vitaminas se les aplican impuestos
 federales, pero no provinciales."
/>
```







donde

- calcode\_id es el código de cálculo al que se aplica esta información.
  - description es una descripción breve del código de cálculo.
  - language\_id es el idioma para el que se aplica esta información.
  - longdescription es la descripción detallada del código de cálculo.
- c. Repita los pasos a y b para cada código de cálculo utilizado en la tienda.
8. Defina las normas de cálculo para la tienda.
- a. Utilizando el ejemplo siguiente como guía, configure la norma de cálculo en el archivo XML para la tabla CALRULE:

```
<calrule
 calrule_id="@calrule_id_10"
 calcode_id="@calcode_id_3"
 startdate="1900-01-01 00:00:00.000000"
 taxcgr_id="@taxcgr_id_1"
 enddate="2100-01-01 00:00:00.000000"
 flags="1"
 identifier="1"
 combination="2"
 calmethod_id="-47"
 calmethod_id_qfy="-46"
/>
```

donde

- calrule\_id es un identificador exclusivo generado.
- calcode\_id es el código de cálculo del que forma parte esta norma de cálculo.
- startdate es el momento en el que entra en vigor esta norma de cálculo.
- taxcgr\_id es la categoría de impuestos para la que está en vigor esta norma de cálculo.
- enddate es la hora en que esta norma de cálculo deja de estar en vigor.
- combination lo utiliza el método CalculationRuleCombineMethod para determinar cómo se puede combinar esta norma de cálculo con otras normas de cálculo. Consulte la tabla CALRULE para obtener más información.

- `identifier` identifica la norma de cálculo, en combinación con el código de cálculo de la misma.
  - `flags` especifica el distintivo de bit para indicar el proceso especial que deberá realizar la implementación por omisión de `CalculationRuleCombineMethod`. Consulte la tabla `CALRULE` en la ayuda en línea de WebSphere Commerce para obtener más información.
  - `calmethod_id` es el método `CalculationRuleCalculateMethod` que calcula un resultado monetario para un conjunto de artículos de pedido. Para determinar qué método de cálculo de norma de cálculo se debe utilizar, realice lo siguiente:
    - Consulte la tabla `CALMETHOD` en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna `SUBCLASS`. Pulse el enlace para la tabla `CALMETHOD: detalles`. Esta tabla lista los tipos de `CALMETHOD` disponibles. El método de cálculo de la norma de cálculo es 7.
    - Abra el archivo de rutina de carga `wcs.bootstrap_xx_XX.xml`, donde `xx_XX` es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:
      -  `unidad:\WebSphere\CommerceServer\schema\xml`
      -  `unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml`
      -  `/usr/WebSphere/CommerceServer/schema/xml`
      -  `/opt/WebSphere/CommerceServer/schema/xml`
      -  `/opt/WebSphere/CommerceServer/schema/xml`
      -  `/qibm/proddata/WebCommerce/schema/xml`
    - Localice la sección que lista los métodos de cálculo disponibles (`CALMETHOD`).
    - Localice los métodos de cálculo con el valor `calusage_ID` para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
    - Localice el método de cálculo que tiene la subclase de 7. Este método de cálculo es -47.
  - `calmethod_id_qfy` es el método `CalculationRuleQualifyMethod` que determina qué artículo de pedido de un conjunto de artículos de pedido debe enviarse al método `CalculationRuleCalculateMethod`. Utilice el método descrito en `calmethod_id` para determinar qué método de calificación de norma de cálculo se debe utilizar.
- b. Repita el paso a para cada norma de cálculo utilizada en la tienda. Tenga en cuenta que cada código de cálculo puede tener varias normas de cálculo, una para cada categoría de impuesto aplicable. Por ejemplo, `calcode_id="@calcode_id_1"` puede estar asociado a varios `calrule_id`.
9. Defina escalas de cálculo para la tienda.

Una escala de cálculo es el conjunto de rangos que se aplicarán al cálculo. Estos rangos crean una escala.

- a. Utilizando el ejemplo siguiente como guía, configure la escala de cálculo en el archivo XML para la tabla `CALSCALE`:







```
<calscale
 calscale_id="@calscale_id_19"
 code="Impuestos sobre la venta 1"
 storeent_id="@storeent_id_1"
```

```

calusage_id="-3"
setccurr="EUR"
calmethod_id="-53"
/>

```

donde

- calscale\_id es un identificador exclusivo generado.
- code es una serie de caracteres que identifica de forma exclusiva esta escala de cálculo, dado un uso de cálculo y una entidad de tienda determinados.
- storeent\_id es la entidad de tienda de la que forma parte esta escala de cálculo.
- calusage\_id indica la clase de cálculo para la que se utiliza esta escala de cálculo. Por ejemplo, la escala de cálculo puede utilizarse para calcular uno de los importes monetarios siguientes:
  - descuentos (-1)
  - costes de envío (-2)
  - impuestos sobre la venta (-3)
  - impuestos sobre el envío (-4)
  - cupones (-5)
- setccurr, si se especifica, indica la moneda para los valores de inicio de rango de los objetos de rango de cálculo para esta escala de cálculo. El método CalculationScaleLookupMethod debe devolver un "número de búsqueda" en esta moneda. En este caso, no se especifica; el método CalculationScaleLookupMethod devolverá un número de búsqueda en la moneda del pedido. No es necesario especificar la moneda a menos que los valores de inicio del rango de escala no sean cero.
- calmethod\_id es el método CalculationScaleLookupMethod que, dado un conjunto de artículos de pedido, determina un número de búsqueda, un importe base, un multiplicador de resultado y un conjunto de directrices matemáticas que la escala de cálculo puede utilizar para calcular un importe monetario. Para determinar qué método CalculationScaleLookupMethod se debe utilizar, realice lo siguiente:
  - Consulte la tabla CALMETHOD en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna SUBCLASS. Pulse el enlace para la tabla CALMETHOD: detalles. Esta tabla lista los tipos de CALMETHOD disponibles. El método MonetaryCalculationScaleLookupMethod es 9.
  - Abra el archivo de rutina de carga wcs.bootstrap\_xx\_XX.xml, donde xx\_XX es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:
    -  unidad:\WebSphere\CommerceServer\schema\xml
    -  unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml
    -  /usr/WebSphere/CommerceServer/schema/xml
    -  /opt/WebSphere/CommerceServer/schema/xml
    -  /opt/WebSphere/CommerceServer/schema/xml
    -  /qibm/proddata/WebCommerce/schema/xml



- Localice la sección que lista los métodos de cálculo disponibles (CALMETHOD).
- Localice los métodos de cálculo con el valor calusage\_ID para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
- Localice el método de cálculo que tiene la subclase de 9. Existen varios métodos de cálculo con la subclase de 9. Seleccione el que satisfaga sus necesidades.

Para obtener más información, consulte la tabla CALSCALE en la ayuda en línea de WebSphere Commerce.

- b. Repita el paso a para cada escala de cálculo utilizada en la tienda.
- c. Utilizando el ejemplo siguiente como guía, añada la información de descripción de escala de cálculo al archivo XML para la tabla CALSCALDS. Si está creando una tienda multicultural, deberá incluir esta información en un archivo XML específico de entorno nacional.

```
<calscaleds
 calscale_id="@calscale_id_19"
 description="5% de impuestos sobre la venta"
 language_id="&es_ES"
/>
```

donde

- calscale\_id es la escala de cálculo a la que se aplica la descripción.
- description es una descripción breve de la escala de cálculo, adecuada para describir a los clientes cómo se realiza un cálculo. Por ejemplo, "0,10 dólares por kilo, importe mínimo de 5,00 dólares" o "10% de descuento en la compra de 5 o más".
- language\_id es el idioma en que se visualiza esta información.

- d. Repita el paso c para cada escala de cálculo utilizada en la tienda.

#### 10. Defina rangos de cálculo para las escalas de cálculo.

- a. Utilizando el ejemplo siguiente como guía, configure el rango de cálculo en el archivo XML para la tabla CALRANGE.







```
<calrange
 calrange_id="@calrange_id_37"
 calscale_id="@calscale_id_19"
 calmethod_id="-59"
 rangestart="0.00000"
 cumulative="0"
/>
```

donde

- calrange\_id es un identificador exclusivo generado.
- calscale\_id es la escala de cálculo de la que forma parte este rango de cálculo.
- calmethod\_id es el método CalculationRangeMethod que determina un importe monetario a partir de CalculationRangeLookupResult. Por ejemplo, FixedAmountCalculationRangeCmd, PerUnitAmountCalculationRangeCmd o PercentageCalculationRangeCmd. Para determinar el método CalculationRangeMethod, realice lo siguiente:

- Consulte la tabla CALMETHOD en la ayuda en línea de WebSphere Commerce. Consulte la descripción de la columna SUBCLASS. Pulse

el enlace para la tabla CALMETHOD: detalles. Esta tabla lista los tipos de CALMETHOD disponibles. El método CalculationRangeMethod es 10.

- Abra el archivo de rutina de carga wcs.bootstrap\_xx\_XX.xml, donde xx\_XX es el código del entorno nacional. Los archivos de rutina de carga se encuentran en el directorio siguiente:
  -  NT unidad:\WebSphere\CommerceServer\schema\xml
  -  2000 unidad:\Archivos de programa\WebSphere\CommerceServer \schema\xml
  -  AIX /usr/WebSphere/CommerceServer/schema/xml
  -  Solaris /opt/WebSphere/CommerceServer/schema/xml
  -  Linux /opt/WebSphere/CommerceServer/schema/xml
  -  400 /qibm/proddata/WebCommerce/schema/xml
- Localice la sección que lista los métodos de cálculo disponibles (CALMETHOD).
- Localice los métodos de cálculo con el valor calusage\_ID para impuestos (-3 para impuestos sobre la venta y -4 para impuestos sobre el envío).
- Localice el método de cálculo cuya subclase es 10. Hay varios métodos de cálculo cuya subclase es 10. Seleccione uno que satisfaga sus necesidades.
- rangestart es un número de búsqueda mayor o igual que RANGESTART o si RANGESTART es NULL, esta fila coincide con el número de búsqueda.
- cumulative es lo siguiente:
  - 0 = sólo se utiliza el rango de cálculo coincidente que tiene el valor RANGESTART más alto.
  - 1 = se utilizan todos los rangos de cálculo coincidentes. Los importes monetarios calculados se suman para llegar al resultado final.

Para obtener más información, consulte la tabla CALRANGE en la ayuda en línea de WebSphere Commerce.

- b. Repita el paso a para cada rango de cálculo asociado a la escala de cálculo utilizada en la tienda. En el ejemplo anterior solamente hay un rango, ya que a todos los importes de los impuestos se les aplica el mismo tipo impositivo.
11. Defina los valores de búsqueda de cálculo para las escalas de cálculo. Los valores de búsqueda de cálculo son los valores asociados a la escala de cálculo. Por ejemplo, una escala de cálculo incluye los rangos siguientes y los tipos impositivos para los impuestos sobre ventas provinciales de Ontario en las comidas de un restaurante:
- el tipo impositivo de cantidades comprendidas entre 0,00 dólares y 3,99 dólares es del 0,00%
  - el tipo impositivo de cantidades de 4,00 y más es del 8,00%
- Los valores de búsqueda son de 0,00 y 8,00.
- a. Utilizando los ejemplos siguientes como guía, configure la búsqueda de cálculo en el archivo XML para la tabla CALRLOOKUP.

```
<calrlookup
 calrlookup_id="@calrlookup_id_37"
 calrange_id="@calrange_id_37"
 value="5.00"
/>
```

donde

- `calrlookup_id` es un identificador exclusivo generado.
- `calrange_id` es el rango de cálculo del que forma parte este resultado de búsqueda de rango de cálculo.
- `value` es el valor del resultado de búsqueda de rango de cálculo, utilizado por el método de rango de cálculo para determinar un resultado monetario. En este ejemplo, el tipo impositivo es de 5,00%.

Para obtener más información, consulte la tabla CALRLOOKUP en la ayuda en línea de WebSphere Commerce.

- Repita los pasos a y b para cada valor de búsqueda asociado a la escala de cálculo utilizada en la tienda. En este ejemplo, solamente hay un valor CALRLOOKUP, ya que CALRLOOKUP.SETCCURR es NULL, y solamente hay un CALRANGE ya que el tipo impositivo es el mismo para todos los importes.
12. Asocie la norma de cálculo y la escala de cálculo.

- Utilizando los ejemplos siguientes como guía, asocie la escala de cálculo a la norma de cálculo en el archivo XML para la tabla CRULESCALE.

```
<crulescale
 calrule_id="@calrule_id_10"
 calscale_id="@calscale_id_19"
/>
```

donde

- `calrule_id` es la norma de cálculo.
  - `calscale_id` es la escala de cálculo.
- Repita el paso a para cada escala de cálculo y asociación de norma. En el ejemplo que se ha utilizado anteriormente, solamente hay una escala de cálculo para cada norma de cálculo.

**Nota:** Si el tipo impositivo varía según la cantidad comprada, necesitará crear escalas cuyos valores de `rangestart` no sean cero. A continuación, tendrá que crear una escala de cálculo para cada moneda soportada (estableciendo `CALSCALE.SETCCURR` en la moneda correcta) para la que no ha establecido una tasa de conversión (consulte la tabla `CURCONVERT`) y asócielas todas con la norma de cálculo para esa categoría de impuestos determinada. Por ejemplo, en Ontario no hay ningún impuesto provincial sobre las ventas de comida cuyo importe esté por debajo de los 4,00 dólares. Si la tienda soporta la venta de comida en dólares americanos, tendrá que establecer una conversión de dólares americanos a dólares de Canadá, o crear una escala de cálculo de impuestos diferente con un valor `rangestart` adecuado, probablemente 6,00 dólares americanos, y asociarla con la misma norma de cálculo de impuestos. Solamente se utilizará la escala de cálculo adecuada, según la moneda del pedido.



Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, “Creación de los datos” en la página 327.

## Creación de elementos de despacho de impuestos







Para que los elementos de impuestos funcionen correctamente en la tienda, deberá asociar los grupos de jurisdicciones fiscales de la tienda al centro de despacho de pedidos utilizado por la tienda y, a continuación, asociar una norma de cálculo a ambos.

Deberá crear los elementos de despacho para poder asociar los elementos de impuestos a un centro de despacho de pedidos. Para obtener más información sobre cómo crear elementos de despacho, consulte “Creación de elementos de despacho de pedidos en WebSphere Commerce” en la página 114.

Después de haber creado los elementos de despacho, asocie a los mismos los elementos de impuestos, añadiendo información a la tabla TAXJCRULE. Efectúe lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, impuestos) asociados al producto o servicio que un cliente ha seleccionado para adquirir.
2. Revise los archivos XML utilizados para crear elementos de despacho de impuestos para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:






-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

Cada tienda de ejemplo incluye un archivo `taxfulfill.xml`, que contiene la información de impuestos. Para ver el archivo `taxfulfill.xml` del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo `taxfulfill.xml` está ubicado en el directorio de datos.

3. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
4. Cree un archivo `taxfulfill.xml`, copiando uno de los archivos `taxfulfill.xml` de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a `taxfulfill.xml`. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar

-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

5. Utilizando el ejemplo siguiente como guía, en el archivo XML añada información para la tabla TAXJCRULE:

```
<taxjcrule
taxjcrule_id="@taxjcrule_id_1"
calrule_id="@calrule_id_10"
ffmcenter_id="@ffmcenter_id_1"
jurstgroup_id="@jurstgroup_id_2"
precedence="0"
/>
```

donde

- taxjcrule\_id es el identificador exclusivo generado.
  - calrule\_id es la norma de cálculo utilizada.
  - ffmcenter\_id es el centro de despacho de pedidos. Si es NULL, esta asociación se aplica a todos los centros de despacho de pedidos.
  - jurstgroup\_id es el grupo de jurisdicciones fiscales. Si es NULL, esta asociación se aplica a todos los grupos de jurisdicciones fiscales.
  - precedence: cuando una dirección de envío está en más de uno de los grupos de jurisdicciones fiscales especificados, para el mismo centro de despacho de pedidos, sólo se habilita la norma de cálculo con el valor TAXJCRULE.PRECEDENCE más alto.
6. Repita el paso 3 para cada grupo de jurisdicciones, centro de despacho de pedidos y asociación de normas de la tienda.



Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, "Creación de los datos" en la página 327.

## Creación de elementos tienda-catálogo-impuestos

Para asociar los impuestos a las mercancías y los servicios de la tienda, deberá asociar un código de cálculo a las entradas de catálogo de la tienda para cada contrato que la tienda incluya.







Para poder crear elementos de tienda-catálogo-impuestos, deberá crear los elementos de tienda y de catálogo. Para obtener más información sobre cómo crear elementos de tienda, consulte "Creación de elementos de datos de tienda en un archivo XML" en la página 45. Para obtener más información sobre cómo crear elementos de catálogo, consulte "Visualización de los elementos de catálogo de la tienda" en la página 77.

Para crear elementos de tienda-catálogo-impuestos, realice lo siguiente:

1. Consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*. La infraestructura de cálculo de WebSphere Commerce calcula los importes monetarios (por ejemplo, envío) asociados al producto o servicio que un cliente ha seleccionado para adquirir.

2. Revise los archivos XML utilizados para crear elementos de tienda-catálogo-impuestos para las tiendas de ejemplo. Todos los archivos para las tiendas de ejemplo se encuentran en el archivo archivador de tienda correspondiente.







Los archivos de archivador de tienda están ubicados en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\samplstores
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  AIX /usr/WebSphere/CommerceServer/samplstores
-  Solaris /opt/WebSphere/CommerceServer/samplstores
-  Linux /opt/WebSphere/CommerceServer/samplstores
-  400 /qibm/proddata/WebCommerce/samplstores

Cada tienda de ejemplo incluye un archivo store-catalog-tax.xml, que incluye la información de despacho de envío. Para ver el archivo store-catalog-tax.xml del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo store-catalog-tax.xml está ubicado en el directorio de datos.

**Nota:** La ayuda en línea de WebSphere Commerce contiene información sobre cada uno de los elementos de datos contenidos en las tiendas de ejemplo.

3. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
4. Cree un archivo store-catalog-tax.xml, copiando uno de los archivos store-catalog-tax.xml de los archivadores de tienda de ejemplo o creando uno nuevo. Para obtener más información, consulte el archivo DTD que corresponda a store-catalog-tax.xml. Los archivos DTD se encuentran en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\xml\sar
-  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  AIX /usr/WebSphere/CommerceServer/xml/sar
-  Solaris /opt/WebSphere/CommerceServer/xml/sar
-  Linux /opt/WebSphere/CommerceServer/xml/sar
-  400 /qibm/proddata/WebCommerce/xml/sar

5. Cree la relación de tienda-catálogo-impuestos añadiendo información a la tabla CATENCALCD. Utilice el ejemplo siguiente como guía:

```
<catencalcd
 calcode_id="@calcode_id_3"
 catencalcd_id="@catencalcd_id_3"
 store_id="@storeent_id_1"
/>
```

donde

- calcode\_id es el código de cálculo.
- catencalcd\_id es un identificador exclusivo generado.
- store\_id es la tienda.



Para obtener más información sobre cómo utilizar @ y &, consulte el Apéndice B, "Creación de los datos" en la página 327.

---





---

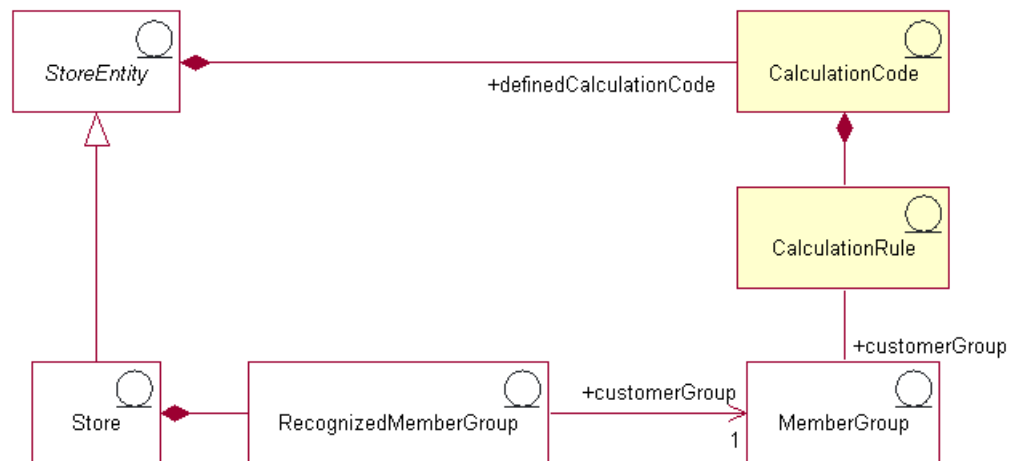
## Capítulo 20. Elementos de descuento

Los descuentos le permiten ofrecer a los clientes incentivos de precio para promocionar una compra. Puede ofrecer descuentos en forma de porcentaje (como, por ejemplo, un 10% menos) o en forma de cantidad fija (como, por ejemplo, 15 euros menos). Los descuentos pueden aplicarse a productos específicos o a la compra total. Por ejemplo, puede ofrecer una reducción del 20% a las personas mayores; o si tiene muchas gorras rojas en existencias, puede ofrecer un 25% de descuento en las gorras durante un tiempo limitado.

---

### Los descuentos en WebSphere Commerce

El diagrama siguiente ilustra la estructura de los descuentos en WebSphere Commerce Server.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

### Código de cálculo

Un *descuento* se representa y se calcula utilizando un *código de cálculo* de descuento. Un código de cálculo de descuento indica cómo se calcula el descuento para los artículos de pedido.

Un código de cálculo forma parte de una entidad de tienda. En una entidad de tienda se pueden definir varios códigos de cálculo. Si se suprime la entidad de tienda, también se suprimirán los códigos de cálculo definidos en dicha entidad de tienda.

Cada código de cálculo de descuento tiene una fecha de inicio y una fecha final, que definen el período de tiempo en que el descuento es efectivo. El código de

cálculo de descuento también se puede asociar con uno o varios grupos de miembros, que definen los grupos de miembros elegibles.

Un código de cálculo de descuento puede adjuntarse a una o más entradas de catálogo y grupos de catálogo. La acción de adjuntar un código de cálculo a un grupo de catálogo tiene el mismo efecto que si se adjunta a todas las entradas de catálogo directamente en el grupo de catálogo. Sin embargo, los códigos de cálculo de descuento asociados al grupo de catálogo A no estarán asociados a los productos y artículos del grupo de catálogo B si el grupo de catálogo A contiene el grupo de catálogo B.

Las entradas de catálogo o los grupos de catálogo pueden tener más de un descuento asociado a ellos. Cuando sea aplicable más de un código de cálculo de descuento a un pedido, los cálculos de descuento se realizarán en secuencia ascendente de sus atributos de secuencia de código de cálculo.

**Nota:** Defina el orden de la secuencia de descuento para implementar descuentos en los descuentos.

Los artículos de pedido se agrupan para el cálculo de uno de los modos siguientes:

- por acuerdo comercial
- por producto
- por oferta
- por dirección de envío

Para obtener más información, consulte la ayuda en línea de WebSphere Commerce.



Para obtener más información sobre el uso de códigos de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

---

### Normas de cálculo

Cada código de cálculo tiene un conjunto de normas de cálculo, que definen las condiciones bajo las cuales se realizará el cálculo. Cada norma de cálculo de descuento se asocia a uno o varios grupos de miembros, para el que está en vigor el descuento. Los grupos de miembros pueden tener derecho a obtener más de un descuento a la vez.

**Nota:** Si hay definido un grupo de miembros elegible en el nivel de código de cálculo, no será necesario volverlo a definir en el nivel de norma de cálculo.



Para obtener más información sobre el uso de normas de cálculo, consulte la publicación *IBM WebSphere Commerce, Guía de infraestructura de cálculo*.

---

---

## Creación de elementos de descuento en WebSphere Commerce

El método principal para crear descuentos en una tienda creada con WebSphere Commerce consiste en utilizar el asistente de descuentos de WebSphere Commerce Accelerator. Para obtener más información sobre cómo crear descuentos utilizando WebSphere Commerce Accelerator, consulte la ayuda en línea de WebSphere Commerce.

Los descuentos también se pueden crear utilizando un archivo XML y, a continuación, se pueden cargar mediante Loader package o publicar mediante los Servicios de tienda. Sin embargo, los descuentos creados de este modo, así como los descuentos importados durante la migración de versiones anteriores, funcionarán correctamente pero puede que no se visualicen del modo adecuado en WebSphere Commerce Accelerator.



---

## Capítulo 21. Elementos de inventario

El inventario incluye todo aquello que se puede contabilizar físicamente que se encuentra en un centro de despacho de pedidos. Se pueden proporcionar definiciones específicas de tipos de inventario, por ejemplo artículos, productos, códigos de artículo, paquetes compuestos y paquetes; pero todos ellos se consideran inventario. Los productos se configuran para el despacho de pedidos en el asistente Producto y el cuaderno Producto. Esto incluye opciones para hacer un seguimiento del inventario, permitir pedidos en espera, forzar pedidos en espera, despachar por separado y especificar que el producto no se debe devolver. WebSphere Commerce Accelerator hace una distinción entre los dos tipos principales de inventario que se puede recibir:

- El inventario esperado, que tiene asociado un registro de inventario esperado.
- El inventario ad hoc o inventario no registrado como esperado.

El inventario esperado se recibe de un proveedor y normalmente se paga con una orden de compra. WebSphere Commerce Accelerator hace un seguimiento del inventario esperado con los registros de inventario esperado y le permite registrar un identificador externo, normalmente un número de orden de compra de un sistema externo. De este modo, puede hacer fácilmente un seguimiento del inventario que ha solicitado, así como de lo que ha llegado y de lo que no. Los detalles del inventario esperado son los datos específicos acerca de los productos de un registro de inventario esperado, por ejemplo el centro de despacho de pedidos que espera el producto, la fecha de recepción esperada, la cantidad esperada y comentarios.

Un registro de inventario esperado no puede suprimirse una vez se ha recibido un inventario que se basa en él, y los detalles de un inventario esperado no pueden modificarse ni suprimirse en cuanto se ha recibido cualquier parte de ese inventario.

Cuando se formalizan pedidos del inventario que está disponible en un centro de despacho de pedidos, el sistema de pedidos asigna inventario a dichos pedidos. La asignación de inventario a un pedido deja el inventario no disponible para el sistema de pedidos. Si el pedido se cancela, el inventario vuelve a estar disponible. Si se formaliza un pedido de inventario que no está disponible, se puede crear un pedido en espera. Si hay inventario esperado que puede utilizarse para formalizar el pedido en espera, se asigna el inventario esperado al pedido en espera y se puede proporcionar al cliente la fecha de envío esperada.

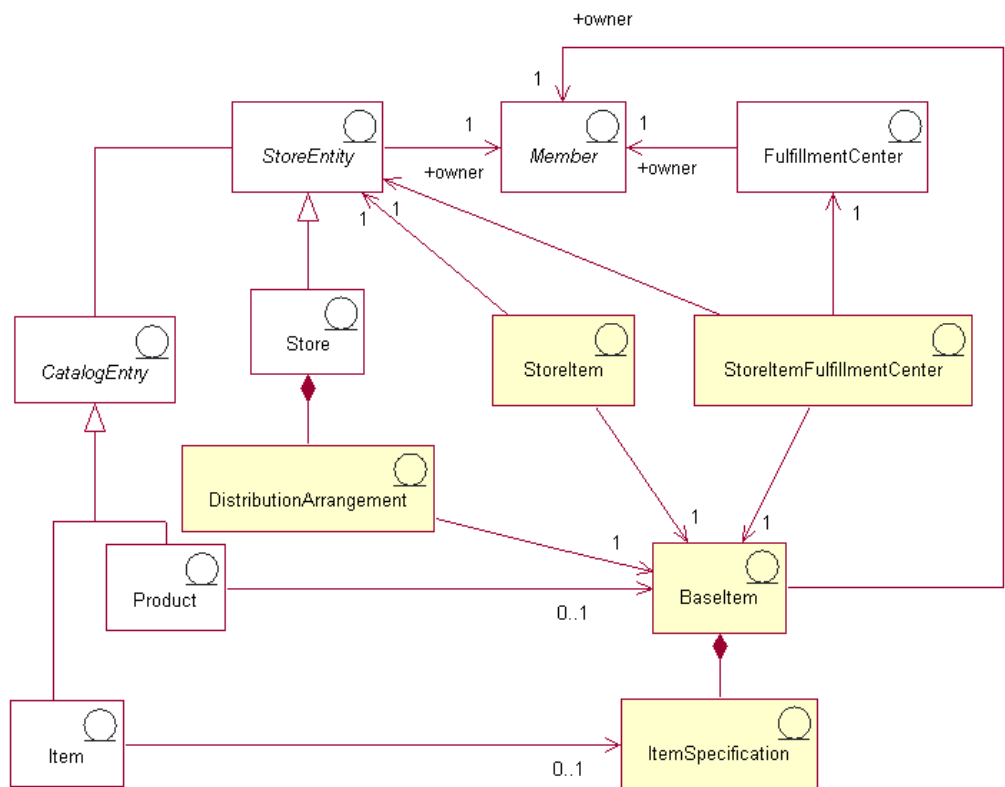
Las recepciones de inventario ad hoc se crean cuando llega inventario a un centro de despacho de pedidos sin un registro de inventario esperado correspondiente. Esto puede deberse a una llegada inesperada de inventario o puede que el comerciante o el vendedor haya elegido no utilizar registros de inventario esperado para registrar las recepciones de inventario.

**Nota:** Para que los productos puedan recibirse, deben existir en el sistema WebSphere Commerce, tanto si la recepción de inventario es esperada como si es ad hoc.

## Los elementos de inventario en WebSphere Commerce

Para comprender los elementos de inventario, es necesario comprender las relaciones entre el inventario y la tienda. Esto puede explicarse mediante el uso de un modelo de información. Las secciones siguientes describen las relaciones y las asociaciones que tiene el inventario con una tienda y otros elementos. Los diagramas proporcionados a continuación muestran las relaciones y las asociaciones para el inventario ATP y el inventario no ATP. El inventario no ATP constituye el modo en que las versiones anteriores del producto manejaban el inventario y aún se puede utilizar si una tienda elige no utilizar las características ATP. A continuación se describe cada diagrama y sus asociaciones. Para obtener más información sobre ATP, consulte la ayuda en línea.

### Inventario ATP



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

### Artículo base

El *artículo base* es el centro del diagrama de inventario y representa una familia general de mercancías con un nombre y una descripción comunes. Los artículos base se utilizan exclusivamente para el despacho y no son particulares de ninguna tienda. Cada entrada de catálogo que representa un producto del catálogo tiene un artículo base correspondiente para el despacho. Los artículos base se definen en la tabla BASEITEM.

### **Especificación de artículo**

Una *especificación de artículo* es un artículo base con valores definidos para todos sus atributos. Cada entrada de catálogo que representa un artículo en el catálogo tiene una especificación de artículo correspondiente para el despacho.

### **Entradas de catálogo**

Los productos y los artículos son *entradas de catálogo*. Las entradas de catálogo están asociadas a entidades de tienda, lo que significa que las entradas de catálogo, por ejemplo los productos y los artículos, se encuentran en las tiendas.

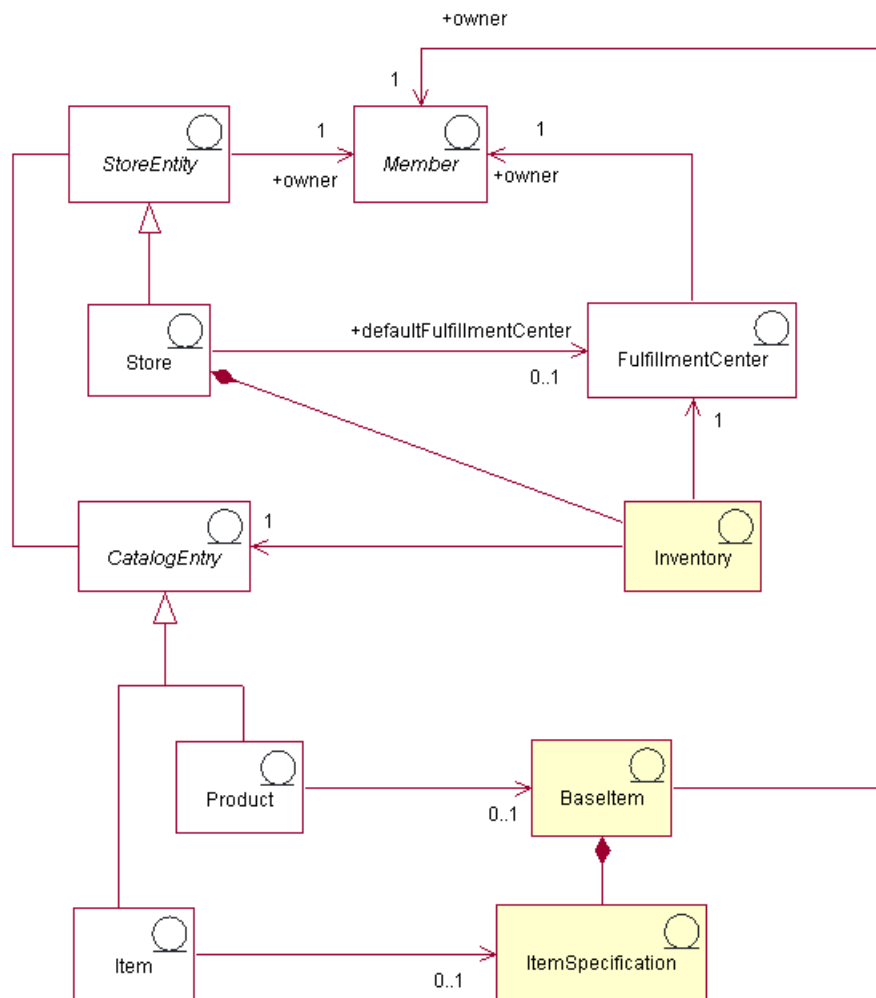
### **Plan de distribución**

Un *plan de distribución* está asociado a un artículo base, lo que permite que una tienda venda su propio inventario. Los planes de distribución se almacenan en la tabla DISTARRANG.

### **Artículo de tienda**

Un *artículo de tienda* representa atributos que afectan el modo en que una tienda o un grupo de tiendas determinado asigna inventario para los artículos especificados de un artículo base determinado, incluido si se debe permitir poner pedidos en espera y hacer un seguimiento del inventario. La tabla STORITMFFC define una estimación del número de segundos que se tardan desde el momento en que un artículo de pedido se entrega para el despacho hasta que se envía al cliente. Esta tabla sólo se llena de datos si una tienda desea definir un cambio en el margen de tiempo para el envío por omisión de FFMCENTER, para un artículo de tienda.

## Inventario no ATP



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

El artículo base también es el centro del diagrama de inventario no ATP. La relación del artículo base con los productos, artículos y entradas de catálogo, es la misma que en el diagrama de inventario general. Un artículo base sigue siendo propiedad de un miembro y, una vez definido por dicho miembro, puede venderse en la tienda. Sin embargo, en este caso, no hay plan de distribución, ni asociación de artículo de tienda ni centro de despacho de pedidos de artículos de tienda.

### Centro de despacho de pedidos

El inventario está asociado a un *centro de despacho de pedidos* y una tienda. Una tienda puede designar un centro de despacho de pedidos por omisión. Al igual que los artículos base, los centros de despacho de pedidos son propiedad de los miembros y éstos comparten dicha propiedad con la tienda. Para obtener más información sobre los elementos de despacho, consulte el Capítulo 11, “Elementos



de despacho de pedidos” en la página 111.



---

Para obtener información más detallada sobre la estructura de los elementos de inventario en WebSphere Commerce Server, consulte los modelos de datos y de objeto de inventario en la ayuda en línea de WebSphere Commerce.

---

---

## Creación de elementos de inventario en WebSphere Commerce

Puesto que el inventario son datos operativos, cambia a diario a medida que los clientes compran productos en la tienda o devuelven artículos a la misma. Como resultado, los niveles de inventario suben y bajan a medida que se venden productos y a medida que los centros de despacho de pedidos reciben inventario nuevo de los proveedores. WebSphere Commerce Accelerator le permite realizar las siguientes tareas relacionadas con el inventario:

- registrar inventario esperado
- recibir inventario esperado y ad hoc de los proveedores
- ajustar inventario
- mantener registros de devoluciones
- mantener razones de devolución
- recibir inventario devuelto por los clientes
- gestionar la disposición del inventario devuelto

Para obtener más información sobre cómo utilizar WebSphere Commerce Accelerator para gestionar inventario, consulte la ayuda en línea de WebSphere Commerce.



---

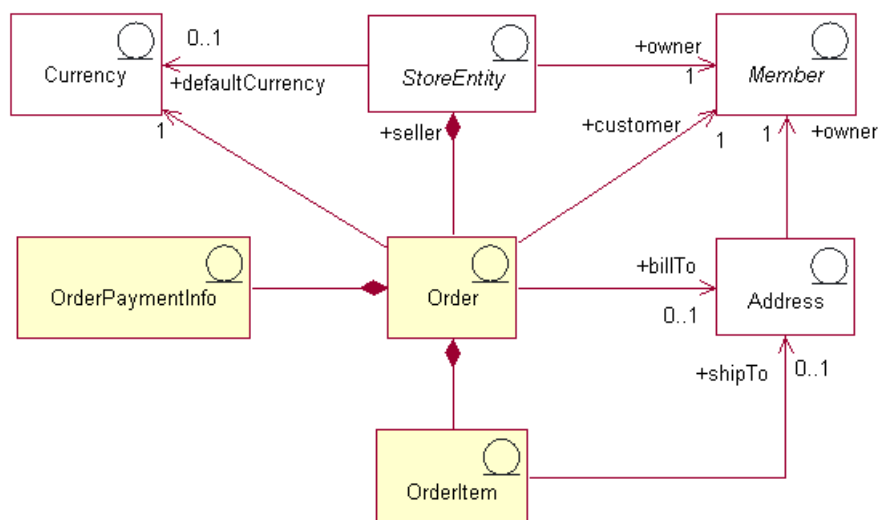
## Capítulo 22. Elementos de pedido

Los elementos de pedido en el sistema WebSphere Commerce proporcionan funciones de carro de la compra, gestión de pedidos y proceso de pedidos. Las posibilidades de proceso de pedidos incluyen la compra o el pedido rápido, los pedidos planificados, los pedidos múltiples pendientes, la repetición de pedidos, pedidos divididos y pedidos en espera. Los servicios relacionados, por ejemplo la fijación de precios, los impuestos, el pago, el inventario y el despacho, también forman parte de los elementos de pedido.

---

### Los elementos de pedido en WebSphere Commerce

El diagrama siguiente ilustra los elementos de pedido en WebSphere Commerce Server. A continuación del diagrama, se proporcionan descripciones de cada elemento.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte "Modelo de información de los datos de la tienda" en la página 27.

### Pedidos y artículos de pedido

En el sistema WebSphere Commerce, para un cliente o un comprador, un *pedido* es una lista de productos seleccionados (por ejemplo, un pedido puede contener dos libros y un CD) y cada producto de esa lista es un *artículo de pedido* (por ejemplo, cada libro y CD es un artículo del mismo pedido). Cuando un cliente formaliza un pedido con la tienda, el cliente debe proporcionar una dirección de facturación a la que la tienda envía la factura. Con cada pedido se asocia un solo identificador de moneda. Desde la perspectiva de una tienda, un pedido es una lista de artículos de pedido. Forma parte de los datos de la tienda.

#### Moneda

Una tienda puede mostrar los precios en una *moneda* o utilizar varias monedas. Cada tienda debe definir también una *moneda por omisión*. También puede permitir

que los clientes seleccionen una *moneda de compra*. Si la moneda de compra es la misma que la moneda por omisión de la tienda, dicha moneda ya está soportada en la tabla STOREENT. Si la moneda de compra no es la moneda por omisión de la tienda, deberá añadir dicha moneda a la tabla CURLIST. Los clientes utilizan la moneda de compra para formalizar pedidos en la tienda.

### **Información de pago**

Una vez que un cliente ha seleccionado una moneda de compra preferida, todos los pagos se procesarán en esa moneda. En función de las políticas y del soporte de pago de la tienda, los clientes pueden pagar las compras utilizando el pago en línea (en el que un cliente proporciona información de pago a través de Internet en el sitio de la tienda) o el pago fuera de línea (en el que el cliente proporciona información de pago sin utilizar el canal de Internet, por ejemplo por teléfono o fax). Independientemente de los métodos de pago en línea o fuera de línea, al formalizar los pedidos, los clientes deben proporcionar *información de pago*, que incluye cualquiera de los elementos siguientes:

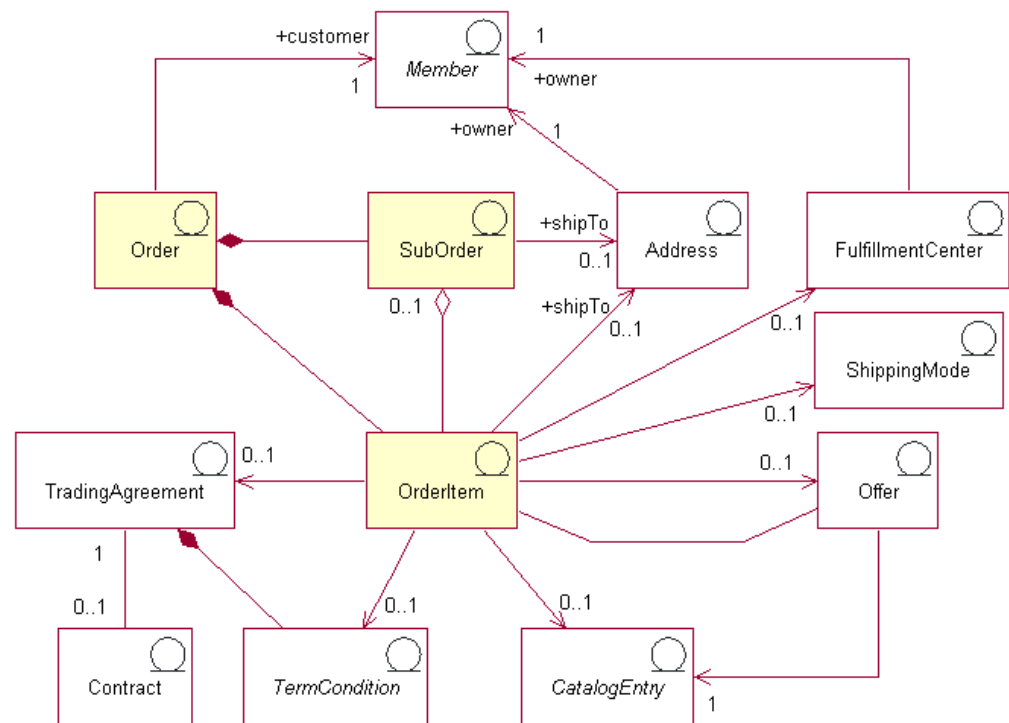
- Método de pago: El método de pago del cliente para el pedido. En función de los casetes de pago configurados en Payment Manager para la tienda, puede configurar la tienda para aceptar el pago fuera de línea, CyberCash para los pagos en línea, SET Secure Electronic Transaction y Merchant Initiated Authorization (MIA) para los pagos en línea que no requieren que los clientes utilicen un monedero en línea o un método de pago personalizado.
- Para los pagos con tarjeta de crédito, información acerca de la tarjeta: El tipo de tarjeta, el número y la fecha de caducidad de la tarjeta de crédito del cliente utilizada para pagar el pedido. Normalmente, la información de la tarjeta de crédito es necesaria si la tienda soporta el pago en línea.
- Número de orden de compra: El número de orden de compra, que el cliente puede haber proporcionado al realizar el pedido en la tienda. El número de orden de compra autentica al cliente como cliente autorizado para realizar pedidos en la tienda, según lo estipulado en los términos del contrato entre la tienda y el cliente.

### **Artículos de pedido**

Los *artículos de pedido* son los productos o artículos individuales que forman un pedido. Un pedido debe tener como mínimo un artículo de pedido. Cada artículo de pedido representa algo que un cliente ha seleccionado para comprar. Además, cada artículo de pedido tiene una referencia a un acuerdo de comercio (generalmente un contrato), una modalidad de envío, un centro de despacho de pedidos y cualquier oferta de precio especial. Los descuentos, los costes de envío y el total de impuestos se almacenan con cada artículo de pedido.

El diagrama siguiente ilustra los elementos de artículo de pedido de WebSphere Commerce. A continuación del diagrama, se proporcionan descripciones de cada

elemento.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información de la tienda, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

## Subpedidos

Los artículos de pedido se agrupan para formar *subpedidos*. Un subpedido es la parte de un pedido que se está enviando a una dirección específica. Por ejemplo, un cliente puede indicar direcciones de envío diferentes para diferentes productos del carro de la compra. Cada dirección de envío y los productos asociados a la misma constituyen un subpedido. Los artículos de pedido de un subpedido tienen la misma dirección de envío y pueden utilizarse para visualizar los subtotales de los importes de los artículos de pedido.

El atributo de cantidad del objeto Artículo de pedido es un número sin unidad que se puede multiplicar por el atributo de cantidad nominal del objeto Información de envío de entrada de catálogo asociado al objeto Entrada de catálogo para llegar a la cantidad real representada por el Artículo de pedido. El objeto de información de envío de entrada de catálogo especifica la unidad de medida en la que se indican las cantidades.

Aunque los pedidos se asocian normalmente a una sola tienda, existe un tipo especial de pedido que se puede asociar a una tienda o un grupo de tiendas que es el perfil de pedido. El perfil de pedido se representa en el modelo de objeto como un Pedido con el estado de ‘Q’. El perfil de pedido contiene información por omisión acerca de un cliente, por ejemplo información de pago, dirección de envío, modalidad de envío y dirección de facturación.

## Otros elementos de artículo de pedido

Un artículo de pedido puede asociarse, si lo desea, a una opción de cada uno de los objetos siguientes.

- Una dirección de envío para el cliente que ha formalizado el pedido que contiene el artículo de pedido. Un cliente debe especificar una dirección de envío durante el proceso de pedido, para que el centro de despacho de pedidos de la tienda pueda utilizar esta dirección para enviar el artículo de pedido en consecuencia.
- Un centro de despacho de pedidos para enviar y recibir los artículos de pedido necesarios para los pedidos de cliente y para almacenar inventario para el artículo de pedido.
- Una modalidad de envío para el artículo de pedido, que es una combinación de una empresa de transportes (una empresa que proporciona servicios de envío desde un centro de despacho de pedidos a un cliente) y el servicio de envío ofrecido por dicha empresa de transportes. Por ejemplo, Compañía de transportes ABC, Servicio 12 horas y Compañía de transportes ABC, Entrega urgente son modalidades de envío.
- Una oferta de precio asociada al artículo de pedido. Al incluir diferentes ofertas en listas de precios diferentes (o "contenedores de propuestas de comercio"), las tiendas pueden presentar precios diferentes para el mismo producto o código de artículo a clientes diferentes. Por ejemplo, una agencia de viajes puede ofrecer billetes de avión en cuatro listas de precios diferentes: precios para adultos, precios para personas mayores, precios para niños y precios para estudiantes.
- Una entrada de catálogo para el artículo de pedido; es decir, cada artículo de pedido solicita un artículo de un catálogo.
- Un acuerdo de comercio que define los términos y condiciones bajo los cuales se realiza el pedido del artículo. Normalmente esto es un contrato, pero puede ser una solicitud de presupuesto (RFQ), que representa una negociación, hasta que el pedido se ha sometido a proceso.



Para obtener información más detallada sobre la estructura de los elementos de pedido en WebSphere Commerce Server, consulte los modelos de datos y de objeto de pedido en la ayuda en línea de WebSphere Commerce.

---

## Creación de elementos de pedido en WebSphere Commerce

Un cliente puede formalizar pedidos en una tienda o solicitar que un Representante de servicio al cliente de la tienda le ayude a realizar esta tarea (utilizando WebSphere Commerce Accelerator). Para crear un pedido en nombre de un cliente de empresa a consumidor, consulte los temas de la ayuda en línea de WebSphere Commerce "Crear un pedido para un cliente registrado" y "Crear un pedido para un cliente no registrado". Para crear un pedido en nombre de un cliente de empresa a empresa, consulte el tema de ayuda "Crear un pedido para un usuario de empresa".

---

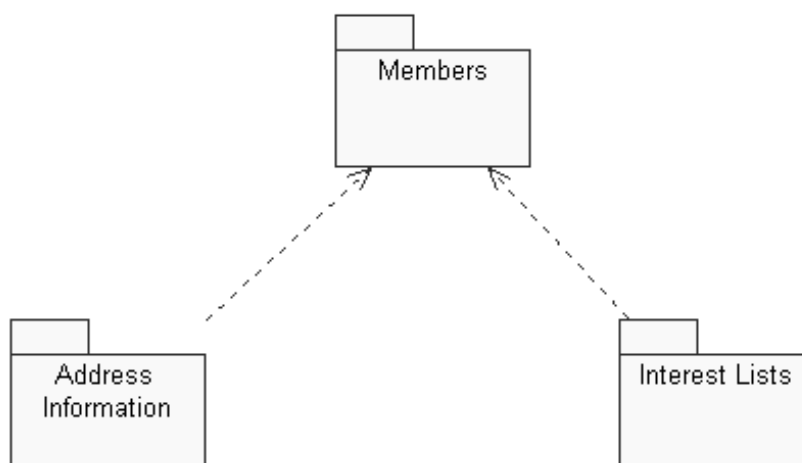
## Capítulo 23. Elementos de cliente y vendedor

Aunque una tienda puede incluir varios participantes de las actividades de la tienda, como mínimo debe tener un cliente y un vendedor.

---

### Los elementos de cliente en WebSphere Commerce

Un *cliente* es la persona que compra en la tienda, crea una lista de intereses si lo desea, formaliza pedidos y realiza compras en la tienda o al vendedor. El diagrama siguiente ilustra los elementos que un cliente necesita para formalizar un pedido en una tienda.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

---

Como se muestra en el diagrama anterior, el sistema WebSphere Commerce contiene miembros. A cada miembro que sea un usuario o una entidad de organización se le puede asignar un rol.

**Nota:** En WebSphere Commerce, un *miembro* puede ser una entidad de organización, un usuario o un grupo de miembros. Consulte el apartado “Miembros” en la página 195 para obtener más información.

En este caso, el miembro usuario es un cliente. Un cliente debe proporcionar información de dirección y puede tener una lista de artículos de interés. El diagrama ilustra la relación recíproca entre un miembro (cliente) y los elementos de cliente asociados a él: un cliente debe proporcionar una dirección de la cual

debe ser propietario y puede tener una lista de intereses para comprar en una tienda; la dirección y la lista de intereses dependen de la existencia de un cliente.

## Información de dirección

Un cliente debe proporcionar tres tipos de información de dirección cuando realice compras en una tienda: la dirección de la persona de contacto, la dirección de facturación y la dirección de envío. A continuación, se describen estos tipos de direcciones; cada dirección puede ser exclusiva o puede ser la misma:

- La dirección de la persona de contacto se utiliza para fines diversos; por ejemplo, en relación al estado o a los cambios de un pedido y para avisar sobre los próximos eventos de la tienda (promociones o mantenimiento de la tienda). La dirección de la persona de contacto del cliente incluye el nombre y el número de la calle, la ciudad, el estado o la provincia, el código postal, el país o la región, la dirección de correo electrónico, el número de teléfono y el número de fax. Normalmente, la dirección de la persona de contacto es el lugar donde se puede localizar al cliente más fácilmente, por ejemplo la dirección del trabajo.
- La dirección de facturación se utiliza para enviar un recibo o factura de las compras realizadas. Una dirección de facturación incluye el nombre y el número de la calle, la ciudad, el estado o la provincia, el código postal, el país o la región, el número de teléfono y la dirección de correo electrónico. La dirección de facturación puede ser la misma que la dirección de la persona de contacto o la dirección de envío, o puede ser una dirección distinta.
- La dirección de envío se utiliza para la entrega de los artículos adquiridos. Una dirección de envío incluye el nombre y el número de la calle, la ciudad, el estado o la provincia, el código postal, el país o la región, el número de teléfono y la dirección de correo electrónico. La dirección de envío puede ser la misma que la dirección de la persona de contacto o la dirección de facturación, o puede ser una dirección distinta.

## Listas de intereses

Las tiendas pueden soportar *listas de intereses*. Es decir, los clientes añaden a sus listas de intereses productos que desean solicitar en el futuro. Una lista de intereses no es un carro de la compra; una lista de intereses puede contener artículos de varias tiendas y no contiene los precios, las direcciones de envío, las modalidades de envío, la disponibilidad de inventario ni los importes calculados como, por ejemplo, los descuentos, los costes de envío y los impuestos.

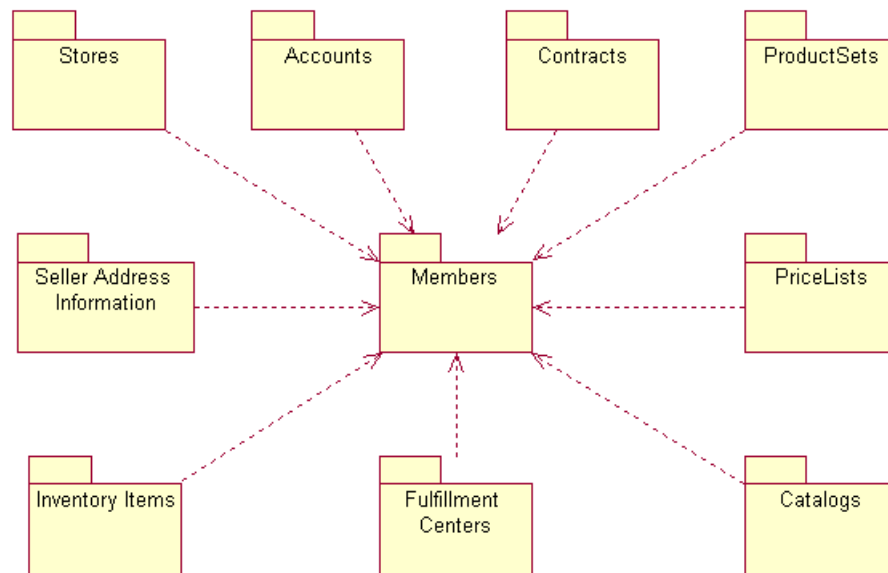
---

## Descripción de los elementos del vendedor en WebSphere Commerce

El *vendedor* supervisa la gestión y los objetivos generales de la tienda, además de hacer el seguimiento de las ventas de la tienda. El vendedor vende los artículos y los servicios al cliente. El rol de vendedor es equivalente al de comerciante y tiene acceso a todas las funciones de WebSphere Commerce Accelerator. El diagrama siguiente ilustra los elementos que un vendedor necesita para mantener una tienda



y vender a los clientes.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

Como se muestra en el diagrama anterior, el sistema WebSphere Commerce contiene miembros. Cada miembro tiene asignado un rol, por ejemplo un Representante de servicio al cliente para la tienda o un receptor de inventario en un almacén. El rol de vendedor puede mantener los elementos siguientes para poder vender a los clientes:

- Tiendas
- Cuentas (opcional)
- Contratos (o al menos el contrato por omisión de WebSphere Commerce)
- Conjuntos de productos
- Listas de precios
- Catálogos
- Centros de despacho de pedidos
- Artículos de inventario

El diagrama anterior ilustra la relación entre un miembro (vendedor) y los elementos de vendedor asociados a él; es decir, un vendedor puede tener los elementos listados anteriormente para mantener una tienda y los elementos necesitan tener un vendedor que los despliegue.

## Tiendas

Una *tienda* en línea de WebSphere Commerce está compuesta por un conjunto de archivos HTML y JavaServer Pages, así como de elementos de impuestos, envío, pago, catálogo y otros elementos de base de datos, que están contenidos en un

archivador de tienda. Una tienda también contiene datos de tienda, que es la información con la que se llena la base de datos de WebSphere Commerce para permitir que una tienda funcione.

Para obtener más información sobre las tiendas de WebSphere Commerce, consulte el Capítulo 6, “Elementos de tienda” en la página 43 y la Parte 4, “Desarrollo de los datos de la tienda” en la página 37.

## Cuentas

Una tienda puede configurar *cuentas* de negocio para los clientes a fin de permitirles comprar en la tienda. Una cuenta contiene la información siguiente:

- El nombre de la cuenta, que es normalmente el nombre de la organización a la que está asociado el cliente. Esta organización tiene contratos definidos con la tienda, que estipulan los términos para que el cliente compre en la tienda. Por ejemplo, la organización IBM puede tener contratos con la Compañía de suministros de oficina ABC.
- El nombre del representante, que es el nombre de la organización representante dentro de la organización del vendedor que es responsable de la cuenta.
- El número de contratos que pertenecen a la cuenta.

Para obtener más información sobre las cuentas de WebSphere Commerce, consulte el apartado “Cuenta (cuenta de negocio)” en la página 100 y la ayuda en línea de WebSphere Commerce.

## Contratos

Normalmente, en WebSphere Commerce todos los clientes deben comprar bajo un *contrato*. Cada cuenta entre el cliente y el vendedor debe estar asociada a uno o más contratos (o al menos un *contrato por omisión* para que los clientes, registrados o no, compren en la tienda o si se desea que los clientes puedan comprar productos no cubiertos por otros contratos). Un contrato permite al cliente comprar productos en una tienda a un precio especificado durante un periodo de tiempo especificado, bajo los términos y condiciones y las políticas de negocio que se estipulan en el contrato. El vendedor despliega el contrato para que los clientes puedan comprar en la tienda.

Para obtener más información sobre los contratos de WebSphere Commerce y el contrato por omisión que un vendedor puede utilizar, consulte el apartado “Contrato” en la página 101.

## Conjuntos de productos

Los *conjuntos de productos* proporcionan un mecanismo para que un vendedor clasifique los catálogos en línea en categorías de subconjuntos lógicos a fin de que un vendedor pueda permitir que diversos clientes se beneficien de diferentes vistas de catálogo. Además, un vendedor puede crear un contrato para un cliente y estipular que el cliente sólo pueda comprar productos bajo un conjunto de productos predefinido.

Para obtener más información sobre los conjuntos de productos de WebSphere Commerce, consulte “Conjuntos de productos” en la página 58.

## Listas de precios

Una *lista de precios* está asociada al precio que un vendedor ofrece o presenta a un cliente. Un vendedor puede listar precios diferentes para el mismo producto para

clientes diferentes. En WebSphere Commerce, una oferta de precio se conoce también como *propuesta de comercio* y representa el precio de una entrada de catálogo y los criterios que el cliente debe satisfacer para poder pagar dicho precio.

En WebSphere Commerce, un objeto Oferta forma parte de un contenedor de propuestas de comercio, que es propiedad de un miembro. Un contenedor de propuestas de comercio contiene propuestas de comercio y puede dejarse disponible para todos los clientes o sólo para clientes de determinados grupos mediante los acuerdos de comercio o los contratos. A veces un contenedor de propuestas de comercio (TradingPositionContainer) se denomina lista de precios. Existen dos clases de listas de precios: una lista de precios estándar que contiene los precios base para los productos del catálogo de tienda o una lista de precios personalizada que especifica la lista de productos y sus precios personalizados.

Para obtener más información sobre las listas de precios de WebSphere Commerce, consulte el Capítulo 9, "Elementos de cálculo de precios" en la página 91.

## Catálogos

Una tienda de WebSphere Commerce utiliza como mínimo un *catálogo* en línea para exponer las mercancías y los servicios que el vendedor tiene en venta. Normalmente un catálogo en línea contiene precios, imágenes y descripciones de los artículos en venta. Un catálogo en línea también puede presentar mercancías en categorías diferentes que faciliten la navegación.

Cada tienda del sistema WebSphere Commerce debe tener un *catálogo maestro*, que se utiliza para la gestión de catálogos. El catálogo maestro es la ubicación central para gestionar la mercancía de un vendedor; es el catálogo individual que contiene todos los productos, artículos, relaciones y precios estándar para todo lo que está en venta en la tienda. Si un vendedor tiene más de una tienda, el catálogo maestro puede compartirse entre estas tiendas.

Para obtener más información sobre los conjuntos de productos de WebSphere Commerce, consulte el Capítulo 8, "Elementos de catálogo" en la página 55.

## Centros de despacho de pedidos

Los *centros de despacho de pedidos* los utilizan las tiendas como almacenes de inventario y como centros de envío y recepción. Un vendedor puede tener uno o muchos centros de despacho de pedidos asociados.

Desde la perspectiva de un servidor WebSphere Commerce, un objeto Centro de despacho de pedidos es independiente del objeto Tienda. Gestiona el inventario y el envío de los productos. Para enviar un pedido, el centro de despacho de pedidos se basa en el objeto Modalidad de envío especificado por el cliente. El objeto Modalidad de envío indica la empresa de transportes y el método de envío para despachar los pedidos. En un centro de despacho de pedidos, el objeto Acuerdo de envío indica que un objeto Tienda ha dispuesto con un objeto Centro de despacho de pedidos enviar los productos utilizando una determinada modalidad de envío.

Para obtener más información sobre los centros de despacho de pedidos de WebSphere Commerce, consulte el Capítulo 11, "Elementos de despacho de pedidos" en la página 111 y el Capítulo 18, "Elementos de envío" en la página 137.

## Artículos de inventario

Los *artículos de inventario* incluyen todo aquello que se puede contabilizar físicamente en el centro de despacho de pedidos de un vendedor. El sistema WebSphere Commerce define tipos específicos de inventario que se pueden despachar, por ejemplo artículos, productos, códigos de artículo, paquetes compuestos y paquetes; pero todos ellos se consideran inventario. Los productos se configuran para el despacho en las herramientas de Gestión de productos en WebSphere Commerce Accelerator.

Para obtener más información acerca de los artículos de inventario de WebSphere Commerce, consulte la ayuda en línea de WebSphere Commerce y el Capítulo 21, “Elementos de inventario” en la página 179.

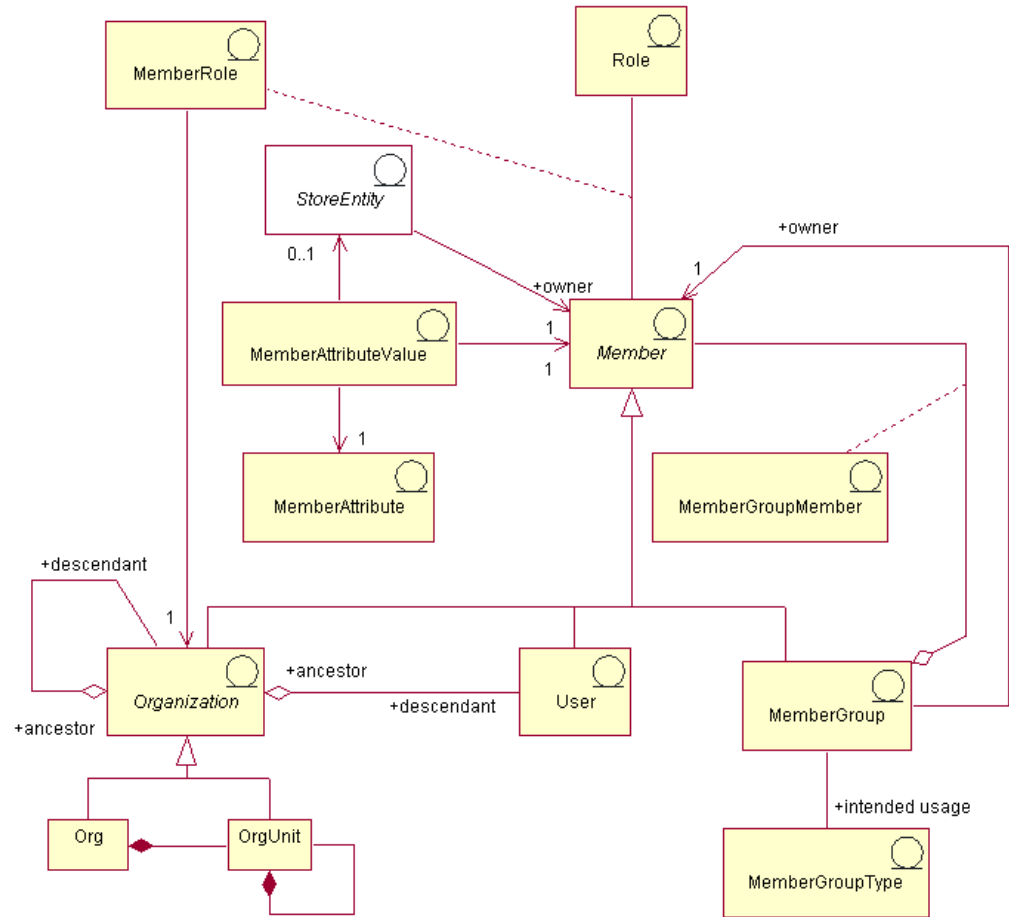
---

## Los elementos de miembro en WebSphere Commerce

Los elementos de miembro de WebSphere Commerce incluyen datos para los participantes del sistema WebSphere Commerce. Un miembro puede ser un usuario, un grupo de usuarios o una entidad de organización. Un administrador, por ejemplo un Administrador de sitio, asigna roles a los usuarios y a los miembros de la entidad de organización. Una vez que se ha asignado un rol a un miembro, el componente de control de acceso autoriza al miembro a participar en las actividades asociadas al rol. Por ejemplo, una organización puede ser un comprador, un vendedor o ambos. A un usuario también se le pueden asignar varios roles. Un administrador puede crear grupos de miembros, que son grupos de usuarios clasificados en categorías por diversas razones del negocio. Utilice la Consola de administración de WebSphere Commerce para crear organizaciones, usuarios, roles y grupos de miembros, y trabajar con ellos.

La lógica de negocio para los elementos de miembro proporciona servicios de gestión de perfiles y de registro de miembros. Otros servicios que están estrechamente relacionados con los elementos de miembros incluyen el control de acceso, la autenticación y la gestión de sesiones. Para obtener más detalles sobre estos temas, consulte la ayuda en línea de WebSphere Commerce.

El diagrama siguiente ilustra los elementos de miembro de WebSphere Commerce. A continuación del diagrama, se proporcionan descripciones de cada elemento.



Este diagrama, así como todos los demás que se encuentran en la sección de datos de la tienda, forma parte del modelo de información de WebSphere Commerce Server. Para obtener más información sobre el modelo de información, consulte “Modelo de información de los datos de la tienda” en la página 27. Para obtener información sobre los convenios utilizados en este diagrama, consulte el Apéndice A, “Descripción de UML” en la página 325.

## Miembros

Un *miembro* en WebSphere Commerce puede ser cualquiera de los siguientes:

- Una *entidad de organización*. Puede ser una organización, por ejemplo “IBM”, o una unidad de organización dentro de una gran organización, por ejemplo la “División de Comercio electrónico” dentro de IBM.
- Un *usuario* (registrado o no registrado). Un usuario registrado tiene un identificador exclusivo y una contraseña que son necesarios para proporcionar los datos de perfil para el registro. Los usuarios registrados pueden clasificarse de acuerdo con el tipo de perfil: el tipo ‘B’ indica un usuario de empresa (o un cliente de empresa a empresa) y el tipo ‘C’ indica un usuario minorista (o un cliente de empresa a consumidor). Para obtener más información sobre usuarios registrados y no registrados, consulte “Miembros” en la ayuda en línea de WebSphere Commerce.

- Un *grupo de miembros*. Se trata de un grupo de usuarios clasificados por categorías por diversas razones del negocio. Las agrupaciones pueden utilizarse para el control de acceso, para las aprobaciones así como para el marketing (por ejemplo calcular descuentos y precios, y visualizar productos).

Cada *entidad de tienda* (es decir, una tienda o un grupo de tiendas) es propiedad de un miembro.

---

## Atributos de miembro

Un miembro de WebSphere Commerce tiene un conjunto de *atributos* y cada atributo tiene un *valor* asociado a él. Un perfil de usuario básico para un miembro incorpora información de registro, datos estadísticos, información de dirección, historial de compras y otros atributos diversos.

Un perfil de usuario de empresa contiene la misma información que un perfil de usuario básico, así como información de empleo, por ejemplo un número de empleado, un puesto de trabajo o una descripción de trabajo. Durante el registro, los usuarios de empresa deben identificar la organización de negocio a la que pertenecen. Los perfiles para las entidades de organización incluyen esta información adicional, por ejemplo el nombre y categoría de negocio de la organización.

Las normas de control de acceso imponen la autorización de usuario para la gestión de perfiles. Los perfiles de miembros pueden contener diversos atributos relacionados con el negocio y personales (por ejemplo, roles, información de pago, direcciones, monedas e idiomas preferidos y dispositivos de tipo "pervasive computing"). Los atributos pueden ser sensibles a la tienda. Estos atributos no se soportan para grupos de miembros.

---

## Roles

Cada miembro usuario puede desempeñar uno o más roles para una organización. Un Administrador de sitio asigna un rol o roles a cada miembro de usuario. Por ejemplo, como miembro de la organización IBM, el rol de Juan Pérez como Representante de servicio al cliente significa que Juan realiza tareas para los clientes de IBM y les ayuda con las consultas y los asuntos relacionados con la información de registro, los pedidos o las devoluciones. Puede que Juan también tenga el rol de Supervisor de servicio al cliente, que tiene todas las responsabilidades de las tareas descritas anteriormente, así como la autorización de aprobaciones y de supervisión de otros Representantes de servicio al cliente.

El sistema WebSphere Commerce proporciona el siguiente conjunto de tipos de roles por omisión:

- Roles de desarrollo de sitio y contenido
- Roles de operaciones técnicas
- Roles de gestión de marketing
- Roles de gestión de productos
- Roles de gestión de relaciones comerciales
- Roles de gestión de logística y operaciones
- Roles de gestión de organización

Para obtener detalles sobre cada uno de estos roles, consulte en la ayuda en línea de WebSphere Commerce el tema "Roles". Un Administrador de sitio puede asignar estos roles, así como cualquier rol nuevo creado por el Administrador de

sitio, por entidad de organización; es decir, los usuarios que pertenecen a una entidad de organización pueden asumir los roles asignados a dicha entidad de organización.

Cuando se asigna un rol a un usuario, el rol se establece en el ámbito de la entidad de organización. Cuando se asigna un rol a un usuario, éste no desempeña necesariamente dicho rol para la entidad de organización a la que pertenece; es decir, cuando un administrador realiza la asignación, el administrador puede seleccionar para qué entidad de organización desempeña dicho rol el usuario. Si el administrador selecciona la Organización raíz, el usuario desempeña ese rol para todas las entidades de organización.



Para obtener información más detallada sobre la estructura de los elementos de miembro en WebSphere Commerce, consulte los modelos de datos y de objeto de miembro en la ayuda en línea de WebSphere Commerce.

---

---

## Creación de elementos de miembro en WebSphere Commerce

Para crear un vendedor (una organización que actúa como propietaria de la tienda) y mantener información acerca del vendedor, utilice la Consola de administración. Para obtener más información, consulte el tema "Crear una organización" de la ayuda en línea de WebSphere Commerce.

El Desarrollador de tiendas no crea un cliente: cuando un cliente se registra en una tienda, el sistema WebSphere Commerce recopila y mantiene la información de registro.





---

## **Parte 5. Adición de control de acceso a la tienda**





---

## Capítulo 24. Control de acceso en la tienda

WebSphere Commerce le permite determinar, mediante el control de acceso, qué tareas puede realizar un usuario determinado, ya sea un cliente o un administrador. Este capítulo describe cómo puede añadir control de acceso a la tienda, con lo que limitará las páginas que pueden ver y las tareas que pueden realizar los clientes.

Para obtener más información sobre el modelo de control de acceso de WebSphere Commerce, y sobre cómo se aplica el control de acceso a nivel de sitio, consulte la publicación *IBM WebSphere Commerce, Guía de control de acceso*. Para obtener más información acerca de cómo implementar el control de acceso en código personalizado, consulte la publicación *IBM WebSphere Commerce, Guía del programador*. Estas guías se pueden obtener en los URL siguientes:

 [http://www.ibm.com/software/webservers/commerce/wc\\_be/lit-tech-general.html](http://www.ibm.com/software/webservers/commerce/wc_be/lit-tech-general.html)

 [http://www.ibm.com/software/webservers/commerce/wc\\_pe/lit-tech-general.html](http://www.ibm.com/software/webservers/commerce/wc_pe/lit-tech-general.html)

---

### Descripción del control de acceso en WebSphere Commerce



El modelo de control de acceso de WebSphere Commerce se describe detalladamente en la publicación *IBM WebSphere Commerce, Guía de control de acceso*. Sin embargo, para comprender cómo el control de acceso afecta al desarrollo de la tienda se proporciona un breve resumen.

El control de acceso en WebSphere Commerce consta de los elementos siguientes: usuarios, acciones, recursos y relaciones.

- Los usuarios son aquellos que utilizan el sistema. Para fines de control de acceso, los usuarios se deben agrupar en grupos de acceso relevantes. Por ejemplo, es posible que en la tienda los usuarios se puedan agrupar en los siguientes grupos de acceso: clientes registrados, clientes invitados o grupos administrativos como, por ejemplo, representantes de servicio al cliente.
- Las acciones son las actividades que los usuarios pueden realizar en el recurso. Para fines de control de acceso, se deben agrupar las acciones en grupos de acciones relevantes. Por ejemplo, una acción que se utiliza comúnmente en una tienda es una vista. Las vistas se invocan para mostrar una página de la tienda a los clientes. Las vistas que se utilizan en la tienda, deben asignarse a un grupo de acciones.
- Los recursos son entidades protegidas. Por ejemplo, si la acción es una vista, el recurso de dicha acción es la clase `com.ibm.commerce.command.ViewCommand`. Para fines de control de acceso, se agrupan los recursos en grupos de recursos.
- Las relaciones son simplemente la relación que existe entre el usuario y el recurso. Es posible que las políticas de control de acceso requieran que se cumpla una relación entre el usuario y el recurso.

## Políticas de control de acceso

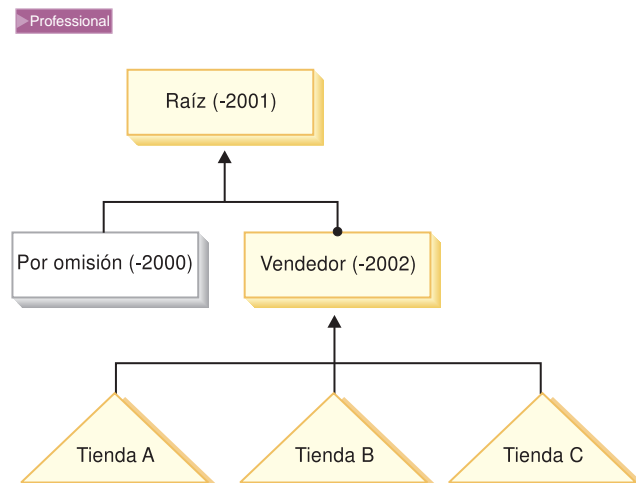
Las políticas de control de acceso autorizan a los grupos de acceso a realizar acciones determinadas en los recursos de WebSphere Commerce, siempre que los usuarios del grupo de acceso tengan una relación determinada con respecto al recurso.

WebSphere Commerce proporciona más de doscientas políticas de control de acceso por omisión que se cargan cuando se crea la instancia. Estas políticas cubren una amplia gama de actividades comunes de la empresa, incluida la creación y el proceso de pedidos y el intercambio, como por ejemplo,  las solicitudes de presupuesto y  los contratos. En la publicación *IBM WebSphere Commerce, Guía de control de acceso* puede encontrar información sobre las políticas por omisión.

Cada política de control de acceso es propiedad de una entidad de organización. Una política de control de acceso solamente se puede aplicar a los recursos que son propiedad del propietario de la política de control de acceso.

**Nota:** En relación con el control de acceso, la propiedad de los recursos tiene un significado especial. Todos los recursos deben implementar la interfaz `com.ibm.commerce.security.Protectable`. Uno de los métodos de esta interfaz es `getOwner()`, que devuelve el ID de miembro del propietario del recurso. Por ejemplo, el bean de entidad de pedido es un recurso protegido ya que su interfaz remota amplía la interfaz `Protectable`. La implementación del pedido de `getOwner()` es tal que un recurso de pedido específico devuelve el propietario de la tienda en la que se ha formalizado el pedido. En el caso de las políticas en las que el recurso es un mandato, por ejemplo, `com.ibm.commerce.command.ViewCommand`, la implementación por omisión de `getOwner()` devuelve el propietario de la tienda que actualmente está en el contexto del mandato. Si no hay ninguna tienda en el contexto del mandato, entonces se utiliza la organización raíz como propietario. Para obtener más información, consulte la publicación *IBM WebSphere Commerce, Guía del programador*.

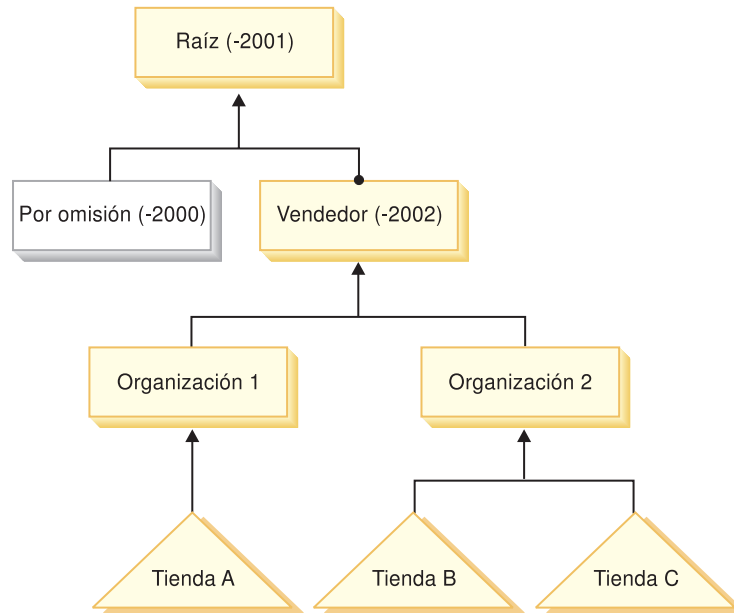
Observe el diagrama siguiente:



En WebSphere Commerce Professional Edition, la organización raíz es la organización con el nivel más alto. Es la propietaria de todas las entidades de organización. Como resultado, las políticas de control de acceso que son propiedad

de la organización raíz se aplican a todos los recursos del sitio. Si una política de control de acceso es propiedad de la organización del vendedor, la política de control de acceso solamente se aplica a los recursos que son propiedad del vendedor.

► Business



En WebSphere Commerce Business Edition, la organización raíz es la organización con el nivel más alto. Es la propietaria de todas las entidades de organización. Como resultado, las políticas de control de acceso que son propiedad de la organización raíz se aplican a todos los recursos del sitio. Si la organización del vendedor es la propietaria de una política de control de acceso, entonces la política de control de acceso solamente se aplicará a los recursos que sean propiedad de las organizaciones 1 y 2, incluidas las tiendas A, B y C. Si la política de control de acceso fuera propiedad de la organización 1, entonces se aplicaría únicamente a los recursos, incluida la tienda A. Si la política de control de acceso fuera propiedad de la organización 2, se aplicaría a los recursos de las tiendas B y C.

► Business

Dado que las políticas de control de acceso son propiedad de las entidades de organización, si va a crear varias tiendas en el sitio y desea aplicar diferentes políticas de control de acceso a las diferentes tiendas, debe crear organizaciones individuales que serán las propietarias de cada tienda.

**Nota:** La mayor parte de las políticas de control de acceso que se proporcionan con WebSphere Commerce son propiedad de la organización raíz y se aplican a todos los recursos del sitio. Las políticas de control de acceso que son propiedad de la organización raíz se denominan políticas de nivel de sitio. Las políticas que son propiedad de otras entidades de organización se denominan políticas de nivel de organización.

## Control de acceso en las tiendas

Todas las tiendas creadas en WebSphere Commerce están sometidas a las políticas de control de acceso por omisión que son propiedad de la organización raíz. Los recursos de las tiendas también están sometidos a las políticas de control de acceso


propiedad de la organización que posee la tienda y también a cualquier política de control de acceso que sea propiedad de los ascendientes de dicha organización.


Por omisión, la organización raíz es la propietaria de la mayor parte de las políticas de control de acceso. Sin embargo, si crea la tienda basándose en una de las tiendas de ejemplo que se incluyen con WebSphere Commerce, creará políticas de control de acceso que serán propiedad de la organización que posee la tienda. Para obtener más información acerca de las políticas de control de acceso, consulte el apartado “Control de acceso en las tiendas de ejemplo”.

Cuando cree su propia tienda, independientemente de si está basada o no en un ejemplo, es posible que desee crear una política de control de acceso nueva o modificar las políticas existentes, que solamente se aplicarán a las tiendas que sean propiedad de dicha organización. Por ejemplo, si crea vistas nuevas para mostrar las páginas de la tienda, deberá asignar políticas de control de acceso a estas vistas.

Los datos de control de acceso a nivel de organización se definen en archivos de políticas de control de acceso de alto nivel. Estos archivos definen las acciones posibles, los grupos de acciones, los recursos, los grupos de recursos y las relaciones que cualquier política puede utilizar. También definen las políticas específicas de una organización determinada. Las tiendas de ejemplo que se incluyen con WebSphere Commerce contienen estos archivos de políticas de control de acceso de alto nivel. El apartado siguiente describe cómo las tiendas de ejemplo utilizan los archivos de políticas de control de acceso para definir la información de la organización.

### Control de acceso en las tiendas de ejemplo

Todas las tiendas de ejemplo contienen archivos de políticas de control de acceso de alto nivel, que definen las políticas de control de acceso que se han creado específicamente para las tiendas. Estas políticas son propiedad de la organización propietaria de la tienda. A continuación, se describen las políticas de control de acceso definidas para las tiendas de ejemplo NewFashion y  ToolTech.

Las políticas de control de acceso creadas para  ToolTech son las siguientes:

- AllUsersForToolTechExecuteToolTechAllUsersViews
- RegisteredApprovedUsersForToolTechExecuteToolTechRegisteredApprovedUsersViews






La política de control de acceso creada para NewFashion es la siguiente:

- AllUsersExecuteNewFashionAllUsersViews

Estas políticas de control de acceso determinan los usuarios que verán determinadas vistas de las tiendas de ejemplo o de las tiendas basadas en estos ejemplos.

La información de control de acceso de estas políticas se define en dos archivos de políticas de control de acceso de alto nivel, que definen las acciones posibles, los grupos de acciones, los recursos y las definiciones de políticas que se utilizan en las tiendas de ejemplo: *nombretiendaejemploAccessPolicies.xml* y *nombretiendaejemploAccessPolicies\_entorno\_nacional.xml*. Estos archivos se encuentran en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\samples\stores\  
*nombretiendaejemplo*


-  unidad:\Archivos de programa\WebSphere\CommerceServer\samples\stores\nombretiendaejemplo
  -  /usr/WebSphere/CommerceServer/samples/stores/nombretiendaejemplo
  -  /opt/WebSphere/CommerceServer/samples/stores/nombretiendaejemplo
  -  /opt/WebSphere/CommerceServer/samples/stores/nombretiendaejemplo
  -  /QIBM/ProdData/WebCommerce/samples/stores/nombretiendaejemplo
- donde *nombretiendaejemplo* es el nombre del archivador de tienda de ejemplo en que está basada la tienda, por ejemplo, NewFashion.

Antes de empaquetar las tiendas de ejemplo como un archivador de tienda, estos dos archivos de políticas de control de acceso se transforman, lo que da como resultados dos archivos XML, adecuados para utilizarlos con Loader package. Estos archivos XML transformados se empaquetan en el archivador de tienda y se publican con el resto del archivador de tienda.

**Descripción de los archivos de políticas de control de acceso de las tiendas de ejemplo:** Para comprender cómo se añade el control de acceso a nivel de tienda, debe familiarizarse con los archivos de políticas de control de acceso de alto nivel de las tiendas de ejemplo. Los ejemplos siguientes se han tomado del archivo

 ToolTechAccessPolicies.xml.

*Definición de acciones:* La primera sección del archivo

 ToolTechAccessPolicies.xml define las acciones nuevas de la tienda, que no quedan cubiertas por las políticas de control de acceso existentes. En este caso, todas las acciones son vistas que se utilizan en la tienda. Para poder visualizar una página de la tienda mediante una vista que pueda llamarse directamente desde un URL o que pueda iniciarse mediante un redireccionamiento desde otro mandato (en lugar de iniciarla mediante un reenvío a la vista), debe definirlo como una acción. Observe el ejemplo siguiente:

```
<!-- [Definiciones de inicio de acción] -->
<!-- [este es el diccionario de acciones posibles -->
<Action Name="GenericApplicationError"
 CommandName="GenericApplicationError">
 </Action>

<Action Name="GenericSystemError"
 CommandName="GenericSystemError">
 </Action>

<Action Name="OrderOptionsView"
 CommandName="OrderOptionsView">
 </Action>

<!--[Fin de definiciones de acción] -->
```

donde

- Action Name es la etiqueta que se utiliza para hacer referencia a esta acción en el archivo XML. En estos ejemplos, la etiqueta es la misma que el nombre de vista.
- CommandName es el nombre de la vista almacenada en la columna VIEWNAME de la tabla VIEWREG. CommandName se almacenará en la columna Action de la tabla ACACTION.

**Nota:** Si ya se ha definido una acción en las políticas por omisión o en cualquier otra política de WebSphere Commerce, no es necesario que vuelva a definirla para cada política que utilice la acción

*Definición de las categorías de recursos:* La segunda sección del archivo define las categorías de recursos. Una categoría de recurso hace referencia a una clase de recurso. Las categorías de recursos identificadas para ToolTech son las clases `com.ibm.commerce.command.ViewCommand` y `com.ibm.commerce.tools.command.ToolsForwardViewCommand`.

```
<!-- [Inicio de definiciones de categorías de recursos] -->
<!-- diccionario de recursos protegibles -->

 <ResourceCategory Name="com.ibm.commerce.command.ViewCommandResourceCategory"
 ResourceBeanClass="com.ibm.commerce.command.ViewCommand">

</ResourceCategory>

 <ResourceCategory Name="com.ibm.commerce.tools.command.
ToolsForwardViewCommandResourceCategory"
 ResourceBeanClass="com.ibm.commerce.tools.command.ToolsForwardViewCommand">

</ResourceCategory>
<!-- [Fin de definiciones de categorías de recursos] -->
```

donde

- `ResourceCategory Name` es la etiqueta utilizada para hacer referencia a esta categoría de recursos en el archivo XML.
- `ResourceBeanClass` es el nombre de la clase.

**Nota:** Si ya se ha definido una categoría de recursos en las políticas por omisión o en cualquier otra política de WebSphere Commerce, no es necesario que vuelva a definirla para cada política que utilice la acción. La categoría de recursos definida en el ejemplo anterior ya está definida en las políticas por omisión, pero aquí se vuelve a definir para ilustrarla mejor.

*Definición de grupos de acciones:* La tercera sección define el grupo de acciones. El grupo de acciones es una agrupación de las acciones definidas en la primera sección del archivo. En el ejemplo ToolTech, todas las vistas de usuario nuevas se agrupan en el grupo `ToolTechAllUserViews`, que se utilizará en una política que permitirá a los usuarios acceder a dichas vistas, o en `ToolTechRegisteredApprovedUserViews`, que se utilizará en una política que permitirá solamente a los usuarios registrados acceder a dichas vistas.

**Nota:** También puede añadir acciones, definidas en cualquier otro lugar de WebSphere Commerce, a los grupos de acciones. Si se han definido en cualquier otro lugar de WebSphere Commerce, no es necesario definir estas acciones en la lista de acciones descrita en el apartado “Definición de acciones” en la página 205.

```
<!-- [Inicio de definiciones de grupo de acciones] -->
<!-- Diccionario de acciones agrupadas utilizables en políticas -->
<!-- vista cruzada de componentes y grupos de acciones relacionados -->
 <ActionGroup Name="ToolTechAllUsersViews"
 OwnerID="RootOrganization">

 <ActionGroupAction Name="UserRegistrationForm"/>
 <ActionGroupAction Name="UserRegistrationErrorView"/>
 <ActionGroupAction Name="GenericApplicationError"/>
 <ActionGroupAction Name="GenericSystemError"/>
 <ActionGroupAction Name="LogonForm"/>
```



```
</ActionGroup>
```

```
<!-- [Fin de definiciones de grupo de acciones] -->
```

donde

- ActionGroup Name es el nombre del grupo de acciones.
- OwnerID es el propietario del grupo de acciones. La organización raíz debe ser la propietaria del grupo de acciones.
- ActionGroupAction Name es el nombre de una acción perteneciente a este grupo. ActionGroupAction Name debe coincidir con el nombre definido en el elemento Action Name del apartado “Definición de acciones” en la página 205.

*Definición de grupos de recursos:* La sección siguiente define los grupos de recursos. Un grupo de recursos identifica un conjunto de recursos relacionados.

```
<!-- [Inicio de definiciones de grupos de recursos] -->
```

```
<!-- Diccionario de recursos agrupados utilizables en políticas -->
```

```
<!-- Recursos agrupados que permiten ejecutar vistas
mediante un mandato que sea una ampliación de com.ibm.commerce.command.ViewCommand
(con o sin la infraestructura de herramientas) -->
```

```
<ResourceGroup Name="ViewCommandResourceGroup" OwnerID="RootOrganization">
```

```
<ResourceGroupResource Name="com.ibm.commerce.command.ViewCommandResourceCategory"/>
</ResourceGroup>
```

```
</ResourceGroup> !-- [Fin de definiciones de grupo de recursos] -->
```

donde

- ResourceGroup Name es el nombre del grupo de recursos.
- OwnerID es el propietario del grupo de recursos. La organización raíz debe ser la propietaria del grupo de recursos.
- ResourceGroupResource Name es el nombre de una categoría de recurso incluida en el grupo.

*Definición de políticas:* La sección final define las nuevas políticas utilizadas en la tienda.

```
!-- [Inicio de definiciones de políticas] -->
```

```
!-- AllUsers for ToolTech can execute ToolTechAllUsersViews -->
```

```
<Policy Name="AllUsersForToolTechExecuteToolTechAllUsersViews"
```

```
OwnerID="MEMBER_ID"
```

```
UserGroup="AllUsers"
```

```
UserGroupOwner="RootOrganization"
```

```
ActionGroupName="ToolTechAllUsersViews"
```

```
ResourceGroupName="ViewCommandResourceGroup">
```

```
</Policy>
```

```
!-- RegisteredApprovedUsers for ToolTech can execute
ToolTechRegisteredApprovedUsersViews -->
```

```
<Policy Name="RegisteredApprovedUsersFor
ToolTechExecuteToolTechRegisteredApprovedUsersViews"
```

```
OwnerID="MEMBER_ID"
```

```
UserGroup="RegisteredApprovedUsers"
```

```
UserGroupOwner="RootOrganization"
```

```
ActionGroupName="ToolTechRegisteredApprovedUsersViews"
```

```
ResourceGroupName="ViewCommandResourceGroup">
```

```
</Policy>
```

```
!-- [Fin de definiciones de política] -->
```

donde

- Policy Name es el nombre de la política que se está definiendo.
- OwnerId es el propietario de la política. En este caso, el propietario de la política es la organización que posee la tienda.
- UserGroup es el grupo de usuarios (el grupo de acceso) al que se aplica la política.
- UserGroupOwner es el propietario del grupo de acceso. En este ejemplo, el propietario del grupo de acceso es distinto del propietario de la política. Si el propietario de la política y UserGroupOwner son el mismo, este elemento se puede omitir.
- ActionGroupName es el grupo de acciones al que se aplica la política.
- ResourceGroupName es el grupo de recursos al que se aplica la política.

---

## Adición de control de acceso a la tienda




Desde el punto de vista del desarrollo de tienda, los tipos de control de acceso más comunes necesarios son para las vistas nuevas y los mandatos que cree para la tienda. Sin embargo, es posible que desee añadir otros tipos de control de acceso a la tienda. Para obtener más información acerca del control de acceso para vistas, mandatos y otras características, consulte la publicación *IBM WebSphere Commerce, Guía de control de acceso*. Antes de continuar con los pasos siguientes que se describen en esta guía, no olvide leer la publicación *IBM WebSphere Commerce, Guía de control de acceso*.

Si va a añadir nuevas características de control de acceso a una tienda basándose en la tienda de ejemplo, edite el archivo XML existente de política de control de acceso de alto nivel. Para obtener información detallada, consulte el apartado “Edición de los archivos de control de acceso del archivador de la tienda” en la página 211. Si va a añadir control de acceso a una tienda sin basarse en una tienda de ejemplo, deberá crear un nuevo archivo XML de política de control de acceso de alto nivel y luego modificarlo. Para obtener información detallada, consulte el apartado “Creación de control de acceso en la tienda”.

## Creación de control de acceso en la tienda

Los elementos de control de acceso son diferentes a los demás elementos de la tienda, ya que en el control de acceso se crean dos archivos XML de control de acceso de alto nivel y luego se transforman. Los archivos XML resultantes se pueden cargar posteriormente en la base de datos mediante Loader package o se pueden utilizar en el archivador de la tienda.

Para crear elementos de control de acceso, efectúe lo siguiente:













1. Revise los archivos XML de alto nivel utilizados para crear los elementos de tienda de las tiendas de ejemplo: *nombretiendaejemploAccessPolicies.xml* y *nombretiendaejemploAccessPolicies\_entorno nacional.xml*. Estos archivos se encuentran en el directorio siguiente:
  -  unidad:\WebSphere\CommerceServer\samples\stores\nombretiendaejemplo
  -  unidad:\Archivos de programa\WebSphere\CommerceServer\samples\stores\nombretiendaejemplo
  -  /usr/WebSphere/CommerceServer/samples/stores/nombretiendaejemplo






- Solaris /opt/WebSphere/CommerceServer/samples/stores/*nombretiendaejemplo*
  - Linux /opt/WebSphere/CommerceServer/samples/stores/*nombretiendaejemplo*
  - 400 /QIBM/ProdData/WebCommerce/samples/stores/*nombretiendaejemplo*  
donde *nombretiendaejemplo* es el nombre del archivador de tienda de ejemplo en que está basada la tienda, por ejemplo, NewFashion.
2. Revise la información del Apéndice B, “Creación de los datos” en la página 327.
  3. Cree un archivo *nombretiendaAccessPolicies.xml*, copiando uno de los archivos *nombretiendaejemploAccessPolicies.xml* o creando uno nuevo. Para obtener más información, consulte el archivo DTD correspondiente a *nombretiendaejemploAccessPolicies.xml*. Los archivos DTD se encuentran en el directorio siguiente:
    - NT unidad:\WebSphere\CommerceServer\xml\policies\dtd
    - 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\policies\dtd
    - AIX /usr/WebSphere/CommerceServer/xml/policies/dtd
    - Solaris /opt/WebSphere/CommerceServer/xml/policies/dtd
    - Linux /opt/WebSphere/CommerceServer/xml/policies/dtd
    - 400 /QIBM/ProdData/WebCommerce/xml/policies/dtd
  4. Cree un archivo XML de alto nivel diferente para cada entorno nacional al que dé soporte la tienda. El archivo específico del entorno nacional debe especificar toda la información de nombre de visualización y descripción, para que se pueda traducir fácilmente. Cree este archivo, *nombretiendaAccessPolicies\_entorno\_nacional.xml* para cada idioma de la tienda, copiando uno de los archivos *nombretiendaejemploAccessPolicies\_ent\_nacional.xml* o creando uno nuevo.


**Nota:** Deberá crear un archivo específico del entorno nacional incluso si la tienda solamente da soporte a un idioma.

5. Añada la información de control de acceso correcta al archivo. Para obtener más información, consulte el apartado “Control de acceso en las tiendas de ejemplo” en la página 204 y la publicación *IBM WebSphere Commerce, Guía de control de acceso*.
6. Copie *nombretiendaAccessPolicies.xml* y *nombretiendaAccessPolicies\_ent\_nacional.xml* en el directorio siguiente:
  - NT unidad:\WebSphere\CommerceServer\xml\policies\xml
  - 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\policies\xml
  - AIX /usr/WebSphere/CommerceServer/xml/policies/xml
  - Solaris /opt/WebSphere/CommerceServer/xml/policies/xml
  - Linux /opt/WebSphere/CommerceServer/xml/policies/xml
  - 400 Cópelo en cualquier directorio de datos de usuario. Especifique la vía de acceso completa a la DTD en los archivos XML existentes. Los

archivos DTD de control de acceso están situados en el directorio siguiente:  
/QIBM/ProdData/WebCommerce/xml/policiess/dtd

7. Ejecute el mandato `xmltransform` para transformar `nombretiendaAccessPoliciess.xml`.
  - a. En un indicador de mandatos, vaya al directorio siguiente:
    -  unidad:\WebSphere\CommerceServer\bin
    -  unidad:\Archivos de programa\WebSphere\CommerceServer\bin
    -  /usr/WebSphere/CommerceServer\bin
    -  /opt/WebSphere/CommerceServer/bin
    -  /opt/WebSphere/CommerceServer/bin
  - b. A continuación, escriba: `xmltransform -infile ..\xml\policiess\xml\samplestorenameAccessPoliciess.xml -transform ..\xml\policiess\xsl\accesscontrol.xsl -outfile ..\xml\policiess\xml\samplestorenameAccessPoliciessOut.xml`   
TRNWCXML INFILE (archivo entrada)  
TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policiess/xsl/accesscontrol.xsl') INSTRoot(raíz\_instancia)  
OUTFILE(archivo\_salida)
  - c. Consulte el archivo de anotaciones cronológicas siguiente para asegurarse de que la transformación se ha llevado a cabo correctamente:
    -  unidad:\WebSphere\CommerceServer\bin  
xmltransform.db2.log
    -  unidad:\Archivos de programa\WebSphere\CommerceServer\bin\xmltransform.db2.log
    -  /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
    -  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
    -  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
    -  /QIBM/UserData/WebCommerce/instancias/nombreinstancia  
/logs/TRNWCXML.tx

Si la transformación se ha llevado a cabo correctamente, se mostrará el mensaje siguiente: "<DATE> <TIME> java.lang.Class main XMLTransformer Transformación satisfactoria"
8. Ejecute el mandato `xmltransform` para transformar `nombretiendaAccessPoliciess_ent_nacional.xml`.
  - a. En un indicador de mandatos, vaya al directorio siguiente:
    -  unidad:\WebSphere\CommerceServer\bin
    -  unidad:\Archivos de programa\WebSphere\CommerceServer\bin
    -  /usr/WebSphere/CommerceServer\bin
    -  /opt/WebSphere/CommerceServer/bin
    -  /opt/WebSphere/CommerceServer/bin
  - b. A continuación, escriba: `xmltransform -infile ..\xml\policiess\xml\nombretiendaAccessPoliciess_ent_nacional.xml -transform ..\xml\policiess\xsl\accesscontrolnls.xsl -outfile ..\xml\policiess\xml\nombretiendaAccessPoliciessOut_ent_nacional.xml`

- c.  TRNWCSXML INFILE(archivo entrada)  
 TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policies/  
 xsl/accesscontrolInIs.xsl') INSTRROOT(raíz\_instancia)  
 OUTFILE(archivo\_salida)
9. Efectúe los cambios siguientes en los archivos XML resultantes:
- En *nombretiendaAccessPolicesOut.xml*, sustituya los códigos de apertura y cierre por los siguientes:
 

```
<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```
  - En *nombretiendaAccessPolicesOut\_ent\_nacional.xml*, sustituya los códigos de apertura y cierre por los siguientes:
 

```
<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```
  - En *nombretiendaAccessPolicesOut\_ent\_nacional.xml*, sustituya @entorno\_nacional por &entorno\_nacional. Por ejemplo, cambie LANGUAGE\_ID="@en\_US" por LANGUAGE\_ID="&en\_US;"
  - En *nombretiendaAccessPolicesOut\_ent\_nacional.xml*, localice la referencia a la tabla "acpoldesc". Suprima el signo @ al final del valor ACPOLICY\_ID. Por ejemplo, cambie "@AllUsersExecuteInFashionAllUsersViews@" por "@AllUsersExecuteInFashionAllUsersViews".
  - En *nombretiendaAccessPolicesOut.xml*, sustituya MEMBER\_ID="MEMBER\_ID" por MEMBER\_ID="&MEMBER\_ID;"
  - En *nombretiendaAccessPolicesOut.xml* sustituya la referencia a la tabla "acpolicy". Suprima "@MEMBER\_ID" al final del valor ACPOLICY\_ID. Por ejemplo, cambie "@AllUsersExecuteInFashionAllUsersViews@MEMBER\_ID" por "@AllUsersExecuteInFashionAllUsersViews"
10. Llegado este punto tiene dos opciones:
- Cargar los datos de control de acceso mediante Loader package. Para obtener más información, consulte la Parte 7, "Publicación de la tienda" en la página 225.
  - Añadir los datos de control de acceso al archivador de la tienda. Para obtener más información sobre cómo crear un archivador de tienda, consulte la Parte 6, "Empaquetado de la tienda" en la página 215.









Para obtener más información sobre cómo utilizar los signos @ y &, consulte el Apéndice B, "Creación de los datos" en la página 327.

## Edición de los archivos de control de acceso del archivador de la tienda

Para cada tienda de ejemplo, WebSphere Commerce proporciona dos archivos XML de control de acceso transformados previamente, uno que se puede aplicar a todos los idiomas (*nombretiendaejemploAccessPolices.xml*), y otro que contiene información específica del entorno nacional (*nombretiendaejemploAccessPolices\_ent\_nacional.xml*). Debe transformar cada uno de estos archivos, con lo que obtendrá dos archivos XML, uno que es aplicable a todos los idiomas (*nombretiendaejemploAccessPolicesOut.xml*) y uno que contiene información específica del entorno nacional (*nombretiendaejemploAccessPolicesOut\_ent\_nacional.xml*).

Para editar el elemento de base de datos de control de acceso del archivador de tienda, efectúe lo siguiente:







1. Localice los archivos XML de control de acceso transformados previamente de la tienda de ejemplo en la que está basada su tienda. Estos archivos son *nombretiendaejemploAccessPoliciess.xml* y *nombretiendaejemploAccessPoliciess\_ent\_nacional.xml*. Por omisión, estos archivos están situados en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\samples\stores\*nombretiendaejemplo*
-  unidad:\Archivos de programa\WebSphere\CommerceServer\samples\stores\*nombretiendaejemplo*
-  /usr/WebSphere/CommerceServer/samples/stores/*nombretiendaejemplo*
-  /opt/WebSphere/CommerceServer/samples/stores/*nombretiendaejemplo*
-  /opt/WebSphere/CommerceServer/samples/stores/*nombretiendaejemplo*
-  /QIBM/ProdData/WebCommerce/samples/stores/*nombretiendaejemplo*

donde *nombretiendaejemplo* es el nombre del archivador de tienda de ejemplo en que está basada la tienda, por ejemplo, NewFashion.








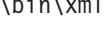









**Nota:** Si modifica los archivos DTD correspondientes es posible que las políticas no se puedan utilizar.

2. Efectúe los cambios necesarios en el archivo. Para obtener más información acerca del archivo existente, consulte el apartado “Descripción de los archivos de políticas de control de acceso de las tiendas de ejemplo” en la página 205. Para obtener más información sobre los métodos de control de acceso disponibles en WebSphere Commerce, consulte la publicación *IBM WebSphere Commerce, Guía de control de acceso*.
3. Copie *nombretiendaejemploAccessPoliciess.xml* y *nombretiendaejemploAccessPoliciess\_ent\_nacional.xml* en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\xml\policiess\xml
-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\policiess\xml
-  /usr/WebSphere/CommerceServer/xml/policiess/xml
-  /opt/WebSphere/CommerceServer/xml/policiess/xml
-  /opt/WebSphere/CommerceServer/xml/policiess/xml
-  Cópielo en cualquier directorio de datos de usuario. Especifique la vía de acceso completa a la DTD en los archivos XML existentes. Los archivos DTD de control de acceso están situados en el directorio siguiente: /QIBM/ProdData/WebCommerce/xml/policiess/dtd

4. Ejecute el mandato `xmltransform` para transformar *nombretiendaejemploAccessPoliciess.xml*.
  - a. En un indicador de mandatos, vaya al directorio siguiente:

-  unidad:\WebSphere\CommerceServer\bin

-  unidad:\Archivos de programa\WebSphere\CommerceServer\bin
  -  /usr/WebSphere/CommerceServer\bin
  -  /opt/WebSphere/CommerceServer/bin
  -  /opt/WebSphere/CommerceServer/bin
- b. A continuación, escriba `xmltransform -infile`  
`..\xml\policias\xml\samplestorenameAccessPolicias -transform`  
`..\xml\policias\xsl\accesscontrol.xsl -outfile`  
`..\xml\policias\xml\samplestorenameAccessPoliciasOut.xml`
-  `TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policias`  
`/xsl/accesscontrol.xsl')`  
`INSTROOT(raíz_instancia) OUTFILE(archivo_salida)`
- c. Consulte el archivo de anotaciones cronológicas siguiente para asegurarse de que la transformación se ha llevado a cabo correctamente:
-  unidad:\WebSphere\CommerceServer\bin\xmltransform.db2.log
  -  unidad:\Archivos de programa\WebSphere\CommerceServer\bin\xmltransform.db2.log
  -  /usr/WebSphere/CommerceServer/bin/xmltransform.db2.log
  -  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
  -  /opt/WebSphere/CommerceServer/bin/xmltransform.db2.log
  -  /QIBM/UserData/WebCommerce/instances/  
nombreinstancia/logs/TRNWCSXML.tx
- Si la transformación se ha llevado a cabo correctamente, se mostrará el mensaje siguiente: "`<DATE> <TIME> java.lang.Class main XMLTransformer Transformación satisfactoria`"
5. Ejecute el mandato `xmltransform` para transformar `nombretiendaejemploAccessPolicias_ent_nacional.xml`.
- a. En un indicador de mandatos, vaya al directorio siguiente:
-  unidad:\WebSphere\CommerceServer\bin
  -  unidad:\Archivos de programa\WebSphere\CommerceServer\bin
  -  /usr/WebSphere/CommerceServer\bin
  -  /opt/WebSphere/CommerceServer/bin
  -  /opt/WebSphere/CommerceServer/bin
- b. A continuación, escriba: `xmltransform -infile`  
`..\xml\policias\xml\samplestorenameAccessPolicias_ent_nacional.xml`  
`-transform ..\xml\policias\xsl\accesscontrolnls.xsl -outfile`  
`..\xml\policias\xml\samplestorenameAccessPoliciasOut_ent_nacional.xml`
- c.  `TRNWCSXML INFILE(archivo entrada)`  
`TRANSFORM('/QIBM/ProdData/WebCommerce/xml/policias/`  
`xsl/accesscontrolnls.xsl')`  
`INSTROOT(raíz_instancia) OUTFILE(archivo_salida)`
6. Efectúe los cambios siguientes en los archivos XML resultantes:
- a. En `nombretiendaejemploAccessPoliciasOut.xml`, sustituya los códigos de apertura y cierre por los siguientes:

```
<?xml version="1.0"?>
<!DOCTYPE accesscontrol-asset SYSTEM "accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```

- b. En *nombretiendaejemploAccessPoliciesOut\_ent\_nacional.xml*, sustituya los códigos de apertura y cierre por los siguientes:

```
<?xml version="1.0" encoding="correct language code for the file"?>
<!DOCTYPE accesscontrol-asset SYSTEM "../accesscontrol.dtd">
<accesscontrol-asset>
</accesscontrol-asset>
```







- c. En *nombretiendaejemploAccessPoliciesOut\_ent\_nacional.xml*, sustituya @entorno\_nacional por &entorno\_nacional. Por ejemplo, cambie LANGUAGE\_ID="@en\_US" por LANGUAGE\_ID="&en\_US;"

- d. En *nombretiendaejemploAccessPoliciesOut\_ent\_nacional.xml*, localice la referencia a la tabla "acpoldesc". Suprima el signo @ al final del valor ACPOLICY\_ID. Por ejemplo, cambie "@AllUsersExecuteInFashionAllUsersViews@" por "@AllUsersExecuteInFashionAllUsersViews".

- e. En *nombretiendaejemploAccessPoliciesOut.xml*, sustituya MEMBER\_ID="MEMBER\_ID" por MEMBER\_ID="&MEMBER\_ID;"

- f. En *nombretiendaejemploAccessPoliciesOut.xml* localice la referencia a la tabla "acpolicy". Suprima "@MEMBER\_ID" al final del valor ACPOLICY\_ID. Por ejemplo, cambie "@AllUsersExecuteInFashionAllUsersViews@MEMBER\_ID" por "@AllUsersExecuteInFashionAllUsersViews"

7. Localice el archivo del archivador de tienda de su tienda (por ejemplo, mitienda.sar ). Por omisión, los archivos de los archivadores de tienda están en el directorio siguiente:

-  NT unidad:\WebSphere\CommerceServer\instances\  
nombreinstancia\sar
-  2000 unidad:\Archivos de  
programa\WebSphere\CommerceServer\instances \nombreinstancia\sar
-  AIX /usr/WebSphere/CommerceServer/instances/nombreinstancia/sar
-  Solaris /opt/WebSphere/CommerceServer/instances/nombreinstancia/sar
-  Linux /opt/WebSphere/CommerceServer/instances/nombreinstancia/sar
-  400 /QIBM/UserData/WebCommerce/instances/nombreinstancia/sar

8. Cambie el nombre de *nombretiendaejemploAccessPoliciesOut.xml* y *nombretiendaejemploAccessPoliciesOut\_ent\_nacional.xml* por *accesscontrol.xml*

**Nota:** por omisión, el archivo *accesscontrol.xml* específico del entorno nacional está situado en el directorio *data/entorno\_nacional*, por ejemplo, *data/en\_US*.

9. Abra el archivo del archivador de tienda utilizando un programa ZIP.
10. Sustituya los archivos *accesscontrol.xml* y *accesscontrol.xml* específico del entorno nacional en el archivo del archivador de tienda por los archivos cuyo nombre ha cambiado en el paso 8.
11. Guarde el archivo del archivador de tienda.



---

## Parte 6. Empaquetado de la tienda



---

## Capítulo 25. Empaquetado de una tienda

Si desea utilizar la tienda como un ejemplo que entregará a otros, desea desplegarla en otro servidor o plataforma, o desea utilizarla como base para crear otras tiendas, puede empaquetarla con el formato de archivador de tienda.

Las tiendas de ejemplo que se proporcionan con WebSphere Commerce están empaquetadas como archivadores de tienda. Los archivos de los archivadores de tienda de ejemplo están agrupados como se indica a continuación:

- **Elementos Web:** Los archivos utilizados para crear las páginas de tienda como, por ejemplo, archivos HTML, archivos JSP, imágenes, gráficos y archivos de inclusión. Los elementos web se han agrupado como un archivo comprimido (`webapp.zip`) en el archivador de tienda.
- **Paquete de recursos de propiedades (opcional):** Contiene el texto para las páginas de la tienda. Si la tienda soporta más de un idioma, el paquete de recursos contendrá varios paquetes, uno por idioma. Los paquetes de recursos de propiedades se han agrupado en el archivo comprimido (`properties.zip`) en el archivador de tienda.
- **Elementos de datos de tienda:** Los datos que se deben cargar en la base de datos. Los elementos de datos de tienda como las campañas, el catálogo, las monedas, la información de despacho de pedidos, cálculo de precios, envío, tienda y la información sobre los impuestos.
- **Elementos de pago:** Información de configuración para IBM Payment Manager.
- **Un descriptor:** un archivo XML, `sarinfo.xml`, que describe el archivador de tienda, incluidos los nombres del archivador comprimido de elementos Web, de los paquetes de recursos y de los archivos XML de elementos de base de datos de la tienda. El archivo `sarinfo.xml` también contiene los nombres de los archivos de inclusión y de los archivos de comprobación de coherencia, así como información acerca del archivo archivador que se necesita durante el proceso de publicación. `sarinfo.xml` es el único archivo obligatorio en un archivador de tienda.


---

### Creación de un archivador de tienda







Para empaquetar la tienda como un archivador de tienda, efectúe lo siguiente:

1. Revise la estructura y el contenido de los archivadores de tienda de ejemplo que se proporcionan con WebSphere Commerce.

Los archivos de archivador de tienda están ubicados en el directorio siguiente:







-  `unidad:\WebSphere\CommerceServer\samplstores`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores`
-  `/usr/WeSphere/CommerceServer/samplstores`
-  `/opt/WebSphere/CommerceServer/samplstores`
-  `/opt/WebSphere/CommerceServer/samplstores`
-  `/qibm/proddata/WebCommerce/samplstores`

Para ver el archivador de tienda, utilice un programa de descompresión.

2. Cree un directorio temporal en WebSphere Commerce Server para la tienda. Por ejemplo, *mitienda*.
3. Agrupe los elementos Web (archivos JSP, HTML, imágenes) en un directorio que se llama *webapp* y está incluido en el directorio temporal. Utilice un programa ZIP y cree un archivador comprimido de la carpeta *webapp*.
4. (Opcional) Agrupe los paquetes de recursos de propiedades en un directorio que se llama *properties* y está incluido en el directorio temporal. Si la tienda soporta más de un idioma, el paquete de recursos contendrá varios paquetes, uno por idioma. Utilice un programa ZIP y cree un archivador comprimido de la carpeta *properties*.
5. Agrupe los elementos de datos de la tienda en un directorio que se llama *data* y está incluido en el directorio temporal. Si la tienda da soporte a varios idiomas, cree subdirectorios para la información específica del idioma utilizando los nombres del entorno nacional. Por ejemplo, *es\_ES*.
6. (Opcional) Copie el archivo *sarrule.xml* a partir de un archivador de tienda existente en el directorio *data*. El archivo *sarrule.xml* se encuentra en el directorio *data* de los archivadores de tienda de ejemplo. El archivo *sarrule.xml* comprueba la coherencia cuando se publica utilizando los Servicios de tienda. Para obtener más información acerca del archivo *sarrule*, consulte el Apéndice D, “*sarrule.xml*” en la página 335.
7. Cree un directorio llamado *SAR-INF* en el directorio temporal.
8. Cree un archivo *sarinfo.xml* para el archivador de tienda. Para obtener información acerca de las especificaciones XML, consulte el descriptor del archivador, *sarinfo.dtd* en el directorio siguiente:
  -  unidad:\WebSphere\CommerceServer\xml\sar
  -  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
  -  /usr/WeSphere/CommerceServer/xml/sar
  -  /opt/WebSphere/CommercServer/xml/sar
  -  /opt/WebSphere/CommercServer/xml/sar
  -  /qibm/proddata/WebCommerce/xml/sar
  - a. Utilizando como guía el ejemplo y la información del Apéndice C, “*sarinfo.xml*” en la página 329, cree un archivo *sarinfo.xml*. El orden en que se publican los elementos de datos es importante ya que algunos elementos de datos se deben publicar antes que otros. Como resultado, el orden de los elementos, tal y como se especifica en el archivo *sarinfo.xml*, deberá coincidir con el orden de los elementos especificados en los archivos *sarinfo.xml* de las tiendas de ejemplo.
 

**Nota:** Si selecciona que se incluya un archivo *sarrule.xml* en el archivador de tienda (vea el paso 6), debe incluir información acerca de *sarrule.xml* en el archivo *sarinfo.xml*. Si selecciona que no se incluya un archivo *sarrule.xml*, no olvide que no debe mencionarlo en el archivo *sarinfo.xml*.







    - b. Guarde *sarinfo.xml* en el directorio *SAR-INF* que ha creado en el paso 6.
9. Cree un archivo ZIP compuesto por el archivo ZIP de los elementos Web, los paquetes de recursos de propiedades, los elementos de datos de la tienda y el directorio *sar-inf*. A este archivo ZIP le dará el nombre *nombreachivadortienda.sar*.

10. Si desea editar o publicar el archivador de tienda utilizando los servicios de tienda, guarde el archivo *nombearchivadortienda.sar* en el directorio siguiente:
  -  NT unidad:\WebSphere\CommerceServer\instances\  
*nombreinstantia\sar*
  -  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\  
\instances\*nombreinstantia\sar*
  -  AIX /usr/WebSphere/CommerceServer/instances/*nombreinstantia/sar*
  -  Solaris /opt/WebSphere/CommerceServer/instances/*nombreinstantia/sar*
  -  Linux /opt/WebSphere/CommerceServer/instances/*nombreinstantia/sar*
  -  400 /QIBM/UserData/WebCommerce/instances/*nombreinstantia/sar*
11. El archivador de tienda figurará ahora en la lista de archivadores de tienda de los servicios de tienda.

---

## Creación de un archivador de tienda de ejemplo

Después de empaquetar la tienda como un archivador de tienda, puede optar por utilizarla como una tienda de ejemplo en los servicios de tienda. Un archivador de tienda de ejemplo es un archivador de tienda que se copia y utiliza como base sobre la que se crean nuevas tiendas. Para utilizar el archivador de tienda como un archivador de tienda de ejemplo, efectúe lo siguiente:

1. Guarde el archivador de tienda en el directorio siguiente:
  -  NT unidad:\WebSphere\CommerceServer\samplstores\  
*nombearchivadortienda*
  -  2000 unidad:\Archivos de programa\WebSphere\CommerceServer\  
\samplstores\*nombearchivadortienda*
  -  AIX /usr/WeSphere/CommerceServer/samplstores/  
*nombearchivadortienda*
  -  Solaris /opt/WebSphere/CommerceServer/samplstores/  
*nombearchivadortienda*
  -  Linux /opt/WebSphere/CommerceServer/samplstores/  
*nombearchivadortienda*
  -  400 /qibm/proddata/WebCommerce/samplstores/  
*nombearchivadortienda*
2. (Opcional) Cree páginas de vista previa. Para poder visualizar las vistas previas de las páginas de la tienda en los Servicios de tienda, deberá crear las páginas de vista previa. Efectúe lo siguiente:
  - a. (Opcional) En los Servicios de tienda, seleccione **Nuevo**. Se muestra la página Crear archivador de tienda. En la lista **Ejemplo**, seleccione una de las tiendas de ejemplo y luego pulse **Vista previa**. Las páginas que desea visualizar se denominan páginas de vista previa. Estas páginas son archivos HTML que presentan un flujo de compra de ejemplo definido previamente y actúan como una vista previa de la tienda de ejemplo.
  - b. Determine el flujo de compra que desea mostrar en las páginas de vista previa.
  - c. (Opcional) Cree algunos datos de ejemplo en una tienda publicada. Por ejemplo, añada artículos al carro de la compra y cree algunas direcciones de







envío y direcciones de facturación. Creará las páginas de vista previa desde esta tienda y los datos harán que las páginas resulten más reales.

- d. Con Internet Explorer, vaya hasta la tienda. Guarde el HTML de cada página, seleccionando Archivo, Guardar como. También deberá guardar la hoja de estilo (.css) y las imágenes. Guarde los archivos en los directorios siguientes:

- *stylesheet.css*













-  unidad:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombearchivadortienda\preview
-  unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombearchivadortienda\preview
-  /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview
-  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview
-  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview
-  /Qibm/UserData/WebAsAdv4/WASnombreinstancia/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview

- HTML

-  unidad:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombearchivadortienda\preview\entorno\_nacional
-  unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombearchivadortienda\preview\entorno\_nacional
-  /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview/entorno\_nacional
-  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview/entorno\_nacional
-  /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview/entorno\_nacional
-  /Qibm/UserData/WebAsAdv4/WASnombreinstancia/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombearchivadortienda/preview/entorno\_nacional

- imágenes independientes del entorno nacional

- **NT** unidad:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombreachivadortienda\preview\images
  - **2000** unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombreachivadortienda\preview\images
  - **AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/images
  - **Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/images
  - **Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/images
  - **400** /Qibm/UserData/WebAsAdv4/WASnombreinstancia/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/images
- imágenes dependientes del entorno nacional
    - **NT** unidad:\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombreachivadortienda\preview\entorno\_nacional\images
    - **2000** unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC\_Enterprise\_App\_nombreinstancia.ear\wcstools.war\tools\devtools\nombreachivadortienda\preview\entorno\_nacional\images
    - **AIX** /usr/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/entorno\_nacional/images
    - **Solaris** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/entorno\_nacional/images
    - **Linux** /opt/WebSphere/AppServer/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/entorno\_nacional/images
    - **400** /Qibm/UserData/WebAsAdv4/WASnombreinstancia/installedApps/WC\_Enterprise\_App\_nombreinstancia.ear/wcstools.war/tools/devtools/nombreachivadortienda/preview/entorno\_nacional/images
- e. Dado que la ubicación de las imágenes y del archivo css se ha cambiado, deberá cambiar las referencias a las imágenes y al archivo css en las páginas HTML. Después de cambiar las referencias, compruebe que se pueden ver las imágenes cuando se abren las página HTML en un navegador.

- f. Cambie los enlaces en las páginas HTML de mandatos a enlaces que hagan referencia a archivos HTML.
3. Cree un archivo HTML que resuma el archivador de tienda. Esta información aparecerá en la **Descripción de ejemplo** de la página Crear archivador de tienda de los Servicios de tienda.
- a. Utilice el ejemplo siguiente como guía y cree el archivo nuevo.
- ```
<doctype html public "-//w3c//dtd html 4.0 transitional//en">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
  Describa aquí la tienda
</body>
</html>
```
- b. Guarde este archivo como `Feature_entorno_nacional.html`, donde `entorno_nacional` es la abreviatura del idioma que está utilizando. Por ejemplo, `es_ES`. Guarde este archivo en el directorio siguiente:
-  `unidad:\WebSphere\CommerceServer\samplestores\nombearchivadortienda`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\samplestores\nombearchivadortienda`
 -  `/usr/WebSphere/CommerceServer/samplestores/nombearchivadortienda`
 -  `/opt/WebSphere/CommerceServer/samplestores/nombearchivadortienda`
 -  `/opt/WebSphere/CommerceServer/samplestores/nombearchivadortienda`
 -  `/qibm/proddata/WebCommerce/samplestores/nombearchivadortienda`
4. Añada el archivador de tienda al archivo `sarregistry.xml`. El archivo `sarregistry.xml` determina qué archivadores de tienda se muestran en la lista **Ejemplo** de la página Crear archivador de tienda de los Servicios de tienda. El archivo `sarregistry.xml` determina también qué páginas de vista previa y qué archivo de características están asociados con cada archivador de tienda. El archivo `sarregistry.xml` se encuentra en el directorio siguiente:
-  `unidad:\WebSphere\CommerceServer\xml\tools\devtools`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\tools\devtools`
 -  `/usr/WebSphere/CommerceServer/xml/tools/devtools`
 -  `/opt/WebSphere/CommerceServer/xml/tools/devtools`
 -  `/opt/WebSphere/CommerceServer/xml/tools/devtools`
 -  `/QIBM/ProdData/WebCommerce/xml/tools/devtools`
- a. Utilice el ejemplo siguiente como guía y añada el nuevo archivador de tienda a `sarregistry.xml`.


```
<SampleSAR fileName="infashion_en_US_es_ES.sar" relativePath="InFashion">  
  <html locale="es_ES" featureFile="InFashion/Feature_es_ES.html"  
    sampleSite="RetailModel/preview/es_ES/index.html"/>  
  <html locale="en_US" featureFile="InFashion/Feature_en_US.html"  
    sampleSite="RetailModel/preview/en_US/index.html"/>  
</SampleSAR>
```

Este ejemplo define las páginas de vista previa de los archivadores de la tienda InFashion en español y en inglés. Las líneas en negrita definen la página de presentación de las páginas de vista previa en inglés y en español.

5. Ahora deberá poder ver el archivador de tienda en la lista **Ejemplo** de la página Crear archivador de tienda de los Servicios de tienda.

Parte 7. Publicación de la tienda

Para crear una tienda que funcione, los elementos Web del escaparate deben publicarse en WebSphere Commerce Server y los datos de la tienda deben publicarse en la base de datos de WebSphere Commerce.

Los capítulos de esta sección describen las opciones de publicación que proporciona WebSphere Commerce:

- Capítulo 26, “Publicación de una tienda completa” en la página 227 - Este capítulo describe la publicación de una tienda entera (elementos de datos de tienda y de escaparate), si la tienda está en forma de archivador de tienda, utilizando la publicación de los Servicios de tienda o de la línea de mandatos.
- Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241 - Este capítulo describe la publicación de los elementos de datos de tienda en la base de datos utilizando Loader package y otros componentes de WebSphere Catalog Manager.
- Capítulo 28, “Carga de grupos de elementos de base de datos de WebSphere Commerce” en la página 279 - Este capítulo describe la publicación de grupos de elementos de datos de tienda o de todos los datos de tienda en la base de datos utilizando Loader package y otros componentes de WebSphere Catalog Manager.
- Capítulo 29, “Publicación de cuentas de negocio y contratos” en la página 293 - Este capítulo describe la publicación de elementos de cuenta, contrato y conjunto de productos.
- Capítulo 30, “Publicación de elementos de escaparate y de archivos de configuración de tienda” en la página 297 - Este capítulo describe la publicación de los elementos de escaparate y los archivos de configuración de la tienda.

Capítulo 26. Publicación de una tienda completa

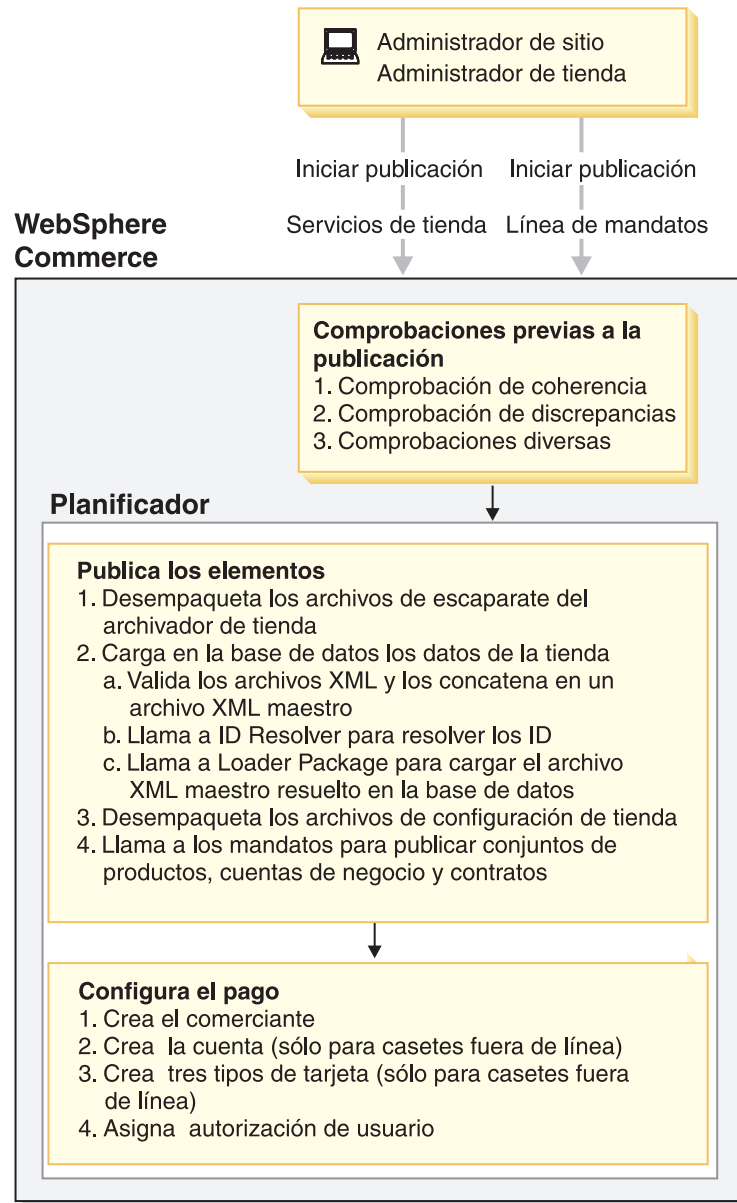
Para crear una tienda que funcione, los elementos Web del escaparate deben publicarse en WebSphere Commerce Server y los datos de la tienda deben publicarse en la base de datos de WebSphere Commerce. Este capítulo describe la publicación de una tienda entera (elementos de escaparate y de datos de tienda), si la tienda está en forma de archivador de tienda, utilizando la publicación mediante los Servicios de tienda o la línea de mandatos.

Nota: Si prefiere no empaquetar la tienda como un archivador de tienda, puede publicar los elementos individualmente. Para obtener más información, consulte el Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241, el Capítulo 28, “Carga de grupos de elementos de base de datos de WebSphere Commerce” en la página 279, el Capítulo 29, “Publicación de cuentas de negocio y contratos” en la página 293 y el Capítulo 30, “Publicación de elementos de escaparate y de archivos de configuración de tienda” en la página 297.

Descripción de la publicación en WebSphere Commerce

La opción de publicación que está disponible en los Servicios de tienda o en la línea de mandatos le permite publicar una tienda completa (elementos de escaparate y de datos de tienda) de forma inmediata. Para utilizar esta opción, los elementos de tienda deben estar empaquetados en forma de archivador de tienda. Para obtener más información sobre cómo empaquetar la tienda como un archivador de tienda, consulte la Parte 6, “Empaquetado de la tienda” en la página 215.

El diagrama siguiente describe los pasos del proceso de publicación.



Iniciar la publicación

Para publicar una tienda, debe tener autorización de Administrador de sitio o de Administrador de tienda (para todas las tiendas). Los Administradores de sitio o Administradores de tienda pueden iniciar el proceso de publicación utilizando uno de los métodos siguientes:

- Servicios de tienda
- Línea de mandatos

Ambos métodos de publicación requieren que se seleccione el archivador de tienda que se desea publicar. Utilizando cualquiera de los métodos, puede seleccionar qué partes del archivador de tienda desea publicar. Por ejemplo, puede seleccionar publicar cualquier combinación de lo siguiente:

- Elementos de base de datos de tienda, con o sin los datos del catálogo en línea
- Elementos Web tales como archivos JSP, archivos HTML e imágenes

- Archivos de propiedades (el texto para la tienda)

La primera vez que se publique, se recomienda publicar el archivador de tienda entero (todo lo anterior), a fin de poder ver una tienda en funcionamiento. Sin embargo, en las publicaciones subsiguientes, puede que sólo desee actualizar uno de los elementos siguientes: los elementos de base de datos, los elementos Web o los paquetes de recursos de propiedades, en lugar de volver a publicar el archivador de tienda entero.

Si elige no publicar los datos del catálogo en línea la primera vez que publique un archivador de tienda, puede que algunos de los elementos de datos (contratos y cuentas) no se publiquen correctamente porque dependen de datos del catálogo. Asimismo, si no publica los datos del catálogo y los elementos Web la primera vez que publique, el ID de tienda, el ID de catálogo y el ID de idioma no se crearán, y no podrá iniciar la tienda desde los Servicios de tienda. Para obtener más información, consulte el apartado “Crea el archivo parameters.jsp” en la página 237.

Los datos del catálogo en línea se componen de los archivos XML siguientes del archivador de tienda:

- catalog.xml
- offering.xml
- store-catalog.xml
- store-catalog-shipping.xml
- store-catalog-tax.xml
- storefulfill.xml

Si elige publicar esta información en una fecha posterior, puede publicarla utilizando los Servicios de tienda o mediante Loader package. Para obtener más información sobre cómo cargar un subconjunto de datos utilizando los grupos de elementos de base de datos, por ejemplo los datos de grupo de catálogo listados a continuación, consulte el apartado “Carga de grupos de elementos de base de datos” en la página 287.

Para obtener información más detallada sobre cómo publicar un archivador de tienda utilizando los Servicios de tienda o la línea de mandatos, consulte la ayuda en línea de WebSphere Commerce.

Nota: Después de haber iniciado el proceso de publicación utilizando los Servicios de tienda o la línea de mandatos, no tiene que hacer nada más. Todos los demás pasos listados en el diagrama anterior y en este capítulo los realiza el sistema WebSphere Commerce.

Comprobaciones previas a la publicación

Una vez que el Administrador de sitio o de tienda ha iniciado la publicación, WebSphere Commerce realiza varias comprobaciones antes de empezar el proceso de publicación real. Estas comprobaciones incluyen lo siguiente:

- Comprobación de coherencia
- Comprobación de discrepancias
- Comprobaciones diversas

Nota: La publicación por línea de mandatos comprueba los parámetros que se le pasan, y completa las comprobaciones de coherencia y discrepancia. Sin embargo, si publica mediante la línea de mandatos, no verá los mensajes

descritos en los siguientes párrafos. En su lugar, si publica utilizando la modalidad de actualización, la línea de mandatos graba encima de la tienda existente sin emitir mensajes de aviso. Si publica utilizando el método de inserción, la línea de mandatos carga la nueva tienda. Si la publicación no es satisfactoria, se muestra un mensaje de error.

Comprobación de coherencia

La comprobación previa a la publicación utiliza las normas del archivo `sarrule.xml` para asegurar que la información de los archivos XML es coherente con los elementos Web del archivador de tienda. Por ejemplo, si el archivo `command.xml` hace referencia a un archivo JSP determinado, la comprobación asegura que el archivo JSP esté en `webapp.zip` del archivador de tienda. Si la comprobación de coherencia encuentra un error, el error se grabará en la anotación cronológica, pero la publicación continuará normalmente. Para obtener más información sobre los archivos de anotaciones cronológicas, consulte el apartado “Archivos de anotaciones cronológicas de la publicación” en la página 238.

Si desea ver un ejemplo de un archivo `sarrule.xml`, consulte el Apéndice D, “`sarrule.xml`” en la página 335.

Comprobación de discrepancias

Durante la comprobación previa a la publicación, WebSphere Commerce comprueba las discrepancias entre la tienda y el catálogo. En particular, comprueba lo siguiente en la base de datos:

- Una tienda existente con el mismo identificador en la tabla STOREENT: Si la tienda ya existe, se visualiza un mensaje solicitando si desea grabar encima de la tienda o cancelar la publicación.
- Un catálogo existente con el mismo identificador de catálogo: Si el catálogo ya existe, se visualiza un mensaje solicitando si desea grabar encima del catálogo o cancelar la publicación. Si el catálogo pertenece a otras tiendas, se visualiza un mensaje que indica a qué tiendas pertenece el catálogo y que solicita si desea continuar la publicación y grabar encima del catálogo o cancelar la publicación.


Comprobaciones diversas

Durante la comprobación previa a la publicación, WebSphere Commerce también comprueba si el planificador está habilitado, si los desencadenantes de antememoria y la antememoria están habilitados y si las tablas de resumen están inhabilitadas. Si no lo están, se visualiza un mensaje de aviso, indicándole qué característica está habilitada o inhabilitada de forma incorrecta.

Nota: La publicación mediante la línea de mandatos no lleva a cabo estas comprobaciones.

Planificador: Tal como se describirá más detalladamente en las secciones siguientes, el planificador ejecuta el trabajo de publicación. Si el planificador está inhabilitado, el proceso de publicación no podrá ejecutarse.

Antememoria y desencadenantes de antememoria: Si se deja activada la antememoria durante la publicación, puede que se invoquen los desencadenantes de antememoria durante la publicación, cuando se actualiza la base de datos. Los desencadenantes de antememoria pueden generar una actividad innecesaria de la base de datos que puede producir un desbordamiento de las anotaciones de las transacciones de la base de datos y afectar el rendimiento de la publicación. Para inhabilitar los desencadenantes de antememoria, consulte la ayuda en línea de WebSphere Commerce.

Tablas de resumen:  Si se dejan las tablas de resumen habilitadas, puede que dichas tablas se actualicen durante la publicación, lo que puede producir un desbordamiento de las anotaciones de las transacciones de la base de datos y afectar el rendimiento de la publicación. Para inhabilitar las tablas de resumen, consulte la ayuda en línea de WebSphere Commerce.

Publicar elementos






La fase de publicación de elementos del proceso de publicación es un trabajo que ejecuta el planificador. Cuando el planificador ejecuta el trabajo de publicación, WebSphere Commerce realiza las acciones siguientes:

- Desempaqueta los archivos de escaparate del archivador de tienda
- Carga en la base de datos los datos de tienda de los archivos XML del archivador de tienda
- Crea el archivo `parameters.jsp`
- Desempaqueta los archivos de configuración de tienda
- Llama a los mandatos para publicar cuentas de negocio y contratos
- Actualiza los componentes de registro

Desempaqueta los archivos de escaparate del archivador de tienda

El desempaquetado de los elementos Web del archivador de tienda en WebSphere Commerce Server es la primera acción de la fase de publicación de elementos. Al desempaquetar los archivos del archivador de tienda, WebSphere Commerce realiza lo siguiente:

Desempaqueta los elementos Web y los copia en las ubicaciones siguientes de WebSphere Commerce Server:

- Los archivos JSP, HTML, los archivos de inclusión, las imágenes y los gráficos se publican en el directorio de tienda (*dir tienda*) bajo el directorio raíz de documentos de la aplicación Web de tiendas:
 -  unidad: \WebSphere\AppServer\installedApps\
WC_Enterprise_App_nombreinstancia.ear\wcstores.war\dir tienda
 -  unidad: \Archivos de programa\WebSphere\AppServer\
installedApps\WC_Enterprise_App_nombreinstancia.ear\wcstores.war\
dir tienda
 -  /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
 -  /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
 -  /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
 -  /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
- Los paquetes de recursos y archivos de propiedades se publican en la vía de acceso de propiedades de la aplicación:
 -  unidad: \WebSphere\AppServer\installedApps\
WC_Enterprise_App_nombreinstancia.ear\wcstores.war\WEB-
INF\classes\dir tienda

- ▶ 2000 unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC_Enterprise_App_nombreinstancia.ear\wcstores.war\WEB-INF\classes\dirtienda
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
- ▶ Linux /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
- ▶ 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda

Carga en la base de datos los datos de tienda de los archivos XML del archivador de tienda

Para cargar en la base de datos los datos de tienda de los archivos XML del archivador de tienda, WebSphere Commerce realiza lo siguiente:





Nota: Sólo se cargan en la base de datos los archivos XML de tipo db-load. El tipo de archivo se especifica en el archivo sarinfo.xml. Para obtener más información sobre el archivo sarinfo.xml, consulte el Apéndice C, “sarinfo.xml” en la página 329

Valida los archivos XML del archivador de tienda y los concatena en un archivo XML maestro: WebSphere Commerce valida los archivos XML utilizando los archivos DTD correspondientes. Los archivos DTD se encuentran en el directorio siguiente:






- ▶ NT unidad:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

Si los archivos XML no son válidos, WebSphere Commerce graba un error en la anotación cronológica de errores. Si los errores de la anotación cronológica de errores exceden el número máximo de errores especificados en la parte DevTools del archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, (por omisión MaxErrorsInSarXML=1) la publicación fallará. El archivo de configuración de WebSphere Commerce, *nombre_instancia.xml* se encuentra en el siguiente directorio:

- ▶ NT unidad:\WebSphere\CommerceServer\instances\nombreinstancia.xml\nombre_instancia.xml
- ▶ 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\instances\nombreinstancia.xml\nombre_instancia.xml

-  /usr/WebSphere/CommerceServer/instances/*nombreinstancia* /xml/*nombre_instancia.xml*
-  /opt/WebSphere/CommerceServer/instances/*nombreinstancia* /xml/*nombre_instancia.xml*
-  /opt/WebSphere/CommerceServer/instances/*nombreinstancia* /xml/*nombre_instancia.xml*
-  /QIBM/UserData/WebCommerce/instances/*nombreinstancia* /xml/*nombre_instancia.xml*

Después de que los archivos XML se hayan validado, se concatenan en un archivo: *nombretiendamaster.xml*. Los archivos se concatenan según la prioridad especificada en el archivo *sarinfo.xml*. Para obtener más información, consulte el Apéndice C, “*sarinfo.xml*” en la página 329. El archivo *nombretiendamaster.xml* está ubicado en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\temp*nombreinstancia*\tools\devtools
-  unidad:\Archivos de programa\WebSphere\CommerceServer\temp*nombreinstancia*\tools\devtools
-  /usr/WebSphere/CommerceServer/temp/*nombreinstancia* /tools/devtools
-  /opt/WebSphere/CommerceServer/temp/*nombreinstancia* /tools/devtools
-  /opt/WebSphere/CommerceServer/temp/*nombreinstancia* /tools/devtools
-  /QIBM/UserData/WebCommerce/instances/*nombreinstancia* /temp/tools/devtools

Llama a ID Resolver para resolver los ID: ID Resolver, que es un programa de utilidad de Loader package, genera identificadores exclusivos para elementos XML de los archivos XML del archivador de tienda. Por ejemplo, ID Resolver sustituye el alias @ utilizado en los archivos XML de la tienda de ejemplo por un valor exclusivo. Para obtener un ejemplo de la resolución de los alias internos utilizados en las tiendas de ejemplo, consulte el Apéndice B, “Creación de los datos” en la página 327.

Nota: ID Resolver también puede resolver identificadores para tiendas ya publicadas, cuando las vuelva a publicar. Por ejemplo, si ha publicado el archivador de tienda una vez y tiene que volver a publicarlo, todo o partes del mismo, ID Resolver recupera los identificadores exclusivos de la base de datos y los utiliza cuando vuelve a publicar.

Para obtener más información sobre ID Resolver y los demás componentes de Loader package, consulte el Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241.

Cuando la publicación mediante los Servicios de tienda o la línea de mandatos llama a ID Resolver, debe especificar qué método de ID Resolver se debe utilizar. ID Resolver tiene varios métodos, que se pueden utilizar para procesar la entrada de ID Resolver; específicamente consisten en tratar los datos como si existieran identificadores en los datos originales (método de actualización) o tratar dichos

datos como si no existieran identificadores en los datos originales (método de carga). El método combinado se utiliza cuando algunos identificadores existen y otros no.

Nota: El método combinado es el método recomendado para los Servicios de tienda. Si publica utilizando la línea de mandatos, quizá también desee especificar la modalidad combinada, particularmente si no sabe si los datos que se publican ya están en la base de datos.

Puede especificar qué método utilizará la publicación mediante los Servicios de tienda o la línea de mandatos en el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*. Por omisión, los Servicios de tienda utilizan el método combinado. Para obtener más información sobre los métodos de ID Resolver, consulte el apartado "Mandato ID Resolve" en la página 245.

La publicación mediante los Servicios de tienda y la línea de mandatos también debe especificar un archivo de personalización para utilizarlo con ID Resolver. Si no especifica ningún archivo de personalización en el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, el código de publicación utilizará uno de los archivos de personalización por omisión: DBConnectionCustomizer u OracleConnectionCustomizer.

Oracle

El archivo de personalización OracleConnectionCustomizer se encuentra en el siguiente directorio:

- ▶ NT unidad:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nombreinstancia.ear\properties
- ▶ 2000 unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC_Enterprise_App_nombreinstancia.ear\properties
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/properties
- ▶ Solaris /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/properties
- ▶ Linux /opt/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/properties
- ▶ 400 /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/WC_Enterprise_App_nombreinstancia.ear/properties

DB2

El archivo DBConnectionCustomizer se encuentra en el siguiente archivo ZIP:

- ▶ NT unidad:\WebSphere\AppServer\installedApps\WC_Enterprise_App_nombreinstancia.ear\lib\loader\idresgen.zip
- ▶ 2000 unidad:\Archivos de programa\WebSphere\AppServer\installedApps\WC_Enterprise_App_nombreinstancia.ear\lib\loader\idresgen.zip
- ▶ AIX /usr/WebSphere/AppServer/installedApps/WC_Enterprise_App_nombreinstancia.ear/lib/loader/idresgen.zip

- **Solaris** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/lib/
loader/idresgen.zip
- **Linux** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/lib/
loader/idresgen.zip
- **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_nombreinstancia.ear/lib/
loader/idresgen.zip

Nota: Si desea especificar su propio archivo de personalización, deberá añadir lo siguiente en la sección DevTools del archivo *nombre_instancia.xml*:

- IDResolverCustomizerFile="myIDResolverCustomizerFile"

Por omisión, el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, no especifica un valor para este atributo.

ID Resolver utiliza el archivo *nombretiendamaster.xml* y el archivo DTD correspondiente, *wcs.dtd*. Después de resolver los ID, ID Resolver crea el archivo siguiente, *nombretiendaindicación_horamaster.xml*, que contiene los identificadores exclusivos. Si se produce un error durante el proceso de resolución de ID, Loader package crea un archivo *error.xml*, *nombretiendamaster.error.xml*.

Nota: El proceso de publicación guarda automáticamente estos archivos temporales cuando falla un intento de publicación. Sin embargo, si una publicación se realiza satisfactoriamente, estos archivos se suprimen por omisión. Puede que desee conservar estos archivos para la resolución de problemas o para publicar y trabajar con una tienda en el entorno de prueba de WebSphere. Para guardar los archivos temporales, consulte "Parámetros de Servicios de tienda" en la ayuda en línea de WebSphere Commerce.

El archivo *nombretiendaindicación_horamaster.xml* y el archivo *nombretiendamaster.error.xml* están ubicados en el directorio siguiente:

- **NT** unidad:\WebSphere\CommerceServer\temp*nombreinstancia*\tools\devtools
- **2000** unidad:\Archivos de programa\WebSphere\CommerceServer\temp*nombreinstancia*\tools\devtools
- **AIX** /usr/WebSphere/CommerceServer/temp/*nombreinstancia*/tools/devtools
- **Solaris** /opt/WebSphere/CommerceServer/temp/*nombreinstancia*/tools/devtools
- **Linux** /opt/WebSphere/CommerceServer/temp/*nombreinstancia*/tools/devtools
- **400** /QIBM/UserData/WebCommerce/instances/*nombreinstancia*/temp/tools/devtools

Llama a Loader package para cargar el archivo XML maestro resuelto en la base de datos: Loader package carga el archivo *nombretiendaindicación_horamaster.xml* resuelto en la base de datos. Si se produce un error durante el proceso de carga, Loader package crea un archivo *error.xml*, *nombretiendaindicación_horamaster.error.xml*.

Para obtener más información sobre Loader package, consulte el Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241.

Cuando la publicación mediante los Servicios de tienda o la línea de mandatos llama a Loader package, debe especificar qué método de Loader se debe utilizar. Los Servicios de tienda pueden utilizar los métodos siguientes de Loader:

- SQL import (Importación de SQL)
- Import (Importación)
- Load (Carga)

Nota: Por omisión, los Servicios de tienda utilizan el método de importación de SQL.

Puede especificar qué método deberá llamar la publicación de los Servicios de tienda o de la línea de mandatos en el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, utilizando el atributo LoaderMode en el elemento DevTools.

- SQL import: Este método utiliza JDBC (Java Database Connectivity) para insertar y actualizar datos, proporcionando el método más flexible de operación pero también el más lento para importar grandes cantidades de datos a un pequeño número de tablas. Permite la actualización a nivel de columna. Se recomienda utilizar SQL import.

Nota: El método de importación de SQL es el método más seguro porque no corromperá la base de datos si los datos no son válidos. Para poder efectuar la carga utilizando la importación de SQL, los registros deben satisfacer las restricciones de esquema de base de datos. Los otros métodos de Loader son más rápidos porque los datos se cargan masivamente en la base de datos sin efectuar muchas comprobaciones. Por tanto, para utilizar los otros métodos debe estar seguro de que los datos sean correctos.

- Import: Este método utiliza funciones de importación nativas de DB2 y permite la actualización a nivel de casilla con una velocidad y flexibilidad medias. Este método no está disponible con Oracle.
- Load: Este método utiliza los recursos nativos de RDBMS (DB2 Load o SQLLoad) y es el método más rápido para cargar grandes cantidades de datos en un pequeño número de tablas.

Para obtener más información sobre los métodos del mandato load, consulte el apartado “Mandato Load” en la página 253.

La publicación mediante los Servicios de tienda y la línea de mandatos también debe especificar un archivo de personalización para utilizarlo con Loader. Si no especifica ningún archivo de personalización en el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, el código de publicación utilizará el archivo de personalización por omisión: MassLoadCustomizer.

Nota: Si desea especificar su propio archivo de personalización, deberá añadir lo siguiente en la sección DevTools del archivo *nombre_instancia.xml*:

- LoaderCustomizerFile="*miArchivoPersonalizaciónLoader*"

Por omisión, el archivo de configuración de WebSphere Commerce, *nombre_instancia.xml*, no especifica un valor para este atributo.

Crea el archivo parameters.jsp

El proceso de publicación crea el archivo `parameters.jsp`. Este archivo incluye tres parámetros: `storeId`, `catalogId` y `langId`. El archivo `index.jsp`, de las tiendas de ejemplo, utiliza estos parámetros para iniciar la tienda.

Para que el proceso de publicación cree el archivo `parameters.jsp`, los datos de catálogo y los elementos Web deben publicarse simultáneamente una vez como mínimo. Si estos dos elementos no se publican a la vez, los Servicios de tienda no podrán iniciar la tienda.

`parameters.jsp` está ubicado en el directorio siguiente:

- ▶ **NT** unidad:\WebSphere\AppServer\installedApps\
WC_Enterprise_App_nombreinstancia.ear/wcstores.war\dirienda\include
- ▶ **2000** unidad:\Archivos de programa\WebSphere\AppServer\
installedApps\WC_Enterprise_App_nombreinstancia.ear/wcstores.war\
dirienda\include
- ▶ **AIX** /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dirienda/include
- ▶ **Solaris** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dirienda/include
- ▶ **Linux** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dirienda/include
- ▶ **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dirienda/include

Desempaqueta los archivos de configuración de tienda

Los archivadores de las tiendas de ejemplo **Business** ToolTech y NewFashion también incluyen los archivos siguientes:

- `tools_properties.zip`
- `tools_xml.zip`
- `runtime_xml.zip`

Estos archivos están registrados en el archivo `sarinfo.xml` y los utilizan los Servicios de tienda para configurar tiendas. Durante el proceso de publicación, estos archivos se desempaquetan y se copian en el directorio especificado en el archivo de configuración de WebSphere Commerce, `nombre_instancia.xml`. Para obtener más información, consulte el apartado “Publicación de elementos de escaparate y archivos de configuración de tienda copiándolos en WebSphere Commerce Server” en la página 298.

Nota: Estos archivos no deben cambiarse, eliminarse, copiarse a otras tiendas ni guardarse en un directorio diferente.

Llama a los mandatos para publicar cuentas de negocio y contratos

Loader package no puede cargar algunos de los elementos de base de datos de tienda (contratos y cuentas de negocio), de modo que la publicación también llama a los mandatos correspondientes para publicar dichos elementos en WebSphere Commerce Server. Estos mandatos son los siguientes:

- `AccountImport` — Crea una cuenta de negocio a partir del archivo `businessaccount.xml` del archivador de tienda.

- `ContractImportApprovedVersion` — Importa un contrato del archivo `contract.xml` del archivador de tienda.
- `ProductSetPublish` — Sincroniza los datos del conjunto de productos en la base de datos con el catálogo, antes de crear las cuentas de negocio y los contratos. La publicación a través de los Servicios de tienda o la línea de mandatos llama al mandato `ProductSetPublish` que, a su vez, llama a los mandatos `AccountImport` y `ContractImportApprovedVersion`.

Para obtener más información sobre la publicación de cuentas de negocio y de contratos, consulte el Capítulo 29, “Publicación de cuentas de negocio y contratos” en la página 293.

Actualiza componentes de registro

La última acción del proceso de publicación es actualizar los componentes de registro. El proceso de publicación actualiza todos los registros de WebSphere Commerce. Para obtener más información sobre los registros, consulte la ayuda en línea de WebSphere Commerce.

Manejo de errores

Si se produce un error durante la fase de publicación de elementos del proceso de publicación, puede ver el mensaje de error en las anotaciones cronológicas de publicación (consulte el apartado “Archivos de anotaciones cronológicas de la publicación”) o mediante la página Resumen de la publicación de los Servicios de tienda.

Configurar el pago

El último paso del proceso de publicación es configurar el pago. WebSphere Commerce da soporte a IBM Payment Manager. Si piensa utilizar Payment Manager como método de proceso de pago, deberá crear un archivo XML de pago tal como se describe en el Capítulo 13, “Elementos de pago” en la página 121. Si se incluye un archivo XML de pago en el archivador de tienda que se está publicando, WebSphere Commerce realizará la configuración de pago siguiente durante la publicación:

- Creará el comerciante.
- Creará la cuenta (sólo para casetes fuera de línea).
- Creará las marcas especificadas en `paymentinfo.xml` (sólo para casetes fuera de línea).
- Asignará autorización de usuario.

Manejo de errores

Si se produce un error durante la fase de configuración de pago del proceso de publicación, puede ver el mensaje de error en las anotaciones cronológicas de publicación (consulte el apartado “Archivos de anotaciones cronológicas de la publicación”).







Archivos de anotaciones cronológicas de la publicación

Los errores encontrados durante la fase de publicación de elementos del proceso de publicación se graban en los archivos de anotaciones cronológicas y de rastreo siguientes:







- `messages.txt`: Contiene mensajes de error de la parte de Loader package del proceso de publicación. Cuando falle la publicación, compruebe primero esta anotación cronológica. Los números de línea o columna mencionados en estos mensajes de error hacen referencia a los archivos `master.xml` temporales: `nombretiendamaster.xml` o `nombretiendaindicación_horamaster.xml`.

- `trace.txt`: Contiene información de rastreo para las partes de Loader package e ID Resolver del proceso de publicación. Por omisión, `trace.txt` está desactivado.
- `ecmsg_nombreinstancia_indicaciónhora.log`: Anota todos los mensajes de error de ejecución de WebSphere Commerce Server.
- `wcs.log`: Contiene la salida de todas las aplicaciones (incluida la publicación) que se ejecutan en WebSphere Application Server que se produce en la consola.

Los archivos de anotaciones cronológicas están ubicados en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\instances\nombreinstancia\logs`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\nombreinstancia\logs`
-  `/usr/WebSphere/CommerceServer/instances/nombreinstancia/logs`
-  `/opt/WebSphere/CommerceServer/instances/nombreinstancia/logs`
-  `/opt/WebSphere/CommerceServer/instances/nombreinstancia/logs`
-  `/QIBM/UserData/WebCommerce/instances/nombreinstancia/logs`

Para configurar los archivos de anotaciones cronológicas `trace.txt` y `messages.txt` (es decir, ajustar el nivel de anotación cronológica u otras opciones), edite el archivo siguiente:

-  `unidad:\WebSphere\CommerceServer\xml\loader\WCALoggerConfig.xml`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\loader\WCALoggerConfig.xml`
-  `/usr/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml`
-  `/opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml`
-  `/opt/WebSphere/CommerceServer/xml/loader/WCALoggerConfig.xml`
-  `/QIBM/UserData/WebCommerce/instances/instancename/xml/WCALoggerConfig.xml`

Nota: Para obtener más información sobre la configuración del archivo `WCALoggerConfig.xml`, consulte la publicación *WebSphere Commerce, Guía del usuario de Catalog Manager*.

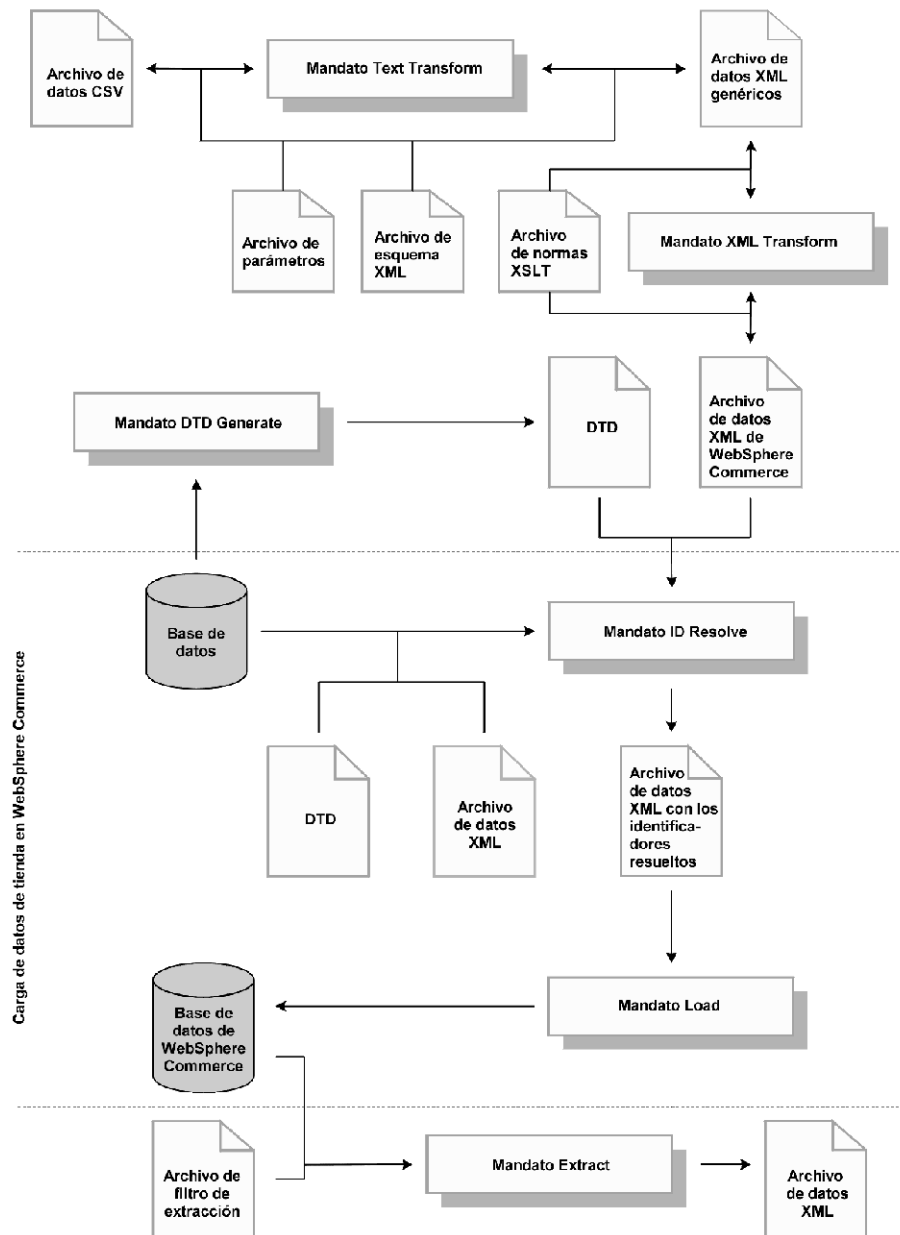
Capítulo 27. Visión general de la carga de datos de tienda

Después de crear los datos de la tienda, puede elegir empaquetarlos como un archivador de tienda y publicarlos utilizando los Servicios de tienda o puede cargarlos directamente en la base de datos de WebSphere Commerce Server utilizando Loader package de WebSphere Commerce Catalog Manager. Consulte el Capítulo 28, “Carga de grupos de elementos de base de datos de WebSphere Commerce” en la página 279 y el apartado “Carga de grupos de elementos de base de datos” en la página 287 si desea obtener información sobre el proceso de carga para grupos de elementos de base de datos de WebSphere Commerce.

Catalog Manager proporciona seis programas de utilidad de línea de mandatos (que aquí se denominan conjuntamente “Loader package”) y tres herramientas administrativas relacionadas que se pueden utilizar para preparar los datos así como para cargarlos en la tienda. Estos mandatos y estas herramientas utilizan archivos de datos XML (Extensible Markup Language) para gestionar la información.

Descripción de la carga de datos en WebSphere Commerce

En la figura siguiente se muestran los procesos de preparación, carga y extracción de datos que puede realizar utilizando los mandatos de Loader package.



Observe que la línea de puntos indica los dos procesos que se utilizan más comúnmente para cargar datos de tienda en una base de datos de WebSphere Commerce Server: la resolución de los identificadores y la carga de los datos. Estos procesos son el tema central de este capítulo.

Para obtener más información sobre cómo preparar los datos para cargarlos en una base de datos de WebSphere Commerce Server, consulte la Parte 4, “Desarrollo de los datos de la tienda” en la página 37.

Los dos programas de utilidad de línea de mandatos siguientes de Loader package se utilizan normalmente para cargar datos en una base de datos de WebSphere Commerce Server:

- **Mandato ID Resolve**

Para cargar datos XML en una base de datos de WebSphere Commerce Server utilizando Loader package, los elementos XML deben correlacionarse directamente con el esquema de la base de datos de WebSphere Commerce Server de destino. Todos los elementos XML que tengan atributos correspondientes a claves exclusivas o primarias en el esquema de base de datos deben tener identificadores exclusivos; y todas las columnas que no puedan tener el valor nulo del esquema de base de datos deben tener atributos correspondientes definidos con valores no nulos. ID Resolver puede generar identificadores exclusivos para atributos de clave exclusiva o primaria de elementos XML calificadores.

Nota: En este documento, un identificador es un valor en una sola columna de una tabla de base de datos que da a cada fila una identidad exclusiva. Si utiliza ID Resolver para generar identificadores, obtiene un valor base de la tabla KEYS o SUBKEYS e incrementa el valor de forma secuencial para determinar un identificador para cada fila en la tabla de base de datos.

Para obtener información sobre este mandato, consulte los apartados “Mandato ID Resolve” en la página 245, “Utilización de los mandatos y scripts de Loader package” en la página 269 y “Ejemplos de resolución de identificadores” en la página 269.

- **Mandato Load**

Loader utiliza archivos XML válidos y con formato correcto como entrada para cargar datos en la base de datos. Los elementos del documento XML se correlacionan con los nombres de tabla de la base de datos y los atributos de elemento se correlacionan con las columnas.

Nota: Consulte las directrices XML de World Wide Web Consortium (W3C) si desea obtener una descripción de las restricciones en cuanto a validez y construcción correcta.

Para obtener información sobre este mandato, consulte los apartados “Mandato Load” en la página 253, “Utilización de los mandatos y scripts de Loader package” en la página 269 y “Ejemplo de carga de datos” en la página 276.

Estos mandatos son el tema principal de este capítulo.

Los siguientes programas de utilidad de línea de mandatos de Loader package también se pueden utilizar para gestionar los datos:

- **Mandato DTD Generate**

DTD Generator genera una definición de tipo de documento (DTD) que describe las tablas y las columnas de la base de datos de destino en la que se deben cargar los datos XML. DTD Generator también puede generar un esquema XML para la base de datos.

DTD Generator puede crear una DTD basándose en el esquema de base de datos de WebSphere Commerce. Si utiliza las DTD proporcionadas con los archivadores de tienda de ejemplo y no modifica el esquema de base de datos, normalmente no necesitará generar una DTD utilizando DTD Generator.

Consulte el apartado “Mandato DTD Generate” en la página 259 para obtener más información.

- **Mandato Extract**

Extractor utiliza una consulta en una base de datos para extraer subconjuntos de datos seleccionados de la base de datos en forma de documento XML.

Puede utilizar este mandato para extraer datos de la base de datos en formato XML.

Consulte el apartado “Mandato Extract” en la página 262 para obtener más información.

- **Mandato Text Transform**

Text Transformer transforma los datos de un formato variable delimitado por caracteres a un formato de datos XML.

Si los datos no se pueden extraer directamente de una base de datos en formato XML, por ejemplo, puede guardar los datos en formato variable delimitado por caracteres y, a continuación, utilizar este mandato para transformarlos a formato XML.

Consulte el apartado “Mandato Text Transform” en la página 265 para obtener más información.

- **Mandato XML Transform**

XML Transformer transforma los datos de un documento XML a un formato XML alternativo. Utiliza XSL (Extensible Stylesheet Language) para definir las normas de correlación para la transformación.

Puede utilizar este mandato para convertir los datos XML a un formato que se correlacione directamente con el esquema de la base de datos de WebSphere Commerce en la que desea cargar los datos.

Consulte el apartado “Mandato XML Transform” en la página 266 para obtener más información.

Estos mandatos no son el tema principal de este capítulo. Para obtener información detallada sobre estos mandatos, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

WebSphere Commerce Catalog Manager también incluye las herramientas siguientes para facilitar la administración de las funciones de gestión de datos:

- **Herramienta de Transformación de texto**

La herramienta de Transformación de texto le ayuda a procesar una transformación de datos entre un formato variable delimitado por caracteres y un formato de datos XML, utilizando el mandato Text Transform.

- **Editor XSL**

El editor XSL le proporciona una interfaz visual para editar archivos XSL que pueden ser utilizados por XML Transformer. Mediante la utilización del editor XSL, se establece la asociación de un elemento de una DTD fuente a un elemento de una DTD de destino al definir las normas de correlación para transformar datos entre formatos XML.

- **Editor Web**

El editor Web le permite crear, modificar y suprimir datos de una base de datos mediante un navegador Web.

Estas herramientas no son el tema principal de este capítulo. Para obtener información detallada sobre estas herramientas, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Mandatos de Loader package para cargar datos de tienda

Mandato ID Resolve

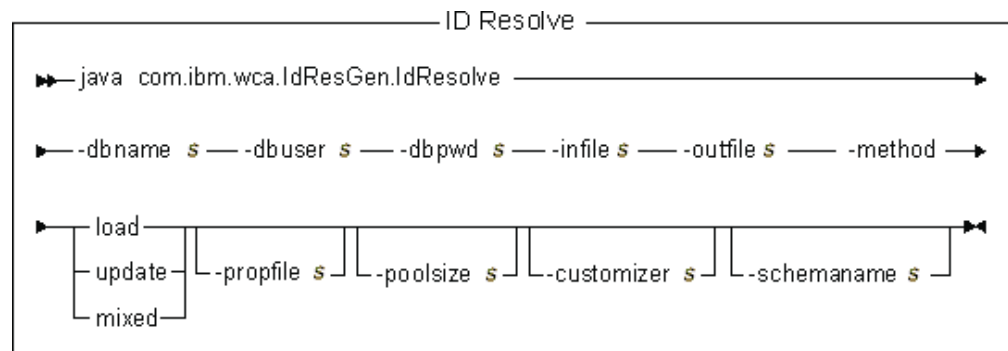
Este mandato genera identificadores exclusivos para los elementos de datos XML que los necesitan para que dichos elementos puedan cargarse en una base de datos. Si los datos XML fuente ya proporcionan los identificadores exclusivos necesarios, no tiene que ejecutar ID Resolver.

El esquema de base de datos de WebSphere Commerce define en las tablas las claves primarias y externas que se utilizan para representar diversas relaciones entre las tablas. Por este motivo, los elementos XML de WebSphere Commerce deben contener atributos que se correspondan con identificadores exclusivos. En la base de datos de WebSphere Commerce Server, las tablas cuyos identificadores deben resolverse son las definidas en las tablas KEYS y SUBKEYS. Estas tablas se denominan *tablas primarias* dentro de WebSphere Commerce. Para obtener más información sobre las tablas KEYS y SUBKEYS, consulte la ayuda en línea de WebSphere Commerce.

Nota: Si es necesario resolver los identificadores de una tabla que no está definida en la tabla KEYS o SUBKEYS, añada la tabla a la tabla SUBKEYS antes de ejecutar ID Resolver.

Dado que los atributos y elementos XML de WebSphere Commerce están diseñados para poder desplazarse entre bases de datos y entre instancias de base de datos, sus identificadores normalmente se representan utilizando alias internos. Para poder cargar los datos en cualquier base de datos de WebSphere Commerce Server, estos alias deben transformarse en identificadores numéricos válidos. Para obtener más información, consulte el Apéndice B, “Creación de los datos” en la página 327.

Windows AIX Solaris Linux



Notas:

1. El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.
2. Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

-dbname

Nombre de la base de datos de destino

-dbuser

Nombre del usuario que se conecta a la base de datos

-dbpwd

Contraseña del usuario que se conecta a la base de datos

-infile Nombre del documento XML de entrada que contiene los registros de tabla

-outfile

Nombre del archivo XML de salida que se genera; este archivo puede utilizarse como entrada a Loader

-method

Método a utilizar al procesar el archivo de entrada

- Utilice el método de carga para procesar el archivo de entrada si *no* existen *todos* los registros del archivo en la base de datos.
- Utilice el método de actualización para procesar el archivo de entrada si *existen todos* los registros del archivo en la base de datos.
- Utilice el método combinado para procesar el archivo de entrada si *sólo existen algunos* registros del archivo en la base de datos.

El método por omisión es el método de carga.

-profile

Archivo de texto que contiene las propiedades Java con el formato de parejas nombre=valor. Este archivo de propiedades establece la forma en que ID Resolver resuelve los identificadores. Se utiliza para describir qué columnas de una entrada primaria deben utilizarse como consultas para las tablas que necesitan el identificador de una fila primaria. Este archivo define los nombres de columna para búsquedas del identificador de clave externa y el predicado select en consultas de la tabla principal (por ejemplo, CATEGORY y PRODUCT). Puede omitir las entradas de este archivo para tablas que tengan definido un índice exclusivo que no incluya el identificador. Este parámetro es opcional. El archivo por omisión es `IdResolveKeys.properties`. Se puede especificar el archivo de propiedades como se muestra en cualquiera de los ejemplos siguientes:

```
-profile d:\WebSphere\CommerceServer\prop\idresprop.properties
```

```
-profile d:\WebSphere\CommerceServer\prop\idresprop
```

Si este archivo existe en el directorio actual, se puede especificar el mismo archivo tal como se muestra en el ejemplo siguiente:

```
-profile idresprop.properties
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
-profile idresprop
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades para utilizarlo con ID Resolver, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

-poolsize

Número de identificadores que se han de reservar. Este parámetro es opcional. El número por omisión es 50.

-customizer

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de ID Resolver.

DB2ConnectionCustomizer.properties es el archivo por omisión. Se puede especificar el archivo de propiedades de personalización como se muestra en los ejemplos siguientes:

```
-customizer d:\WebSphere\CommerceServer\prop\idres.properties
```

```
-customizer d:\WebSphere\CommerceServer\prop\idres
```

Si este archivo existe en el directorio actual, se puede especificar el mismo archivo tal como se muestra en el ejemplo siguiente:

```
-customizer idres.properties
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
-customizer idres
```

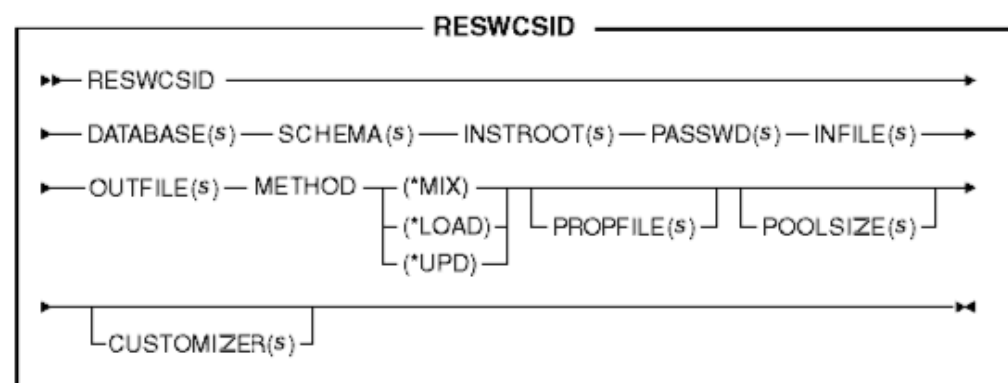
Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

-schemaname

Nombre del esquema de base de datos de destino. Este parámetro es opcional.

Si no se especifica este parámetro al ejecutar el mandato, éste busca una pareja nombre=valor en el archivo de propiedades de personalización que especifica el valor de SchemaName. Si esta pareja se encuentra en el archivo de propiedades, el mandato utiliza el valor especificado. Si no se especifica ni en el archivo de propiedades ni en la línea de mandatos, el mandato toma, por omisión, el nombre de esquema de la tabla en la base de datos.

▶ 400



Nota: Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

DATABASE

Nombre de la base de datos de destino, tal como aparece en el directorio de la base de datos relacional.

SCHEMA

Nombre del esquema de base de datos de destino; es igual que el nombre de la instancia.

INSTROOT

Nombre completo de la vía de acceso raíz de la instancia de WebSphere Commerce, por ejemplo, /QIBM/UserData/WebCommerce/instances
/nombre_instancia

PASSWD

Contraseña para la instancia de WebSphere Commerce

INFILE

Nombre del documento XML de entrada que contiene los registros de tabla

OUTFILE

Nombre del archivo XML de salida que se genera; este archivo puede utilizarse como entrada a Loader

METHOD

Método a utilizar al procesar el archivo de entrada

- Utilice el método de carga (*LOAD) para procesar el archivo de entrada si *no existen todos* los registros del archivo en la base de datos.
- Utilice el método de actualización (*UPD) para procesar el archivo de entrada si *existen todos* los registros del archivo en la base de datos.
- Utilice el método combinado (*MIX) para procesar el archivo de entrada si *sólo existen algunos* registros del archivo en la base de datos.

PROPROFILE

Archivo de texto que contiene las propiedades Java con el formato de parejas nombre=valor. Este archivo de propiedades establece la forma en que ID Resolver resuelve los identificadores. Se utiliza para describir qué columnas de una entrada primaria deben utilizarse como consultas para las tablas que necesitan el identificador de una fila primaria. Este archivo define los nombres de columna para búsquedas del identificador de clave externa y el predicado select en consultas de la tabla principal (por ejemplo, CATEGORY y PRODUCT). Puede omitir las entradas de este archivo para tablas que tengan definido un índice exclusivo que no incluya el identificador. Este parámetro es opcional. El archivo por omisión es `IdResolveKeys.properties`. Se puede especificar el archivo de propiedades como se muestra en cualquiera de los ejemplos siguientes:

```
PROPROFILE(/wc/prop/idresprop.properties)
```

```
PROPROFILE(/wc/prop/idresprop)
```

Si este archivo existe en el directorio actual, se puede especificar el mismo archivo tal como se muestra en el ejemplo siguiente:

```
PROPROFILE(idresprop.properties)
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
PROPROFILE(idresprop)
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades para utilizarlo con ID Resolver, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

POOLSIZE

Número de identificadores que se han de reservar. Este parámetro es opcional. El número por omisión es 50.

CUSTOMIZER

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de ID Resolver. El archivo por omisión es `ISeries_RESWCSID_Customizer.properties`. Se puede especificar el archivo de propiedades de personalización como se muestra en cualquiera de los ejemplos siguientes:

```
CUSTOMIZER(/wc/prop/idres.properties)
```

```
CUSTOMIZER(/wc/prop/idres)
```

Si este archivo existe en el directorio actual, se puede especificar el mismo archivo tal como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(idres.properties)
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema `CLASSPATH`, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(idres)
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Técnicas de resolución:

ID Resolver resuelve los identificadores utilizando una combinación de dos o tres de las técnicas siguientes, en función de si se utiliza un archivo de propiedades o no.

- **Resolución de alias interno**

Cuando se utiliza la resolución de ID de alias interno, se sustituye un alias por la clave (identificador) exclusiva en el documento XML fuente. Este alias se utiliza luego en cualquier otra parte del archivo XML para hacer referencia a ese elemento.

Los alias internos deben utilizarse de forma coherente en todo el archivo XML. Por ejemplo, si el alias de un ID de listín, `ADDRBOOK_ID`, es `@addrbook_1`, todas las referencias de clave externa a dicho ID del archivo deberán utilizar `@addrbook_1`.

Tenga en cuenta que los alias son transitorios para el archivo XML específico. No se guardan y no se pueden utilizar en un archivo XML independiente sin entrar el alias otra vez. Sin embargo, durante la publicación con los Servicios de tienda, la publicación concatena los archivos XML para que pueda producirse la resolución en todos los datos.

- **Resolución de índice exclusivo**

ID Resolver también puede analizar el esquema de base de datos para determinar si existe un índice exclusivo que satisfaga los requisitos. ID Resolver sólo busca un índice exclusivo cuando no hay ninguna entrada en el archivo de

propiedades para la tabla que se está analizando o cuando no hay ningún archivo de propiedades. Si estas condiciones son verdaderas, se realiza una comprobación de índice exclusivo. El índice exclusivo se considera válido si existe y no incluye la clave primaria para la tabla.

- **Especificación de archivo de propiedades**

ID Resolver le permite utilizar un archivo de propiedades Java alternativo para describir qué columnas de una entrada primaria deben utilizarse como consultas para las tablas que necesitan el identificador de una fila primaria.

Los archivadores de tienda de ejemplo proporcionados con WebSphere Commerce utilizan alias internos en sus archivos XML. Esto permite que los archivadores de tienda se puedan transportar entre bases de datos. Aunque las técnicas de especificación de índice exclusivo y de archivos de propiedades también permiten la portabilidad entre bases de datos, un usuario puede cambiar las columnas exclusivas y causar problemas cuando estas técnicas se utilicen posteriormente para la resolución de ID. Si, por ejemplo, un usuario cambia una columna exclusiva, el nombre de la columna debe cambiarse en la definición del archivo de propiedades. Sin embargo, con la técnica de alias interno, un cambio en la base de datos no requiere un cambio en los archivos XML o de propiedades. Durante la publicación con los Servicios de tienda o utilizando Loader package, ID Resolver sustituye el alias por un valor exclusivo. Una vez que se han cargado los datos, los alias son transparentes para el usuario. Para obtener más información, consulte el Apéndice B, “Creación de los datos” en la página 327.

ID Resolver utiliza el proceso siguiente:

- Si los datos XML de entrada tienen un elemento de una tabla primaria que ya tiene un identificador con codificación fija (por ejemplo “12345”), ID Resolver no crea un identificador nuevo para dicho elemento.
- Si los datos XML de entrada tienen un elemento de una tabla primaria que no tiene ningún identificador, ID Resolver examina la base de datos para ver si ya existe una fila para dicho elemento.

Para buscar el elemento en la base de datos es necesario que se utilicen otras columnas del elemento para formar una clave exclusiva. Estas otras columnas pueden especificarse en el archivo de propiedades o se puede permitir que ID Resolver determine qué columnas se deben utilizar.

- Si se está utilizando un archivo de propiedades y existe una entrada en el archivo de propiedades para la tabla que se está analizando, ID Resolver utiliza las columnas especificadas en el archivo de propiedades para formar la clave exclusiva.
- Si no se está utilizando ningún archivo de propiedades o no hay ninguna entrada en el archivo de propiedades para la tabla que se está analizando, ID Resolver utiliza la resolución de índice exclusivo.

La resolución de índice exclusivo utiliza cualquiera de los índices exclusivos especificados de una tabla como medio para localizar el identificador. Por ejemplo, MEMBER_ID más IDENTIFIER es un índice exclusivo de la tabla CATALOG y por consiguiente se puede utilizar como punto de resolución para la clave externa CATALOG_ID de la tabla CATALOGDSC.

Se considera que el elemento ya existe en la base de datos si hay una fila con la misma clave exclusiva; de lo contrario, se considera como una parte nueva de datos.

- Si el elemento ya existe en una fila de la base de datos, su identificador se recupera y se guarda para poderse utilizar posteriormente. De lo contrario, ID Resolver genera un identificador nuevo utilizando un valor disponible de la tabla KEYS o SUBKEYS.
- Si ha especificado un alias interno para el elemento (por ejemplo "@store_id_1") en el documento XML, dicho alias se asocia con el identificador para poder buscar ese identificador posteriormente utilizando el mismo alias interno.
- Los elementos de documento XML subsiguientes que necesitan hacer referencia a un elemento de la tabla primaria utilizan el alias interno, si el elemento de tabla primaria tenía alias interno (por ejemplo "@store_id_1") o utilizan los valores de las columnas de búsqueda si dicho elemento no tenía alias interno (por ejemplo "@WC2001@100"). En cualquiera de los casos, se utiliza el valor especificado para buscar el identificador real y el valor se sustituye por dicho identificador.
- Cuando se genera el documento XML de salida, todos los elementos de tabla primaria contienen identificadores reales y todos los elementos que hacen referencia a dichos elementos de tabla primaria hacen referencia a ellos utilizando los identificadores reales, no los alias internos ni los valores de columna de búsqueda mencionados anteriormente. Esto constituye el documento XML totalmente resuelto.

Métodos para el mandato ID Resolve:


El mandato ID Resolve le permite elegir entre el método de carga, el método de actualización o el método combinado para procesar el archivo de entrada.




Método de carga:


El método de carga para ID Resolver se utiliza para generar identificadores nuevos para todos los registros nuevos que se cargan en la base de datos.

Nota: Si especifica el método de carga para ID Resolver, los registros del archivo de entrada no deben existir aún en la base de datos. Si se utiliza el método de carga con ID Resolver y ya existe un registro del archivo XML fuente en la base de datos de destino, Loader generará un error al cargar los datos. ID Resolver asignará una clave primaria nueva al registro del archivo XML durante la resolución de ID; pero cuando cargue los datos en la base de datos, se generará un error. Loader no se detendrá en el momento de procesar el registro duplicado, pero informará de un error y el registro duplicado no se cargará en la base de datos.

El ejemplo siguiente se utiliza para generar identificadores para elementos de datos que son nuevos en la base de datos:

- 

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```
-   


```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method load -customizer customizer -schemaname wcsadmin
```
- 




```
QWEBCOMM/RESWCID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser)
PASSWD(mi contraseña) INFILE(input.xml) OUTFILE(output.xml)
METHOD(*LOAD)
```


Nota: Consulte el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269 para obtener información sobre la ubicación del mandato o script apropiados de ID Resolve.

Método de actualización:

Si especifica el método de actualización para ID Resolver, los registros del archivo de entrada ya deben existir en la base de datos. ID Resolver localiza los identificadores en la base de datos, tal como se describe en la página 250. Si no existe un registro en la base de datos, ID Resolver no puede resolver el identificador para este registro e indica que se ha producido un error. El ejemplo siguiente se utiliza para localizar identificadores para elementos de datos que ya existen en la base de datos:

- 

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```
-   


```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method update -customizer customizer -schemaname wcsadmin
```
- 




```
QWEBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mi contraseña)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*UPD)
```


Nota: Consulte el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269 para obtener información sobre la ubicación del mandato o script apropiados de ID Resolve.

Método combinado:

Si el archivo de datos de entrada contiene registros que ya existen en la base de datos así como algunos registros que son nuevos, se deberá ejecutar ID Resolver utilizando el método combinado. Con este método, ID Resolver crea identificadores nuevos para los registros sólo si los registros no existen en la base de datos. De lo contrario, se obtiene el identificador existente de la base de datos. El ejemplo siguiente se utiliza para generar identificadores para datos nuevos y para localizar identificadores para elementos de datos que ya existen en la base de datos:

- 

```
idresgen -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```
-   

```
./idresgen.sh -dbname db -dbuser user -dbpwd pwd -infile input.xml
-outfile output.xml -method mixed -customizer customizer -schemaname wcsadmin
```
- 

```
QWEBCOMM/RESWCSID DATABASE(DATABASE_NAME) SCHEMA(WCSADMIN)
INSTROOT(/QIBM/UserData/WebCommerce/instances/mser) PASSWD(mi contraseña)
INFILE(input.xml) OUTFILE(output.xml) METHOD(*MIX)
```

Notas:

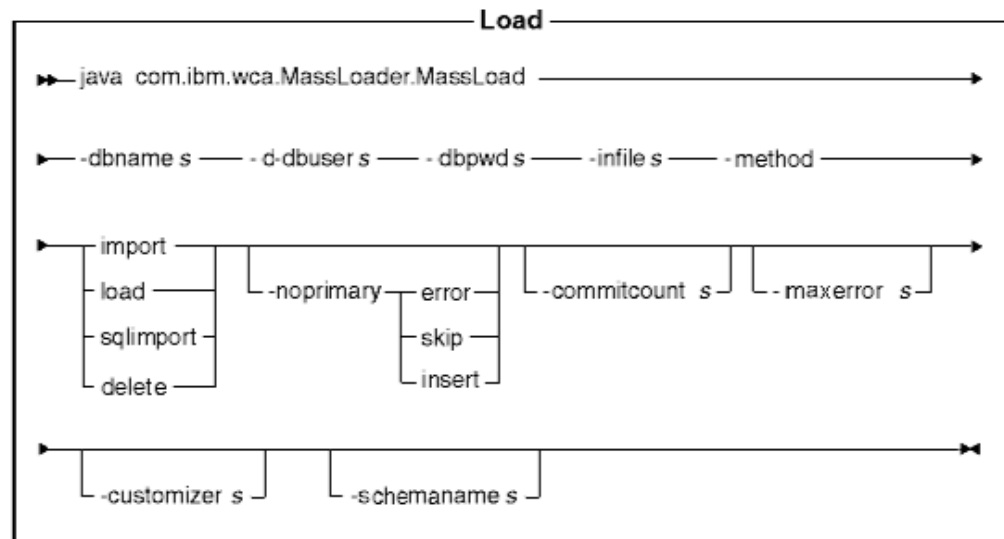
1. Consulte el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269 para obtener información sobre la ubicación del mandato o script apropiados de ID Resolve.
2. El método combinado es el método recomendado para los Servicios de tienda.

Si desea obtener información detallada sobre cómo configurar y personalizar los archivos utilizados para ejecutar este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Mandato Load

Este mandato carga un archivo de entrada XML en una base de datos de destino.

Windows AIX Solaris Linux



Notas:

1. El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado "Utilización de los mandatos y scripts de Loader package" en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.
2. Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

-dbname

Nombre de la base de datos de destino

-dbuser

Nombre del usuario que se conecta a la base de datos

-dbpwd

Contraseña del usuario que se conecta a la base de datos

-infile Nombre del archivo XML de entrada

-method

Modalidad de operación que deberá utilizar Loader al insertar datos en la base de datos

- El método de carga (load) utiliza el cargador nativo del proveedor de la base de datos. Puede utilizar el método de carga para bases de datos Oracle locales y remotas, pero el método de carga sólo se puede utilizar para bases de datos DB2 locales.

- Aunque el método de importación se puede utilizar para cargar datos en bases de datos locales o remotas, normalmente se utiliza para cargar datos en bases de datos DB2 remotas. Este método utiliza la opción de importar o actualizar si el proveedor de la base de datos la proporciona. Si especifica este método para una base de datos en la que la opción de importación o actualización no está disponible, como Oracle, se utilizan sentencias SQL que usan JDBC para actualizar la base de datos.
- El método de importación de SQL (sqlimport) se puede utilizar con bases de datos locales y remotas.
- El método de supresión (delete) suprime los datos de la base de datos.

-noprimary

Acción que Loader debe realizar cuando falte la clave primaria para un registro del archivo de entrada

- La opción "error" indica que se debe informar de que falta la clave primaria como un error y que se debe terminar.
- La opción "skip" se salta los registros del archivo de entrada que no tengan una clave primaria.
- La opción "insert" intenta insertar o suprimir los datos.

Este parámetro es opcional. La acción por omisión es "error".

-commitcount

Número de registros procesados antes de que se comprometa la base de datos cuando se utiliza el método de importación de SQL. Este parámetro es opcional. El número por omisión es 1.

-maxerror

Número de errores después de los cuales Loader finalizará el método de operación de importación de SQL. Este parámetro es opcional.

-customizer

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de Loader.

MassLoadCustomizer.properties es el archivo por omisión. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:

```
-customizer d:\WebSphere\CommerceServer\prop\ml.properties
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
-customizer ml
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

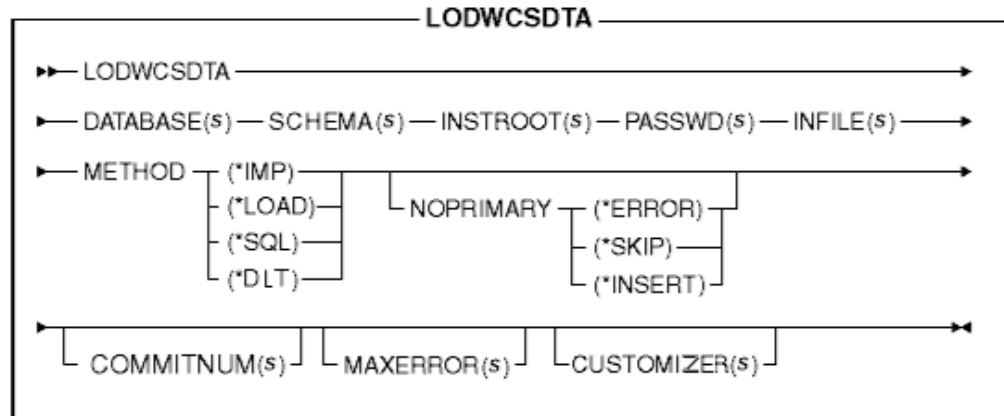
-schemaname

Nombre del esquema de base de datos de destino. Este parámetro es opcional.

Si no se especifica este parámetro al ejecutar el mandato, éste busca una pareja nombre=valor en el archivo de propiedades de personalización que especifica el valor de SchemaName. Si esta pareja se encuentra en el archivo de propiedades, el mandato utiliza el valor especificado. Si no se

especifica ni en el archivo de propiedades ni en la línea de mandatos, el mandato toma, por omisión, el nombre de esquema de la tabla en la base de datos.

▶ 400



Nota: Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

DATABASE

Nombre de la base de datos de destino tal como se visualiza en el directorio de base de datos relacional

SCHEMA

Nombre del esquema de base de datos de destino; es igual que el nombre de la instancia.

INSTROOT

Nombre completo de la vía de acceso raíz de la instancia de WebSphere Commerce, por ejemplo, /QIBM/UserData/WebCommerce/instancias /nombre_instancia

PASSWD

Contraseña para la instancia de WebSphere Commerce

INFILE

Nombre del archivo XML de entrada

METHOD

Modalidad de operación que deberá utilizar Loader al insertar datos en la base de datos

- El método de carga (*LOAD) utiliza el cargador nativo del proveedor de la base de datos. Puede utilizar el método de carga (*LOAD) para las bases de datos Oracle locales y remotas, aunque el método de carga (*LOAD) solamente se puede utilizar para las bases de datos DB2 locales.
- Aunque el método de importación (*IMP) se puede utilizar para cargar datos en bases de datos locales o remotas, normalmente se utiliza para cargar datos en bases de datos DB2 remotas. Este método utiliza la opción de importar o actualizar si el proveedor de la base de datos la

proporciona. Si no está disponible la opción de importar o actualizar, se utilizan las sentencias SQL mediante JDBC para actualizar la base de datos.

- El método de importación de SQL (*SQL) se puede utilizar con las bases de datos locales y remotas.
- El método delete (*DLT) suprime los datos de la base de datos.

NOPRIMARY

Acción que Loader debe realizar cuando falta la clave primaria para un registro del archivo de entrada

- La opción de error (*ERROR) indica que se debe informar de que falta la clave primaria como un error y finalizar.
- La opción de ignorar (*SKIP) ignora los registros del archivo de entrada que no tengan una clave primaria.
- La opción de insertar (*INSERT) intenta procesar (insertar o suprimir) los datos.

Este parámetro es opcional. La acción por omisión es "error".

COMMITNUM

Número de registros procesados antes de que se comprometa la base de datos cuando se utiliza el método de importación de SQL. Este parámetro es opcional. El número por omisión es 1.

MAXERROR

Número de errores después de los cuales Loader finalizará el método de operación de importación de SQL. Este parámetro es opcional.

CUSTOMIZER

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de ID Resolver. El archivo por omisión es `ISeries_LODWCSDTA_Customizer.properties`. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(/wc/prop/ml.properties)
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema `CLASSPATH`, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(ml)
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Métodos para el mandato Load:

Antes de cargar datos, debe determinar cuál de los tres métodos de proceso producirá los mejores resultados.


Método de carga:

Tome en consideración el método de carga en cualquiera de las situaciones siguientes:



- Los datos fuente están limpios y la base de datos no contiene datos

Nota: Datos limpios son datos que no violan ninguna de las restricciones de las tablas en las que se están cargando.





- Los datos fuente están limpios y la base de datos no contiene los datos que se están cargando
- Los datos fuente están limpios, una o varias de las tablas de destino no contienen claves primarias y la base de datos no contiene los datos que se están cargando
- La base de datos es una base de datos DB2 local
- La base de datos es una base de datos Oracle local o remota
- Mientras está teniendo lugar la carga no están accediendo a la base de datos otros usuario u otras aplicaciones

 400 Con el método de carga, se cargan datos en la base de datos. Si los datos ya existen, el mandato falla como resultado de un error de clave duplicada y se visualiza un mensaje de error de duplicado.


Existen las restricciones siguientes al utilizar el método de carga:

- El método de carga no puede insertar o actualizar datos en campos de datos de bits.
-  Con el método de carga, sólo se insertan registros nuevos en la base de datos; los registros existentes no se actualizan.
-  El método de carga sólo se puede utilizar para bases de datos DB2 locales, no remotas.

Método de importación:

    Con el método de importación para DB2, también se cargan datos en la base de datos. Si los datos ya existen, no se suprimen sino que se actualizan con valores nuevos. Tenga en cuenta este método en cualquiera de las situaciones siguientes:

- El sistema de gestión de bases de datos es DB2
- No sabe si los datos están limpios o no
- Tiene que actualizar grandes conjuntos de datos homogéneos a nivel de columna
- Todas las tablas en las que se están importando datos tienen claves primarias

 400 Con el método de importación, también se cargan datos en la base de datos. Si los datos ya existen, no se suprimen sino que se actualizan con valores nuevos. Tenga en cuenta este método en cualquiera de las situaciones siguientes:

- No sabe si los datos están limpios o no
- Los datos ya existen en la base de datos
- Todas las tablas en las que se están importando datos tienen claves primarias

Existen las restricciones siguientes al utilizar el método de importación:

- Para utilizar el método de importación, el sistema de gestión de bases de datos debe ser DB2.
- El método de importación no puede insertar o actualizar datos en campos de datos de bits.
- Con el método de importación, Loader sólo inserta o actualiza tablas que tienen definidas claves primarias; el método de importación no puede insertar o actualizar datos en tablas que no tienen una clave primaria. Si el registro de entrada sólo tiene valores para columnas que son primarias, se rechaza el registro.

Método de importación de SQL:

Con el método de importación de SQL, se utilizan sentencias de JDBC o SQL para actualizar o insertar datos en la base de datos. Se insertan datos si aún no existen y los datos existentes se actualizan. Tenga en cuenta este método en cualquiera de las situaciones siguientes:

- Está actualizando datos existentes y necesita actualizaciones a nivel de columna
- Algunos de los datos no están limpios
- La base de datos no es local

Nota: Si está utilizando la sincronización de espacios de búsqueda del Asesor de productos, deberá utilizar el método de importación de SQL para cargar datos.

Método de supresión:

El método de supresión se utiliza para suprimir datos que están en el documento XML de entrada de la base de datos. El elemento debe contener los valores para la clave primaria o el índice exclusivo para la tabla. Si los datos que se están suprimiendo tienen datos en otra tabla que son dependientes de ellos con la "supresión en cascada" habilitada, los datos dependientes también se suprimen.

Comparación de los métodos:

- **Comparación de los métodos de carga e importación de SQL**
El método de importación de SQL comprueba la coherencia de los datos, incluidas las referencias externas, y le permite actualizar los datos existentes. El método de carga no realiza dichas acciones.
- **Comparación de los métodos de importación y de exportación de SQL**
Los métodos de importación y de exportación de SQL realizan funciones similares. El método de importación es normalmente más rápido, pero requiere espacio de disco para los archivos temporales.
El método de importación sólo puede insertar o actualizar tablas que tienen definidas claves primarias, mientras que el método de exportación de SQL no necesita que las tablas contengan claves primarias.
- **Comparación de métodos basándose en el producto de base de datos utilizado**
Los métodos de importación y carga utilizan programas de utilidad nativos que están optimizados para DB2, mientras que el método de importación de SQL utiliza llamadas JDBC (que son genéricas para muchos productos de base de datos).

Consideraciones acerca del rendimiento:

Cuando utilice Loader para cargar documentos grandes en una base de datos, tenga en cuenta los elementos siguientes:

- **Tamaño de almacenamiento dinámico de Java Virtual Machine (JVM)**
Por omisión, la cantidad máxima de memoria asignada al almacenamiento dinámico de JVM es de 64 MB. Si dicha cantidad no se incrementa, JVM puede quedarse finalmente sin memoria durante el proceso de carga. La cantidad máxima de memoria asignada al almacenamiento dinámico de Java puede modificarse utilizando la opción -mx de JVM en el mandato Java.
- **Anotaciones cronológicas de rastreo**
El anotador de rastreo puede agotar el almacenamiento dinámico de JVM al cargar un documento XML grande. La información de rastreo se utiliza principalmente para depurar una ejecución si ésta falla. Si no es necesario rastrear el proceso de carga, se deberá desactivar el rastreo. Hay un aumento significativo del rendimiento cuando el rastreo está desactivado. El rastreo se

desactiva modificando el documento XML de configuración de anotación cronológica. Para obtener información sobre cómo modificar el documento XML de configuración de anotaciones cronológicas, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

- **Número de compromisos**

El número de compromisos por omisión para Loader cuando éste está funcionando en modalidad de importación de SQL es 1. Por consiguiente, las transacciones se comprometen por omisión para cada actualización o inserción en la base de datos. Para mejorar el rendimiento de Loader para documentos grandes, se deberá incrementar el número de compromisos. Se sugiere un valor de "100"; pero puede ser mayor en función de la cantidad de memoria física del servidor, del tamaño de la anotación cronológica de transacciones de DBMS, etc. El número de compromisos para Loader se modifica utilizando la opción `-commitcount número` para el mandato Load (donde *número* es el número de sentencias ejecutadas antes de que se comprometa la transacción).

- **Configuración de la anotación cronológica**

Se puede producir un proceso inusualmente lento al cargar datos como resultado de una de las situaciones siguientes:

- El usuario que invoca Loader no tiene permiso para grabar en el directorio o para actualizar el archivo especificado en el documento de configuración de anotación cronológica.
- El directorio especificado como ubicación del archivo en el documento de configuración de anotación cronológica no existe.
- La unidad especificada como ubicación del archivo en el documento de configuración de anotación cronológica no tiene suficiente espacio.

Cuando corrija cualquiera de estos problemas, puede que necesite cambiar la ubicación especificada del archivo modificando el documento de configuración de anotación cronológica (por omisión `WCALoggerConfig.xml`). Para obtener información sobre cómo modificar el documento XML de configuración de anotaciones cronológicas, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.



Si desea obtener información detallada sobre cómo configurar y personalizar los archivos utilizados para ejecutar este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.





Mandatos de Loader package para transformar y extraer datos

Mandato DTD Generate

Este mandato crea una DTD para utilizarla con Loader package. Esta DTD se utilizará durante todo el proceso de carga de datos. En función de cómo invoque el mandato, DTD Generator puede generar una DTD sola o una DTD junto con un esquema XML.

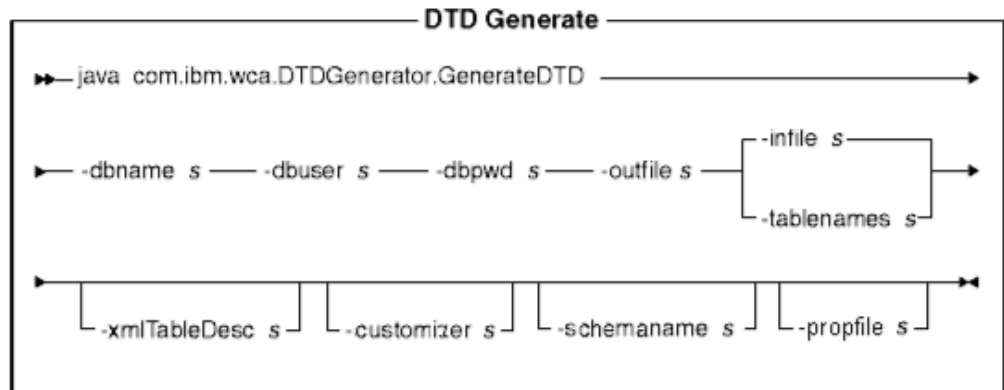
DTD Generator puede crear una DTD basándose en el esquema de base de datos de WebSphere Commerce. Si utiliza las DTD proporcionadas con los archivadores de tienda de ejemplo y no modifica el esquema de base de datos, no necesitará generar una DTD utilizando DTD Generator. Las DTD que se proporcionan están ubicadas en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\xml\sar`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar`

-  /usr/WebSphere/CommerceServer/xml/sar
-   /opt/WebSphere/CommerceServer/xml/sar
-  /QIBM/ProdData/WebCommerce/xml/sar

Se recomienda utilizar las DTD proporcionadas. Sin embargo, si personaliza un esquema de base de datos, deberá editar la DTD proporcionada para que coincida con los cambios o crear una DTD nueva.



Notas:

1. El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado "Utilización de los mandatos y scripts de Loader package" en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.
2. Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

-dbname

Nombre de la base de datos de destino

-dbuser

Nombre del usuario que se conecta a la base de datos

-dbpwd

Contraseña del usuario que se conecta a la base de datos

-outfile

Nombre del archivo DTD de salida (preferiblemente con una extensión .dtd)

-infile

Nombre de un archivo de entrada que contiene un nombre de tabla de base de datos en cada línea

-tablenames

Nombres de tablas separados por comas y escritos entre comillas ("")

-xmltabledesc

Vía de acceso del archivo de esquema XML que se debe crear. Este parámetro es opcional.

-customizer

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de DTD Generator. DB2ConnectionCustomizer.properties es el archivo por omisión. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:

```
-customizer d:\WebSphere\CommerceServer\prop\dtdgen.properties
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
-customizer dtdgen
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

-schemaname

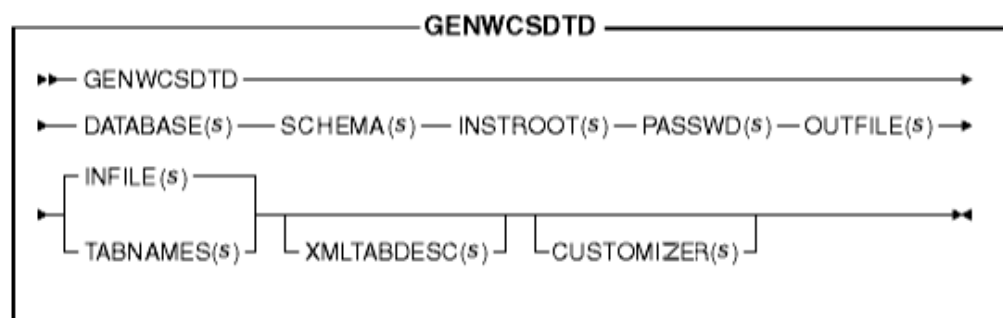
Nombre del esquema de base de datos de destino. Este parámetro es opcional.

Si no se especifica este parámetro al ejecutar el mandato, éste busca una pareja nombre=valor en el archivo de propiedades de personalización que especifica el valor de SchemaName. Si esta pareja se encuentra en el archivo de propiedades, el mandato utiliza el valor especificado. Si no se especifica ni en el archivo de propiedades ni en la línea de mandatos, el mandato toma, por omisión, el nombre del usuario de la base de datos.

-profile

Archivo de propiedades donde se puede almacenar texto de ayuda, valores por omisión e información de descripción de campos para una descripción de formulario de editor Web. Este parámetro es opcional.

▶ 400



Nota: Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

DATABASE

Nombre de la base de datos de destino, tal como aparece en el directorio de la base de datos relacional.

SCHEMA

Nombre del esquema de base de datos de destino; es igual que el nombre de la instancia.

INSTROOT

Nombre completo de la vía de acceso raíz de la instancia de WebSphere Commerce, por ejemplo, /QIBM/UserData/WebCommerce/instances/*/nombre_instancia*

PASSWD

Contraseña para la instancia de WebSphere Commerce

OUTFILE

Nombre del archivo DTD de salida (preferiblemente con una extensión .dtd)

INFILE

Nombre de un archivo de entrada que contiene un nombre de tabla de base de datos en cada línea

TABNAMES

Nombres de tablas separados por comas y escritos entre comillas ("")

XMLTABDESC

Vía de acceso del archivo de esquema XML que se debe crear. Este parámetro es opcional.

CUSTOMIZER

Nombre del archivo de propiedades de personalización que se va a utilizar. Este parámetro es opcional. El archivo de propiedades de personalización establece la forma de funcionamiento de DTD Generator. El archivo por omisión es `ISeries_GENWCSDTD_Customizer.properties`. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(/wc/prop/dtdgen.properties)
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema `CLASSPATH`, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(dtdgen)
```

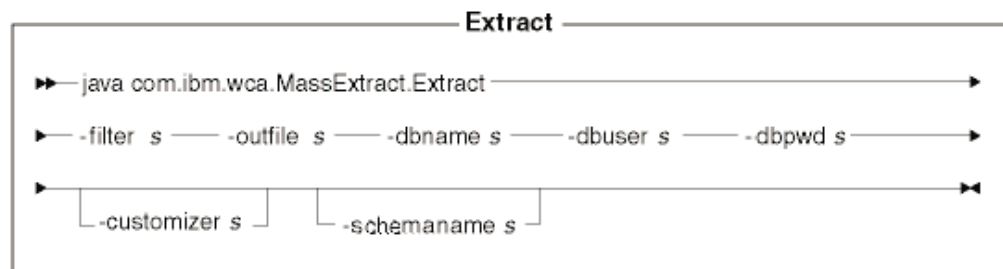
Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Si desea obtener información detallada sobre cómo configurar y personalizar los archivos utilizados para ejecutar este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Mandato Extract

Este mandato extrae un subconjunto seleccionado de datos de una base de datos en forma de archivo XML.

Para extraer datos de una base de datos utilizando Extractor, debe especificar los datos que desea extraer utilizando un archivo de filtro de extracción. El filtro de extracción que utilice dependerá del tipo de datos que desee extraer.



Notas:

1. El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.
2. Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

-filter Nombre del archivo de filtro de extracción

-outfile

Nombre del archivo XML de salida donde se almacenarán los datos extraídos

-dbname

Nombre de la base de datos de la que se están extrayendo datos

-dbuser

Nombre de usuario de la base de datos de la que se están extrayendo datos

-dbpwd

Contraseña asociada con el nombre de usuario para la base de datos de la que se están extrayendo datos

-customizer

Nombre del archivo de propiedades de personalización que se va a utilizar. El archivo de propiedades de personalización establece la forma de funcionamiento de Extractor. El archivo por omisión es `DB2ConnectionCustomizer.properties`. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:
`-customizer d:\WebSphere\CommerceServer\prop\extract.properties`

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema `CLASSPATH`, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

`-customizer extract`

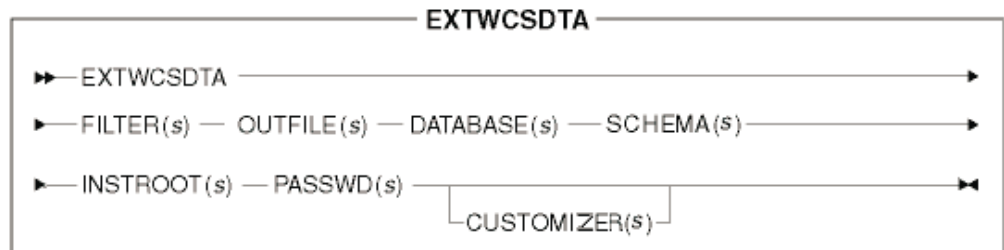
Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

-schemaname

Nombre del esquema de la base de datos de la que se están extrayendo datos. Este parámetro es opcional.

Si no se especifica este parámetro al ejecutar el mandato, éste busca una pareja nombre=valor en el archivo de propiedades de personalización que especifica el valor de SchemaName. Si esta pareja se encuentra en el archivo de propiedades, el mandato utiliza el valor especificado. Si no se especifica ni en el archivo de propiedades ni en la línea de mandatos, el mandato toma, por omisión, el nombre de esquema de la tabla en la base de datos.

▶ 400



Nota: Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

FILTER

Nombre del archivo de filtro de extracción

OUTFILE

Nombre del archivo XML de salida donde se almacenarán los datos extraídos

DATABASE

Nombre de la base de datos de la que se están extrayendo datos tal como se visualiza en el directorio de base de datos relacional.

SCHEMA

Nombre del esquema de base de datos del que se están extrayendo datos; es igual que el nombre de instancia

INSTROOT

Nombre completo de la vía de acceso raíz de la instancia de WebSphere Commerce, por ejemplo, /QIBM/UserData/WebCommerce/instances/*/nombre_instancia*

PASSWD

Contraseña para la instancia de WebSphere Commerce

CUSTOMIZER

Nombre del archivo de propiedades de personalización que se va a utilizar. El archivo de propiedades de personalización establece la forma de funcionamiento de Extractor. El archivo por omisión es ISeries_EXTWCSDTA_Customizer.properties. Se puede especificar el archivo de propiedades de personalización como se muestra en el ejemplo siguiente:

```
CUSTOMIZER(/wc/prop/extract.properties)
```

Si este archivo se encuentra en el directorio especificado en la variable de entorno del sistema CLASSPATH, se puede especificar el mismo archivo, como se muestra en el ejemplo siguiente:

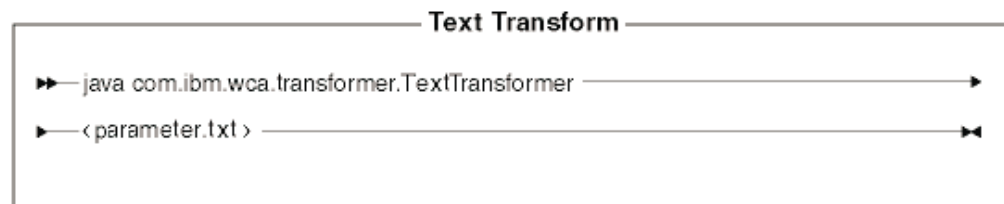
```
CUSTOMIZER(extract)
```

Para obtener más información sobre la creación y especificación de un nuevo archivo de propiedades de personalización, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Para obtener más información sobre este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Mandato Text Transform

Este mandato transforma los datos entre un formato variable delimitado por caracteres y un formato XML.



Nota: El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado "Utilización de los mandatos y scripts de Loader package" en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.

Valores de los parámetros:

Los valores siguientes se especifican y se separan mediante comas en un archivo de parámetros (*parameter.txt*):

input file

Nombre del archivo que se debe transformar

schema file

Nombre del archivo de esquema XML que se debe utilizar en la transformación

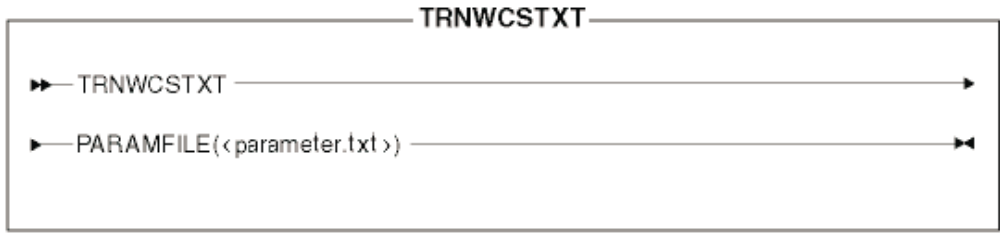
output file

Nombre del archivo de salida en el que se almacenarán los datos transformados

transformation method

Método a utilizar al añadir los datos en el archivo de salida. Especifique **Create** si se debe crear un archivo nuevo o especifique **Append** si los datos de salida deben añadirse a un archivo de datos existente.

Este archivo también se conoce como archivo de "manifiesto" o de "mandatos". Puede contener varias líneas de cuatro parámetros cada una.



Valores de los parámetros:

Los valores siguientes se especifican y se separan mediante comas en un archivo de parámetros (*parameter.txt*):

input file

Nombre del archivo que se debe transformar

schema file

Nombre del archivo de esquema XML que se debe utilizar en la transformación

output file

Nombre del archivo de salida en el que se almacenarán los datos transformados

transformation method

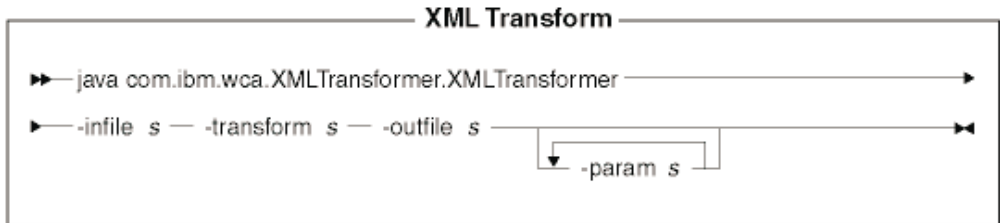
Método a utilizar al añadir los datos en el archivo de salida. Especifique **Create** si se debe crear un archivo nuevo o especifique **Append** si los datos de salida deben añadirse a un archivo de datos existente.

Nota: Este archivo también se conoce como archivo de "manifiesto" o de "mandatos".

Para obtener más información sobre este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Mandato XML Transform

Este mandato convierte un archivo XML en un formato XML alternativo.



Notas:

1. El diagrama anterior está principalmente destinado a utilizarse como referencia para los parámetros de mandato. El script o archivo de mandatos proporcionado para este mandato y listado bajo el apartado "Utilización de los mandatos y scripts de Loader package" en la página 269 actúa como wrapper para el mandato Java real y acepta los mismos parámetros; por consiguiente, se recomienda que utilice el script o archivo de mandatos en lugar de invocar directamente el mandato Java.

2. Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

-infile Nombre del archivo que se debe transformar

-transform

Nombre del archivo de normas de correlación de XSL de transformación

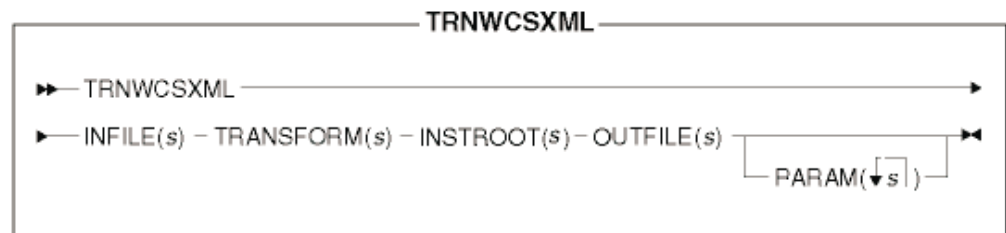
-outfile

Nombre del archivo XML de salida en el que se almacenarán los datos transformados

-param

Parámetro que se debe pasar al archivo de normas de correlación de XSL. *Este parámetro es opcional.* Este parámetro puede especificarse varias veces para pasar varias parejas nombre=valor.

▶ 400



Nota: Los nombres de archivo especificados como parámetros para este mandato pueden ir precedidos por vías de acceso relativas o absolutas.

Valores de los parámetros:

INFILE

Nombre del archivo que se debe transformar

TRANSFORM

Nombre del archivo de normas de correlación de XSL de transformación

INSTROOT

Nombre completo de la vía de acceso raíz de la instancia de WebSphere Commerce, por ejemplo, /QIBM/UserData/WebCommerce/instancias /nombre_instancia

OUTFILE

Nombre del archivo XML de salida en el que se almacenarán los datos transformados

PARAM

Parámetro que se debe pasar al archivo de normas de correlación de XSL. *Este parámetro es opcional.* La serie puede contener varios valores para pasar varias parejas nombre=valor.

Para obtener más información sobre este mandato, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Herramientas relacionadas con los mandatos de Loader package

Herramienta de Transformación de texto

La herramienta de Transformación de texto le ayuda a procesar una transformación de datos entre un formato variable delimitado por caracteres y un formato XML, utilizando el mandato Text Transform. Se proporcionan las vistas siguientes:

1. La Vista de edición de esquema de texto le permite crear y modificar el archivo de esquema XML que se va a utilizar en una transformación.
2. La Vista de edición de mandato de transformación le permite crear y modificar los mandatos reales utilizados para ejecutar el proceso de transformación.
3. La Vista de proceso de mandato de transformación le permite iniciar el proceso de transformación.

Para obtener más información sobre esta herramienta, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Editor XSL

XML Transformer utiliza XSL para definir las normas para transformar un archivo XML en otro archivo XML. La función de correlación en el editor XSL le proporciona una interfaz visual con la que puede establecer la asociación de un elemento de una DTD fuente a un elemento de una DTD de destino. Con dos DTD específicas, puede desarrollar normas XSL que determinen cómo un archivo XML que se ajusta a la primera DTD (fuente) se transforma en un archivo que se ajusta a la segunda DTD (destino).

Para obtener más información sobre esta herramienta, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Editor Web

El editor Web le permite crear, suprimir y cambiar los datos de catálogo a través de un navegador Web. Los formularios de entrada de datos para ver y actualizar la información son centrales para el editor Web. En el caso más simple, los formularios se corresponden con las tablas de la base de datos de WebSphere Commerce Server. El Desarrollador de tiendas puede optar por utilizar los formularios que se proporcionan o por personalizarlos.

Para obtener más información sobre esta herramienta, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Carga de los datos de la tienda

Esta sección proporciona ejemplos de cómo cargar datos de la tienda en la base de datos de WebSphere Commerce Server utilizando los programas de utilidad de línea de mandatos de Loader package.

Notas:

1. Los ejemplos de esta sección se realizan en un entorno Windows NT. Para obtener información sobre cómo ejecutar estos mandatos en otros entornos, consulte la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

- Aunque los programas de utilidad de línea de mandatos de Loader package soportan bases de datos DB2, DB2 para iSeries y Oracle, en los ejemplos siguientes sólo se incluyen los mandatos y las opciones para DB2. Si está utilizando una base de datos distinta de DB2, asegúrese de modificar los archivos de propiedades de personalización tal como se describe en la versión más reciente de la publicación *IBM WebSphere Commerce 5.4, Guía del usuario de Catalog Manager*.

Consulte el Capítulo 28, “Carga de grupos de elementos de base de datos de WebSphere Commerce” en la página 279 y el apartado “Carga de grupos de elementos de base de datos” en la página 287 si desea obtener información sobre el proceso de carga para grupos de elementos de base de datos de WebSphere Commerce.

Utilización de los mandatos y scripts de Loader package

Para ejecutar los mandatos de Loader package, utilice los scripts o mandatos proporcionados en el directorio siguiente de WebSphere Commerce:

- ▶ **NT** *unidad:*\WebSphere\CommerceServer\bin
- ▶ **2000** *unidad:*\Archivos de programa\WebSphere\CommerceServer\bin
- ▶ **AIX** /usr/WebSphere/CommerceServer/bin
- ▶ **Solaris** ▶ **Linux** /opt/WebSphere/CommerceServer/bin
- ▶ **400** Biblioteca nativa QWEECOMM

Los scripts y mandatos son los siguientes:

- ▶ **Windows**

dtdgen.cmd	Mandato DTD Generate
idresgen.cmd	Mandato ID Resolve
massload.cmd	Mandato Load
massextract.cmd	Mandato Extract
txtransform.cmd	Mandato Text Transform
xmltransform.cmd	Mandato XML Transform
- ▶ **AIX** ▶ **Solaris** ▶ **Linux**

dtdgen.sh	Script de shell DTD Generate
idresgen.sh	Script de shell ID Resolve
massload.sh	Script de shell Load
massextract.sh	Script de shell Extract
txtransform.sh	Script de shell Text Transform
xmltransform.sh	Script de shell XML Transform
- ▶ **400**

GENWCSDTD	Mandato DTD Generate
RESWCSID	Mandato ID Resolve
LODWCSDTA	Mandato Load
EXTWCSDTA	Mandato Extract
TRNWCSTXT	Mandato Text Transform
TRNWCSXML	Mandato XML Transform

Ejemplos de resolución de identificadores

Los ejemplos de resolución de identificadores descritos en esta sección utilizan los archivos de elementos de la **Business** tienda de ejemplo ToolTech.

Puesto que este ejemplo se basa en la carga de nuevos datos en la base de datos de WebSphere Commerce, utilizaremos el método de carga.

Si posteriormente necesita modificar determinados elementos en el documento XML, puede hacerlo utilizando el método de actualización. El método de actualización se ejecutará más rápidamente que el método de carga porque no se asignan identificadores nuevos con el método de actualización. Con el método de actualización, se realiza una consulta de base de datos para localizar el identificador y se informa de un error si no se encuentra el identificador. Consulte la sección que empieza en la página 250 para obtener más información sobre el funcionamiento de este proceso.

Si el archivo XML de entrada contiene elementos que ya existen en la base de datos así como elementos que no existen, utilice el método combinado. Con el método combinado, primero se realiza una búsqueda de la base de datos y se asigna un identificador al elemento si no se encuentra el registro. Cuando tenga dudas, utilice el método combinado. Aunque los métodos de carga y actualización proporcionan un rendimiento más rápido que el método combinado, el archivo XML resuelto producido por la utilización del método combinado tiene más probabilidades de cargarse sin errores.

Para obtener más información sobre el funcionamiento de ID Resolver, consulte el apartado “Métodos para el mandato ID Resolve” en la página 251.

Resolución de identificadores en archivos XML con alias internos

Para resolver identificadores utilizando los alias internos antes de cargar los datos en la base de datos de WebSphere Commerce Server, ejecute el mandato ID Resolve tal como se muestra en el ejemplo siguiente:

1. Asegúrese de que la vía de acceso incluya el directorio que contiene el script o mandato ID Resolve apropiado que se lista en el apartado “Utilización de los mandatos y scripts de Loader package” en la página 269.

Para este ejemplo, se utilizará el mandato ID Resolve `idresgen.cmd`.

2. Cree un directorio de trabajo.

Para este ejemplo, cree el directorio siguiente:
`c:\WebSphere\CommerceServer\runtime\test`


3. Asegúrese de que el archivo XML de entrada así como cualquier archivo DTD al que se haga referencia estén en una ubicación donde ID Resolver pueda encontrarlos.

Para este ejemplo, realice lo siguiente:

- a. Copie `c:\WebSphere\CommerceServer\samplestores\ToolTech\ToolTech_en_US_fr_FR.sar` en `c:\WebSphere\CommerceServer\runtime\test`.

- b. En un indicador de mandatos de Windows, entre el mandato siguiente:
`cd c:\WebSphere\CommerceServer\runtime\test`

- c. En el indicador de mandatos de Windows, entre el mandato siguiente:
`jar -xvf ToolTech_en_US_fr_FR.sar`

Esta operación extrae los  archivos XML de la tienda de ejemplo en `c:\WebSphere\CommerceServer\runtime\test\data`.

- d. En el indicador de mandatos de Windows, entre el mandato siguiente:
`cp c:\WebSphere\CommerceServer\xml\sar\store.dtd
c:\WebSphere\CommerceServer\runtime\test\data`

Esto copia el archivo store.dtd en
c:\WebSphere\CommerceServer\runtime\test\data.

- e. En el indicador de mandatos de Windows, entre el mandato siguiente:

```
cp c:\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd  
c:\WebSphere\CommerceServer\runtime\test\data
```

Esta operación copia el archivo DBLoadMacros.dtd en
c:\WebSphere\CommerceServer\runtime\test\data.

- f. En el indicador de mandatos de Windows, entre el mandato siguiente:

```
cp c:\WebSphere\CommerceServer\xml\sar\fulfillment.dtd  
c:\WebSphere\CommerceServer\runtime\test\data
```

Esta operación copia el archivo fulfillment.dtd en
c:\WebSphere\CommerceServer\runtime\test\data.

4. Asegúrese de que el esquema de WebSphere Commerce se carga en la base de datos junto con los datos de rutina de carga necesarios, creando una instancia de base de datos adecuada de WebSphere Commerce Server.

Nota: Para obtener más información sobre cómo crear una instancia, consulte la publicación WebSphere Commerce, Guía de instalación correspondiente a su sistema operativo.

La instancia de base de datos de WebSphere Commerce Server que utiliza este ejemplo se denomina *mall*. Las claves primarias y externas se obtendrán de las tablas KEYS y SUBKEYS de esta base de datos; por consiguiente, ID Resolver no podrá resolver los identificadores si la base de datos no se carga correctamente.

5. En el archivo store.xml, encontrará el elemento siguiente:

```
<storeent  
  STOREENT_ID="@storeent_id_1"  
  MEMBER_ID("&MEMBER_ID;"  
  TYPE="S"  
  IDENTIFIER="ToolTech"  
  SETCURR="USD"  
>
```

Este elemento de store.xml se correlaciona con la tabla storeent de la base de datos y sus atributos STOREENT_ID MEMBER_ID, TYPE, IDENTIFIER y SETCURR se correlacionan con columnas de esa tabla. La especificación @storeent_id_1 es un alias interno para el valor del atributo STOREENT_ID y &MEMBER_ID; es un parámetro de entidad. El valor de la entidad &MEMBER_ID; tiene que sustituirse para poderse cargar utilizando Loader. El valor de &MEMBER_ID; se define en el archivo de macros DBLoadMacros.dtd y el valor por el que se sustituye proviene de ese archivo. Cuando ID Resolver encuentra @storeent_id_1, examina la antememoria de tablas primarias para ver si existe storeent. Storeent existe, puesto que es una tabla primaria. ID Resolver busca el contador para dicha tabla, lo incrementa y sustituye el alias interno por el resultado. Todas las demás entradas de este tipo del archivo store.xml se procesan del mismo modo.

6. En el indicador de mandatos de Windows, entre el mandato siguiente:

```
idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml  
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml -method load
```

El primer fragmento XML de salida de store1.xml tiene este aspecto:

```

<storeent
  STOREENT_ID="10001"
  MEMBER_ID="-2001"
  TYPE="S"
  IDENTIFIER="ToolTech"
  SETCCURR="USD"
/>

```

El segundo fragmento XML de store1.xml tiene este aspecto:

```

<store
  STORE_ID="10001"
  DIRECTORY="ToolTech"
  FFMCENTER_ID=""
  LANGUAGE_ID="-1"
  STOREGRP_ID="-1"
  ALLOCATIONGOODFOR="43200"
  BOPMPADFACTOR="0"
  DEFAULTBOOFFSET="2592000"
  FFMCELECTIONFLAGS="0"
  MAXBOOFFSET="7776000"
  REJECTEDORDEXPIRY="259200"
  RTNFFMCTR_ID=""
  PRICEREFFLAGS="0"
  STORETYPE="B2B"
/>

```

Los atributos FFMCENTER_ID y RTNFFMCTR_ID no se han resuelto. En lugar de resolverse el alias interno @ffmcenter_id_1, éste se ha sustituido por comillas vacías (""). Esto es un error. No podrá cargar los datos correctamente utilizando Loader debido a una violación de restricción de clave externa. ID Resolver no ha podido resolver este alias interno porque la tabla FFMCENTER no se ha procesado antes de utilizar un alias que hace referencia a ella. Para solucionar este problema, seleccione una de las siguientes tres opciones:

- Opción 1:

- a. Entre el mandato siguiente para ejecutar ID Resolver en el archivo fulfillment.xml (donde está definida la tabla FFMCENTER):

```

idresgen -dbname mall -dbuser db2admin -dbpwd db2admin
-infile fulfillment.xml -outfile
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml
-method load

```

El elemento resuelto del archivo de salida fulfillment1.xml tiene este aspecto:

```

<fulfillment-asset>
<ffmcenter
  FFMCENTER_ID="10001"
  MEMBER_ID="-2001"
  NAME="ToolTech Home"
  DEFAULTSHIPOFFSET="0"
  MARKFORDELETE="0"
/>
</fulfillment-asset>

```

- b. Obtenga la clave FFMCENTER_ID del archivo de salida resultante (fulfillment1.xml) y sustituya dicha clave para todas las ocurrencias de @ffmcenter_id_1 en la copia de trabajo de store.xml en c:\WebSphere\CommerceServer\runtime\test\data.

- c. Entre el mandato siguiente:

```

idresgen -dbname mall -dbuser db2admin -dbpwd db2admin -infile store.xml
-outfile c:\WebSphere\CommerceServer\runtime\test\data\store1.xml
-method load

```

Los elementos totalmente resueltos del archivo de salida store1.xml tienen este aspecto:

```
<store-asset>
<storeent
  STOREENT_ID="10151"
  MEMBER_ID="-2001"
  TYPE="S"
  IDENTIFIER="ToolTech"
  SETCCURR="USD"
/>
<store
  STORE_ID="10151"
  DIRECTORY="ToolTech"
  FFMCENTER_ID="10001"
  LANGUAGE_ID="-1"
  STOREGRP_ID="-1"
  ALLOCATIONGOODFOR="43200"
  BOPMPADFACTOR="0"
  DEFAULTBOFFSET="2592000"
  FFMSELECTIONFLAGS="0"
  MAXBOFFSET="7776000"
  REJECTEDORDEXPIRY="259200"
  RTNFFMCTR_ID="10001"
  PRICEREFFLAGS="0"
  STORETYPE="B2B"
/>
<vendor
  VENDOR_ID="10001"
  STOREENT_ID="10151"
  VENDORNAME="Tooltech Vendor"
  MARKFORDELETE="0"
/>
<dispentrel
  AUCTIONSTATE="0"
  CATENTRY_ID="0"
  CATENTTYPE_ID="ProductBean"
  DEVICEFMT_ID="-1"
  DISPENTREL_ID="10001"
  MBRGRP_ID="0"
  PAGENAME="CatalogProductDisplay.jsp"
  STOREENT_ID="10151"
  RANK="0"
/>
<dispentrel
  AUCTIONSTATE="0"
  CATENTRY_ID="0"
  CATENTTYPE_ID="ItemBean"
  DEVICEFMT_ID="-1"
  DISPENTREL_ID="10002"
  MBRGRP_ID="0"
  PAGENAME="CatalogItemDisplay.jsp"
  STOREENT_ID="10151"
  RANK="0"
/>
<dispcgprel
  CATGROUP_ID="0"
  DEVICEFMT_ID="-1"
  DISPCGPREL_ID="10001"
  MBRGRP_ID="0"
  PAGENAME="CatalogCategories.jsp"
  STOREENT_ID="10151"
  RANK="0"
/>
<invadjcode
  ADJUSTCODE="PCNT"
  INVADJCODE_ID="10001"
```

```

        MARKFORDELETE="0"
        STOREENT_ID="10151"
    />
<invadjcode
    ADJUSTCODE="SPLG"
    INVADJCODE_ID="10002"
    MARKFORDELETE="0"
    STOREENT_ID="10151"
    />
<invadjcode
    ADJUSTCODE="DISC"
    INVADJCODE_ID="10003"
    MARKFORDELETE="0"
    STOREENT_ID="10151"
    />
<rtreason
    REASONTYPE="C"
    RTNREASON_ID="10001"
    STOREENT_ID="10151"
    MARKFORDELETE="0"
    CODE="WPR"
    />
<rtreason
    REASONTYPE="B"
    RTNREASON_ID="10002"
    STOREENT_ID="10151"
    MARKFORDELETE="0"
    CODE="DEF"
    />
<rtreason
    REASONTYPE="M"
    RTNREASON_ID="10003"
    STOREENT_ID="10151"
    MARKFORDELETE="0"
    CODE="ERR"
    />
<rtreason
    REASONTYPE="M"
    RTNREASON_ID="10004"
    STOREENT_ID="10151"
    MARKFORDELETE="0"
    CODE="WPS"
    />
</store-asset>

```

- Opción 2:
 - a. Fusione los archivos fulfillment.xml y store.xml añadiendo el contenido que sea exclusivo de fulfillment.xml (incluida la referencia a fulfillment.dtd) en store.xml, asegurándose de que el elemento ffmcenter que se muestra más abajo precede al elemento store.

```





<ffmcenter
    FFMCENTER_ID="@ffmcenter_id_1"
    MEMBER_ID="&MEMBER_ID;"
    NAME="ToolTech Home"
    DEFAULTBOFFSET="0"
    MARKFORDELETE="0"
    />






```

- b. Ejecute ID Resolver en el archivo fusionado.
- Opción 3: Cargue el grupo de datos de elementos de tienda utilizando el proceso que se describe en el apartado “Carga de grupos de elementos de base de datos” en la página 287.


Especificación de un archivo de propiedades con ID Resolver

Puede modificar el modo en que ID Resolver resuelve los identificadores utilizando el parámetro `-propfile`. El archivo de propiedades por omisión es `IdResolveKeys.properties`; pero puede modificarlo o especificar un archivo propio al invocar el mandato ID Resolve.

-     `IdResolveKeys.properties` está ubicado en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\properties`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\properties`
-  `/usr/WebSphere/CommerceServer/properties`
-   `/opt/WebSphere/CommerceServer/properties`

Si no pone este archivo en el directorio actual al ejecutar ID Resolver, puede colocarlo en un directorio definido en la variable de entorno `classpath`. También puede especificar la vía de acceso completa a este archivo.

-  Para cambiar `IdResolveKeys.properties`, cópielo del directorio `/QIBM/ProdData/WebCommerce/properties`, guárdelo en el directorio `/instroot/xml` y, a continuación, realice los cambios necesarios en el nuevo archivo.

Nota: El directorio anterior está en la variable `classpath` utilizada por el mandato `RESWCSID`.

La especificación del archivo de propiedades tiene prioridad respecto al uso de alias internos.

A continuación se muestra un fragmento XML de ejemplo del archivo `store.xml`:

```
<store
  STORE_ID="@storeent_id_1"
  DIRECTORY="ToolTech"
  FFMCENTER_ID="@ffmcenter_id_1"
  LANGUAGE_ID("&es_ES;"
  STOREGRP_ID="-1"
  ALLOCATIONGOODFOR="43200"
  BOPMPADFACTOR="0"
  DEFAULTBOOFFSET="2592000"
  FFMCELECTIONFLAGS="0"
  MAXBOOFFSET="7776000"
  REJECTEDORDEXPIRY="259200"
  RTNFFMCTR_ID="@ffmcenter_id_1"
  PRICEREFFLAGS="0"
  STORETYPE="B2B"
/>
```

Si ejecuta ID Resolver con `-propfile` especificado como `c:\WebSphere\CommerceServer\runtime\test\data\miArchivoProp` y el archivo especificado, `miArchivoProp.properties`, contiene las entradas siguientes:

```
NAMEDELIMETER=@
SELECTDELIMETER=:
FFMCENTER=@FFMCENTER_ID@MEMBER_ID:10051 -2001
```

ID Resolver consulta en la base de datos la tabla `FFMCENTER` con una cláusula `where` de `10051` y `-2001` cuando se procesa el elemento de tienda. Entonces se utiliza el índice que se devuelve para este valor para resolver el identificador para `FFMCENTER_ID`.

Para obtener más información sobre la utilización de este mandato, consulte el apartado "Mandato ID Resolve" en la página 245.

Ejemplo de carga de datos

Una vez que haya resuelto los identificadores del archivo XML, si son necesarios, estará listo para cargar los datos en la base de datos de WebSphere Commerce Server.

Nota: Si ha resuelto correctamente los identificadores en sus datos XML, el archivo XML fuente no debería contener ninguno de estos elementos:

- palabras precedidas por el símbolo arroba (@)
- palabras precedidas por el símbolo y (&)
- identificadores con comillas vacías ("")

La presencia de alguno de ellos indica que el archivo XML no está preparado para cargarse.

El ejemplo de carga de datos descrito en esta sección utiliza el archivo `fulfillment1.xml` que se ha resuelto en el apartado "Ejemplos de resolución de identificadores" en la página 269.

Para cargar datos en la base de datos de WebSphere Commerce Server, ejecute el mandato Load tal como se muestra en el ejemplo siguiente:

1. Asegúrese de que la vía de acceso incluye el directorio que contiene el script o el mandato Load apropiado tal como se lista en el apartado "Utilización de los mandatos y scripts de Loader package" en la página 269.

Para este ejemplo, utilice `massload.cmd`.

2. Cree un directorio de trabajo.

Para este ejemplo, utilice el directorio denominado

`c:\WebSphere\CommerceServer\runtime\test\data` que ha creado en el apartado "Ejemplos de resolución de identificadores" en la página 269.

3. Asegúrese de que el archivo XML de entrada está en una ubicación donde Loader puede encontrarlo.

Para este ejemplo, asegúrese de que el archivo de salida `fulfillment1.xml` que ha creado en el apartado "Ejemplos de resolución de identificadores" en la página 269 está en `c:\WebSphere\CommerceServer\runtime\test\data`.

4. Vaya al directorio de trabajo.

Para este ejemplo, entre el mandato siguiente en una línea de mandatos de Windows:

```
cd c:\WebSphere\CommerceServer\runtime\test\data
```

5. Haga una copia de seguridad de su base de datos de WebSphere Commerce Server de forma que pueda restaurar la base de datos si se produce un error irreparable.

6. Ejecute el mandato Load contra el archivo XML resuelto para cargar los datos en la base de datos de destino.

Para este ejemplo, entre el mandato siguiente en una línea de mandatos de Windows:

```
massload -dbname mall -dbuser db2admin -dbpwd db2admin -infile  
c:\WebSphere\CommerceServer\runtime\test\data\fulfillment1.xml  
-method sqlimport -commitcount 50
```

Aunque haya menos de 50 elementos a cargar, este ejemplo especifica un valor de 50 para `-commitcount`. Es por motivos de rendimiento. Por omisión, el número de

compromisos es 1. Si se utiliza este valor por omisión, se graba en la base de datos una operación de compromiso para cada registro. Si establece el número en 50 en el ejemplo anterior, se asegurará de que la E/S de la base de datos sólo se produzca una vez si la carga es satisfactoria y de que no se grabe nada en la base de datos si se producen errores. Sin embargo, si tiene que cargar una gran cantidad de datos, se recomienda que no establezca el número de compromisos en un valor tan grande como el del número de elementos por las razones siguientes:

- Un valor de número de compromisos alto produce un alto consumo de memoria.
- Cuando el valor del número de compromisos es más pequeño que el número de elementos, al menos se han grabado algunos datos en la base de datos. En función del valor de `-maxerror`, un valor más pequeño para `-commitcount` asegura que algunos datos se graben en la base de datos antes de que se exceda el número máximo de errores y que termine la herramienta. El valor por omisión para `-maxerror` es 1.

El valor por omisión de la opción `-noprimary` es error para que la herramienta informe de los errores y termine cuando falten claves primarias.

Dado que estos ejemplos no cargan los elementos de tienda en el orden utilizado por los Servicios de tienda y que se describe en el apartado “Secuencia de carga de elementos de base de datos” en la página 280, puede que el archivo `store1.xml` creado en el apartado “Ejemplos de resolución de identificadores” en la página 269 viole las restricciones de integridad de algunas tablas. Si ha intentado cargar `store1.xml` sin modificar utilizando el método de carga, una violación de restricción hará que la base de datos entre en estado de pendiente. Por consiguiente, para una mayor sencillez, este ejemplo de utilización del mandato `Load` se basa en la versión resuelta del archivo `fulfillment.xml`, cuya única clave externa es la del `MEMBER_ID` definido en la tienda de ejemplo. Este ejemplo carga el archivo `fulfillment1.xml` resuelto que se ha generado en el apartado “Ejemplos de resolución de identificadores” en la página 269 y utiliza el método `SQL`. Cuando no esté seguro de si el contenido del archivo XML está limpio, utilice el método de importación `SQL` tal como se muestra en este ejemplo, con los parámetros `-commitcount` y `-maxerror` establecidos en los valores adecuados, de forma que se informe de cualquier violación de restricción de base de datos sin alterar la base de datos ni poner en peligro la integridad de la misma.

Para obtener más información sobre la utilización de este mandato, consulte el apartado “Mandato `Load`” en la página 253.

Capítulo 28. Carga de grupos de elementos de base de datos de WebSphere Commerce

Si no desea crear todos los elementos de base de datos y empaquetarlos en un archivo archivador de tienda antes de la publicación, puede cargar los grupos de elementos de base de datos utilizando Loader package de WebSphere Commerce.

La primera parte de este capítulo explica los grupos de elementos de base de datos de WebSphere Commerce y cómo se determinan las agrupaciones. La segunda parte describe el proceso de carga de estos grupos de elementos de base de datos, en la base de datos de WebSphere Commerce. Antes de leer esta sección, debe revisar cuidadosamente la información del Capítulo 27, "Visión general de la carga de datos de tienda" en la página 241, que ayuda a entender lo que debe saber para cargar grupos de elementos de base de datos con Loader package.

Grupos de elementos de base de datos

Los elementos de base de datos se dividen en grupos para simplificar los procesos de creación y carga. Estos *grupos de elementos de base de datos* constan de un conjunto de tablas relacionadas de forma lógica. El orden en el que se organiza un grupo de elementos de base de datos es importante para la carga, porque es necesario que los datos existan antes de cargar la relación entre ellos.

Para cargar el conjunto entero de elementos de base de datos para una tienda, debe seguir las instrucciones del apartado "Secuencia de carga de elementos de base de datos" en la página 280. Para cargar un grupo individual de elementos de base de datos, debe asegurarse de que este grupo sea lógicamente completo. Por ejemplo, al publicar un archivador de tienda, puede elegir omitir los elementos de base de datos de catálogo, que puede publicar posteriormente. En ese caso, los elementos de base de datos dependientes del catálogo (inventario, listas de precio y algunos datos de envío e impuestos) también quedan sin publicarse. Para publicar los datos omitidos, asegúrese de que los elementos de base de datos de catálogo sean lógicamente completos: es decir, se deben proporcionar los elementos base, las entradas de catálogo, los atributos, etc. También debe publicar los elementos de base de datos dependientes que deben ser lógicamente completos por sí mismos. En otras palabras, cada código de artículo debe tener definidos el inventario, el precio, el envío y los impuestos apropiados. En este caso, los datos de catálogo relacionados que están lógicamente completos se denominan colectivamente *grupo de elementos de base de datos de catálogo*.


Los elementos de base de datos de WebSphere Commerce descritos en los capítulos anteriores de esta guía pueden disponerse en grupos. Un grupo es un conjunto de datos completo lógicamente, que se puede cargar de forma individual. Cada grupo de elementos de base de datos consta de tablas de base de datos de WebSphere Commerce y tiene dependencias externas, tal como se describe en el Apéndice E, "Grupos de elementos de base de datos" en la página 339. La lista de tablas se basa en las tiendas de ejemplo de WebSphere Commerce, aunque la lista es aplicable a cualquier tienda genérica. Recuerde que la lista de tablas para cada grupo de elementos de base de datos no es exhaustiva, sino que se proporciona como directriz general. Puede que necesite incluir o excluir algunas tablas en función de las necesidades específicas de su tienda.

Secuencia de carga de elementos de base de datos

Existe un orden determinado a seguir para cargar los grupos de elementos de base de datos de forma satisfactoria. Se considera que cada grupo es estructuralmente completo e independiente de los otros grupos de elementos de base de datos. Sin embargo, dentro de un grupo de elementos de base de datos hay relaciones de clave externa. Estas relaciones (con datos de otros grupos) se denominan *dependencias externas* de un grupo de elementos de base de datos.

Las dependencias externas de un grupo de elementos de base de datos deben satisfacerse antes de cargar dicho grupo en la base de datos de WebSphere Commerce. Los grupos definidos como dependencia externa de un grupo concreto de elementos de base de datos, deben cargarse en primer lugar. Puede encontrar la lista de dependencias externas y tablas relacionadas en el apartado “Dependencias de grupos de elementos de base de datos” en la página 339.

Nota: Una tienda de WebSphere Commerce requiere un propietario de la tienda. Puede utilizar la organización por omisión como el propietario por omisión.

 Para cargar este grupo, debe crear una nueva organización en lugar de utilizar la organización por omisión.

Cargue los grupos de elementos de base de datos en el siguiente orden:

1. Sólo grupos de elementos de base de datos dependientes de datos de rutina de carga.
 - a. En primer lugar, cargue los elementos de base de datos de la **organización**.
2. Grupo de elementos de base de datos dependiente del propietario del centro de despacho.
 - a. Elementos de la base de datos de **despacho de pedidos**. A excepción del grupo de elementos de base de datos de la organización, hay otros grupos de elementos de base de datos que tienen una dependencia externa directa o indirecta con los datos definidos en este grupo.
3. Grupos de elementos de base de datos dependientes de la organización de propietarios de la tienda.
 - a. Los elementos de la base de datos del **control de acceso** son dependientes de la organización propietaria de la tienda (ORGENTITY_ID). Ninguno de los demás grupos de elementos de base de datos tiene una dependencia en los datos definidos en este grupo, lo que significa que los elementos de base de datos de control de acceso pueden cargarse en cualquier momento. Sin embargo, el propietario del control de acceso debe ser el mismo que el propietario de la tienda.
 - b. Los elementos de la base de datos de la **tienda** son dependientes de la organización propietaria de la tienda (ORGENTITY_ID).

La tienda puede hacer referencia a un centro de despacho de pedidos. La organización propietaria de la tienda también puede ser la organización propietaria del centro de despacho de pedidos.


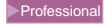
4. Grupos de elementos de base de datos dependientes de los elementos de base de datos de la tienda. Los grupos siguientes pueden cargarse en cualquier orden:
 - a. Elementos de la base de datos de **campañas**.
 - b. Elementos de la base de datos de **mandatos**.
 - c. Elementos de la base de datos de **monedas**.
 - d. Elementos de la base de datos de **políticas**.

- e. Elementos de la base de datos de **envíos**.
 - f. Elementos de la base de datos de **impuestos**.
5. Otros grupos de elementos de base de datos.
- a. Los elementos de base de datos del **catálogo** son dependientes de los grupos de elementos de base de datos de envío e impuestos.
 - b. Los elementos de base de datos de la **tienda por omisión** tienen dependencias externas con el grupo de elementos de base de datos de envío. Si no hay elementos de base de datos de envío, no es necesario rellenar con datos este grupo.
 - c. Los elementos de base de datos del **contrato** son dependientes de los elementos de la organización. Los elementos de la base de datos del contrato no se cargan directamente. Consulte el apartado “Publicación de elementos de contrato” en la página 295 para obtener más información. Debe cargar los elementos de contrato después de cargar los demás grupos de elementos de base de datos.

En el Apéndice E, “Grupos de elementos de base de datos” en la página 339 puede consultar el contenido de los grupos de elementos de base de datos, tal como están formados por las tiendas de ejemplo de WebSphere Commerce.

Carga de una tienda







Para ayudarle a cargar elementos de base de datos, en el sitio Web de WebSphere Commerce encontrará paquetes de ejemplo. Estos paquetes se basan en las tiendas de ejemplo de WebSphere Commerce y contienen los archivos que se indican en los pasos siguientes. Puede bajar estos paquetes de:

-  http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html
-  http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Para cargar datos XML para una tienda entera en la base de datos de WebSphere Commerce, realice lo siguiente:







1. Revise la información siguiente:
 - a. El Apéndice B, “Creación de los datos” en la página 327
 - b. El Apéndice E, “Grupos de elementos de base de datos” en la página 339, ya que debe conocer qué tablas de base de datos y archivos de elementos de base de datos de WebSphere Commerce están afectados.
 - c. El Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241, que proporciona información básica sobre Loader package.
2. Planifique el proceso de carga para un conjunto completo de elementos de base de datos de la tienda. Tanto si desea cargar un solo grupo de elementos de base de datos, según las instrucciones del apartado “Carga de grupos de elementos de base de datos” en la página 287, como si desea cargar una tienda entera, el proceso básico sigue siendo el mismo. En los pasos siguientes, utilizará o creará los archivos siguientes para el proceso de carga:
 - a. uno o más archivos de elementos de base de datos para cada grupo. Si carga la tienda completa, necesitará todos los archivos de elementos de base de datos creados. Por ejemplo, necesitará un archivo *elemento_base_datos.xml* (por ejemplo, *campaign.xml*, *catalog.xml* o *currency.xml*) y archivos *elemento_base_datos.xml* aparte, específicos del

entorno nacional al que la tienda dé soporte. Con las tiendas de ejemplo de WebSphere Commerce, se envían ejemplos de archivos de este tipo en el directorio siguiente:

-  *unidad:* \WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data
-  *unidad:* \Archivos de programa\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data
-  /usr/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo\data
-   /opt/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data/
-  /QIBM/ProdData/WebCommerce/samplstores/
nombre tienda ejemplo/data/

Tenga en cuenta que no todos los grupos de elementos de base de datos requieren información específica del entorno nacional.

- b. un nuevo archivo XML que agrupa todos los archivos de elementos de base de datos XML para su tienda, que contiene las referencias a entidades XML y que también contiene el elemento raíz para la tienda entera. Este archivo se denomina *archivo XML principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado *store-data-assets.xml*.
- c. un archivo DTD nuevo que define todos los tipos de datos necesarios para los archivos XML de un grupo de elementos de base de datos, que se denomina *archivo DTD principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado *store-data-assets.dtd*.
- d. un segundo archivo DTD que define las dependencias externas. Quizá tenga que incluir este archivo en el *archivo DTD principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado *ForeignKeys.dtd*.
- e. un tercer archivo DTD que contiene la definición de todas las tablas de WebSphere Commerce. El archivo *wcs.dtd* ya existe en WebSphere Commerce y está ubicado en el directorio siguiente:

-  *unidad:* \WebSphere\CommerceServer\schema\xml\
nombre tienda ejemplo\data
-  *unidad:* \Archivos de programa\WebSphere\CommerceServer\
schema\xml
-  /usr/WebSphere/CommerceServer/schema/xml/
nombre tienda ejemplo\data
-   /opt/WebSphere/CommerceServer/schema/xml/
nombre tienda ejemplo/data/
-  /QIBM/ProdData/WebCommerce/schema/xml/
nombre tienda ejemplo/data/

Quizá tenga que incluir este archivo en el *archivo DTD principal del grupo de elementos de base de datos*. Si no ha personalizado el esquema de WebSphere Commerce, puede utilizar este archivo sin modificarlo.

3. Cree archivos XML de elementos de base de datos como se ha indicado en los capítulos anteriores de esta guía. Si ha completado las tareas de los capítulos de los elementos, estos archivos XML ya existen. Los archivos de elementos de base de datos no deben contener ninguna declaración de DTD ni directivas de página al principio del archivo porque esto puede producir conflictos al

concatenar los archivos. Para una mayor simplicidad, quizá decida no crear ningún elemento raíz. El único archivo que debe tener un elemento raíz es el archivo XML principal de elementos de base de datos.

Nota: Si tiene archivos de elementos de base de datos para más de un idioma, cada archivo debe empezar por `<?xml encoding = codificación específica del entorno nacional>`. Por ejemplo, los archivos de elementos de base de datos ingleses deben especificar `<?xml encoding = "UTF-8"?>`, pero los archivos franceses deben especificar `<?xml encoding = "ISO-8859-1"?>`.

4. Cree el archivo XML principal del grupo de elementos de base de datos para todo el conjunto de datos de la tienda. Este archivo contiene entidades de referencia para incluir diversos archivos XML de elementos de base de datos para la tienda. Mediante la utilización de entidades de referencia externa puede concatenar los archivos XML para simplificar el mandato ID Resolve y el proceso de carga. Además, los alias internos utilizados dentro de cada archivo XML pueden ser externos con respecto a otro archivo XML de elementos de base de datos dentro de un grupo o en otros grupos, al cargar más de un grupo a la vez. Un analizador XML sustituirá la referencia externa por el contenido del archivo al que hace referencia la entidad de referencia externa.

Utilizando el ejemplo siguiente como guía para cargar el conjunto entero de datos de tienda, puede crear el archivo de grupo de elementos de base de datos basándose en este extracto:

```
<?xml version="1.0"?>
<!DOCTYPE import SYSTEM "all-store-assets.dtd">
<import>
<!Fulfillment data group -->
&fulfillment.xml;

<!-- Store data group -->
&store.xml;
&en_US_store.xml;
&fr_FR_store.xml;

<!-- Tax data group -->
&tax.xml;
&en_US_tax.xml;
&fr_FR_tax.xml;
&taxfulfill.xml;

<!-- Shipping data group -->
&shipping.xml;
&en_US_shipping.xml;
&fr_FR_shipping.xml;
&shipfulfill.xml;

<!-- Catalog data group -->
&catalog.xml;
&en_US_catalog.xml;
&fr_FR_catalog.xml;
&storecatalog.xml;
&storefulfill.xml;
&offering.xml;
&store-catalog-tax.xml;
&store-catalog-shipping.xml;

<!-- Currency data group -->
&currency.xml;
&en_US_currency.xml;
&fr_FR_currency.xml;
```

```

<!-- Campaign data group -->
&campaign.xml;
&en_US_campaign.xml;
&fr_FR_campaign.xml;

<!-- Business policy data group -->
&businesspolicy.xml;
&en_US_businesspolicy.xml;
&fr_FR_businesspolicy.xml;

<!-- Access control data group -->
&accesscontrol.xml;
&en_US_accesscontrol.xml;
&fr_FR_accesscontrol.xml;

<!-- Other data groups -->
&command.xml;
&store-default.xml;
</import>

```

donde

- `import` es el elemento raíz del documento XML. El elemento raíz ya se ha definido en el archivo `wcs.dtd`, proporcionado con WebSphere Commerce, e incluye las definiciones para todas las tablas de la base de datos de WebSphere Commerce. Sin embargo, si ha personalizado el esquema de WebSphere Commerce, puede que necesite utilizar un elemento raíz diferente. Puede generar un nuevo archivo DTD que refleje el esquema personalizado o puede actualizar el archivo `wcs.dtd` existente.
 - `all-store-assets.dtd` hace referencia al nombre del archivo DTD principal de grupo de elementos de base de datos que creará en el paso siguiente.
 - el texto comentado separa los distintos grupos de elementos de base de datos para la tienda.
 - `&elemento base datos.xml`; es una referencia de entidad XML al archivo XML de elementos de base de datos. La vía de acceso y ubicación se definen en el archivo DTD de grupo de elementos de base de datos. Este nombre cambiará para coincidir con los archivos de elementos de base de datos ya creados para cada grupo.
 - `&entorno nacional_elemento base datos.xml`; es necesario para cada idioma que soporte la tienda. Si la tienda es monolingüe, entonces solamente hará referencia a un archivo. Si la tienda soporta más de un idioma, entonces necesitará una referencia para cada idioma. El extracto anterior supone que la tienda soporta los idiomas inglés y francés.
5. Cree un archivo DTD principal de grupo de elementos de base de datos que defina las entidades de referencia anteriores y los demás archivos DTD que son necesarios para los elementos de base de datos.

Utilizando el ejemplo siguiente como guía para todo el conjunto de elementos de bases de datos de la tienda, puede crear el archivo DTD principal de grupo de elementos de base de datos:

```

<!ENTITY % wcs.dtd SYSTEM "vía completa para archivo wcs.dtd de WebSphere Commerce">
%wcs.dtd;

<!ENTITY %NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
<!ENTITY fulfillment.xml SYSTEM "data/fulfillment.xml">
<!ENTITY en_US_fulfillment.xml SYSTEM "data/en_US/fulfillment.xml">
<!ENTITY fr_FR_fulfillment.xml SYSTEM "data/fr_FR/fulfillment.xml">

<!ENTITY store.xml SYSTEM "data/store.xml">

```

```

<!ENTITY en_US_store.xml SYSTEM "data/en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "data/fr_FR/store.xml">

<!ENTITY tax.xml SYSTEM "data/tax.xml">
<!ENTITY en_US_tax.xml SYSTEM "data/en_US/tax.xml">
<!ENTITY fr_FR_tax.xml SYSTEM "data/fr_FR/tax.xml">
<!ENTITY taxfulfill.xml SYSTEM "data/taxfulfill.xml">

<!ENTITY shipping.xml SYSTEM "data/shipping.xml">
<!ENTITY en_US_shipping.xml SYSTEM "data/en_US/shipping.xml">
<!ENTITY fr_FR_shipping.xml SYSTEM "data/fr_FR/shipping.xml">
<!ENTITY shipfulfill.xml SYSTEM "data/shipfulfill.xml">

<!ENTITY catalog.xml SYSTEM "data/catalog.xml">
<!ENTITY en_US_catalog.xml SYSTEM "data/en_US/catalog.xml">
<!ENTITY fr_FR_catalog.xml SYSTEM "data/fr_FR/catalog.xml">
<!ENTITY store-catalog.xml SYSTEM "data/store-catalog.xml">
<!ENTITY storefulfill.xml SYSTEM "data/storefulfill.xml">
<!ENTITY offering.xml SYSTEM "data/offering.xml">
<!ENTITY store-catalog-tax.xml SYSTEM "data/store-catalog-tax.xml">
<!ENTITY store-catalog-shipping.xml SYSTEM "data/store-catalog-shipping.xml">

<!ENTITY currency.xml SYSTEM "data/currency.xml">
<!ENTITY en_US_currency.xml SYSTEM "data/en_US/currency.xml">
<!ENTITY fr_FR_currency.xml SYSTEM "data/fr_FR/currency.xml">

<!ENTITY campaign.xml SYSTEM "data/campaign.xml">
<!ENTITY en_US_campaign.xml SYSTEM "data/en_US/campaign.xml">
<!ENTITY fr_FR_campaign.xml SYSTEM "data/fr_FR/campaign.xml">

<!ENTITY businesspolicy.xml SYSTEM "data/businesspolicy.xml">
<!ENTITY en_US_businesspolicy.xml SYSTEM "data/en_US/businesspolicy.xml">
<!ENTITY fr_FR_businesspolicy.xml SYSTEM "data/fr_FR/businesspolicy.xml">

<!ENTITY accesscontrol.xml SYSTEM "data/accesscontrol.xml">
<!ENTITY en_US_accesscontrol.xml SYSTEM "data/en_US/accesscontrol.xml">
<!ENTITY fr_FR_accesscontrol.xml SYSTEM "data/fr_FR/accesscontrol.xml">

<!ENTITY command.xml SYSTEM "data/command.xml">
<!ENTITY store-defaults.xml SYSTEM "data/store-defaults.xml">

```

donde

- `wcs.dtd` hace referencia al archivo DTD que contiene datos definidos fuera de su grupo de elementos de base de datos. Este archivo, que se proporciona con WebSphere Commerce, también define el elemento raíz utilizado en el archivo XML de grupo de elementos de base de datos.
- `NonStoreForeignKeys.dtd` hace referencia al archivo DTD que define elementos distintos del elemento raíz. Este archivo contiene todas las declaraciones y definiciones de referencias de entidad XML para las dependencias externas fuera del grupo de elementos de base de datos. Por ello, los archivos XML contienen referencias a valores de clave externa que no se crean como parte del grupo de elementos de base de datos y que deben cargarse en la base de datos antes que este grupo.

Nota: Asegúrese de que la vía de acceso esté identificada correctamente. En este ejemplo, el archivo está en el mismo directorio que el archivo DTD principal de grupo de elementos de base de datos.

- `store.xml`, `en_US_store.xml` y `fr_FR_store.xml` son las entidades de referencia externa utilizadas en el archivo XML principal de grupo de elementos de base de datos, suponiendo que la tienda soporte los idiomas inglés y francés. Para utilizar la referencia, siga el convenio de referencia de entidad: *&nombre_alias;*.

- *elemento base datos.xml* hace referencia al nombre de los archivos XML de los que se cargan los elementos de base de datos. Este nombre cambiará para coincidir con los archivos de elementos de base de datos ya creados para cada grupo. Tenga en cuenta que los archivos XML específicos del entorno nacional se encuentran bajo el siguiente directorio:
 - **NT** *unidad:\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data\entorno nacional*
 - **2000** *unidad:\Archivos de
programa\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data\entorno nacional*
 - **AIX** */usr/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data/entorno nacional/*
 - **Solaris** **Linux** */opt/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data/entorno nacional/*
 - **400** */QIBM/ProdData/WebCommerce/samplstores/
nombre tienda ejemplo/data/entorno nacional/*
 - los archivos *vía de acceso_elemento base datos.xml* son necesarios para cada idioma que soporte la tienda, y se encuentran bajo los directorios anteriores. Si la tienda es monolingüe, entonces solamente hará referencia a un archivo. Si la tienda soporta más de un idioma, entonces necesitará un archivo específico del entorno nacional para cada idioma. El extracto anterior supone que la tienda soporta los idiomas inglés y francés.
6. Cada grupo de elementos de base de datos necesita información definida fuera de su dominio o su conjunto de datos, puesto que cada grupo puede tener dependencias externas. Puede proporcionar estos datos en un archivo DTD. Por ejemplo, el grupo de elementos de base de datos de tienda tiene las dependencias externas siguientes:

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

Al cargar un grupo de elementos de base de datos o el conjunto entero de elementos de tienda, se deberán definir las dependencias externas de la base de datos de WebSphere Commerce. Para utilizar estos datos, siga la referencia de entidad XML correspondiente. Por ejemplo, para utilizar los datos definidos por la entidad *ffmcenter_id*, deberá escribir *&ffmcenter_id*; en el archivo XML. Utilizando como guía el ejemplo siguiente de elementos de base de datos de tienda, puede crear su archivo DTD basándose en este extracto, denominado *NonStoreForeignKeys.dtd*:



```
<!ENTITY es_ES "-1">
<!ENTITY fr_FR "-2">
<!ENTITY de_DE "-3">
<!ENTITY it_IT "-4">
<!ENTITY es_ES "-5">
<!ENTITY pt_BR "-6">
<!ENTITY zh_CN "-7">
<!ENTITY zh_TW "-8">
<!ENTITY ko_KR "-9">
<!ENTITY ja_JP "-10">
<!ENTITY MEMBER_ID "-2000">
<!ENTITY ffmcenter_id "10001">
```

donde








- *MEMBER_ID* es el número de referencia interno que identifica al propietario de la tienda.

- `ffmcenter` es el número de referencia del centro de despacho de pedidos de la tienda. Puesto que la tienda puede utilizar más de un centro de despacho de pedidos, se puede definir más de uno en el archivo `NonStoreForeignKeys.dtd`.
- *entorno nacional* es el número de referencia de WebSphere Commerce para cada entorno nacional (que se identifica mediante el país o región y el idioma). Los valores están ubicados en la tabla de base de datos `LANGUAGE`.

Nota: Si está dividiendo un archivador de tienda existente en grupos de elementos de base de datos, asegúrese de que todas las referencias, por ejemplo, al alias `@ffmcenter_id` se sustituyen por su referencia de entidad correspondiente: `&ffmcenter_id`;

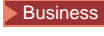

7. Una vez que se hayan creado todos los archivos de datos, ejecute el mandato `IDResolve` para el archivo XML principal de grupo de elementos de base de datos para resolver los datos, tal como se describe en el apartado “Mandato ID Resolve” en la página 245.
8. Ejecute el mandato `Load` en el archivo de datos resueltos tal como se describe en el apartado “Mandato Load” en la página 253. Para comprobar el proceso de carga, consulte los archivos de anotaciones:
 -  `idresgen.db2.log` y `massload.db2.log`
 -  `idresgen.oracle.log` y `massload.oracle.log`

Los archivos de anotaciones se encuentran bajo:

-  `unidad:\WebSphere\CommerceServer\logs\`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\logs\`
 -  `/usr/WebSphere/CommerceServer/logs/`
 -   `/opt/WebSphere/CommerceServer/logs/`
 -  `/QIBM/ProdData/WebCommerce/logs/`
9.  Ejecute el mandato `AccountImport` tal como se describe en el apartado “Publicación de elementos de cuenta de negocio” en la página 294.
 10. Si es aplicable, publique los contratos tal como se describe en el apartado “Publicación de elementos de contrato” en la página 295.
 11. Complete las tareas del apartado “Publicación de elementos de escaparate y archivos de configuración de tienda copiándolos en WebSphere Commerce Server” en la página 298







Carga de grupos de elementos de base de datos

Para ayudarle a cargar elementos de base de datos, en el sitio Web de WebSphere Commerce encontrará paquetes de ejemplo. Estos paquetes se basan en las tiendas de ejemplo de WebSphere Commerce y contienen los archivos que se indican en los pasos siguientes. Puede bajar estos paquetes de:

-  http://www.ibm.com/software/webservers/commerce/wc_be/downloads.html
-  http://www.ibm.com/software/webservers/commerce/wc_pe/downloads.html

Para cargar datos XML para un solo grupo de elementos de base de datos, en la base de datos de WebSphere Commerce, realice lo siguiente:







1. Revise la información siguiente:
 - a. El Apéndice B, “Creación de los datos” en la página 327
 - b. El Apéndice E, “Grupos de elementos de base de datos” en la página 339, ya que debe conocer qué tablas de base de datos y archivos de elementos de WebSphere Commerce están afectados.
 - c. El Capítulo 27, “Visión general de la carga de datos de tienda” en la página 241, que proporciona información básica sobre Loader package.
2. Planifique el proceso de carga y decida qué grupo de elementos de base de datos cargará. Tanto si desea cargar el conjunto entero de elementos de base de datos de tienda, según las instrucciones del apartado “Carga de una tienda” en la página 281, como si desea cargar un solo grupo de elementos de base de datos, el proceso básico sigue siendo el mismo. En los pasos siguientes, utilizará o creará los archivos siguientes para el proceso de carga:
 - a. uno o más archivos de elementos de base de datos, en función del grupo que elija. Por ejemplo, si carga los elementos de grupo de base de datos de tienda, necesitará un archivo `store.xml` y un archivo `store.xml` independiente para cada entorno nacional que soporte la tienda. Con las tiendas de ejemplo de WebSphere Commerce, se entregan ejemplos de archivos de este tipo en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data`
-  `unidad:\Archivos de
programa\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data`
-  `/usr/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data`
-   `/opt/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data`
-  `/QIBM/ProdData/WebCommerce/samplstores/
nombre tienda ejemplo/data`

Tenga en cuenta que no todos los grupos de elementos de base de datos requieren información específica del entorno nacional.

- b. un nuevo archivo XML que agrupa todos los archivos de elementos de base de datos XML, que contiene las referencias a entidades XML y que también contiene el elemento raíz para los elementos de base de datos. Este archivo se denomina *archivo XML principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado `store-all-assets.xml`.
- c. un archivo DTD nuevo que define todos los tipos de datos necesarios para los archivos XML de un grupo de elementos de base de datos, que se denomina *archivo DTD principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado `store-all-assets.dtd`.
- d. un segundo archivo DTD que define las dependencias externas. Quizá tenga que incluir este archivo en el *archivo DTD principal del grupo de elementos de base de datos*. Encontrará este archivo en el paquete de ejemplo, denominado `ForeignKeys.dtd`.

e. un tercer archivo DTD que contiene la definición de todas las tablas de WebSphere Commerce. El archivo wcs.dtd ya existe en WebSphere Commerce y está ubicado en el directorio siguiente:

-  `unidad:\WebSphere\CommerceServer\schema\xml\`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\schema\xml\`
-  `/usr/WebSphere/CommerceServer/schema/xml/`
-   `/opt/WebSphere/CommerceServer/schema/xml/`
-  `/QIBM/ProdData/WebCommerce/schema/xml/`

Quizá tenga que incluir este archivo en el *archivo DTD principal del grupo de elementos de base de datos*. Si no ha personalizado el esquema de WebSphere Commerce, puede utilizar este archivo sin modificarlo.

3. Cree archivos XML de elementos de base de datos para el grupo que cargará, tal como se ha indicado en los capítulos anteriores de esta guía. Si ha completado las tareas de los capítulos de los elementos, estos archivos XML ya existen. Los archivos de elementos de base de datos no deben contener ninguna declaración de DTD ni directivas de página al principio del archivo, porque esto puede producir conflictos al concatenar los archivos. Además, y para una mayor simplicidad, quizá decida no crear ningún elemento raíz. El único archivo que debe tener un elemento raíz es el archivo XML principal de elementos de base de datos.

Nota: Si tiene archivos de elementos de base de datos para más de un idioma, cada archivo debe empezar por `<?xml encoding = codificación específica del entorno nacional>`. Por ejemplo, los archivos de elementos de base de datos ingleses deben especificar `<?xml encoding = "UTF-8"?>`, pero los archivos franceses deben especificar `<?xml encoding = "ISO-8859-1"?>`.

4. Cree el archivo XML principal del grupo de elementos de base de datos para cada grupo que desee cargar. Este archivo contiene entidades de referencia para incluir diversos archivos XML en un grupo de elementos de base de datos, o más. Mediante la utilización de entidades de referencia externa puede concatenar los archivos XML para simplificar el mandato ID Resolve y el proceso de carga. Además, los alias internos utilizados dentro de cada archivo XML pueden ser externos con respecto a otro archivo XML de datos dentro de un grupo o en otros grupos, al cargar más de un grupo a la vez. Un analizador XML sustituirá la referencia externa por el contenido del archivo al que hace referencia la entidad de referencia externa.

Utilizando el ejemplo siguiente como guía para cargar un solo grupo de elementos de base de datos de tienda, puede crear el archivo XML de grupo de elementos de base de datos basándose en este extracto:

```
<?xml version="1.0"?>
<!DOCTYPE import SYSTEM "store-assets.dtd">
<import>
&store.xml;
&en_US_store.xml;
&fr_FR_store.xml;
</import>
```

donde

- `import` es el elemento raíz del documento XML. El elemento raíz ya se ha definido en el archivo wcs.dtd, proporcionado con WebSphere Commerce, e

incluye las definiciones para todas las tablas de la base de datos de WebSphere Commerce. Sin embargo, si ha personalizado el esquema de WebSphere Commerce, puede que necesite utilizar un elemento raíz diferente. Puede generar un nuevo archivo DTD que refleje el esquema personalizado o puede actualizar el archivo `wcs.dtd`.

- `store-assets.dtd` hace referencia al nombre del archivo DTD principal de grupo de elementos de base de datos que creará en el paso siguiente.
 - `&store.xml`; es una referencia de entidad XML al archivo XML de grupo de elementos de base de datos. La vía de acceso y ubicación se definen en el archivo DTD de grupo de elementos de base de datos. Este nombre cambiará para coincidir con los archivos de elementos ya creados para cada grupo.
 - `entorno_nacional_store.xml`; es necesario para cada idioma que soporte la tienda. Si la tienda es monolingüe, entonces solamente hará referencia a un archivo. Si la tienda soporta más de un idioma, entonces necesitará una referencia para cada idioma. El extracto anterior supone que la tienda soporta los idiomas inglés y francés.
5. Cree un archivo DTD principal de grupo de elementos de base de datos que defina las entidades de referencia anteriores y los demás archivos DTD que son necesarios para el grupo.

Utilizando el ejemplo siguiente como guía para cargar un solo grupo de elementos de base de datos de tienda, puede crear el archivo DTD principal de grupo de elementos de base de datos para cualquier grupo de datos:

```
<!ENTITY % wcs.dtd SYSTEM "vía completa para archivo wcs.dtd de WebSphere Commerce">
%wcs.dtd;
```

```
<!ENTITY %NonStoreForeignKeys.dtd SYSTEM "NonStoreForeignKeys.dtd">
%NonStoreForeignKeys.dtd;
<!ENTITY store.xml SYSTEM "store.xml">
<!ENTITY en_US_store.xml SYSTEM "en_US/store.xml">
<!ENTITY fr_FR_store.xml SYSTEM "fr_FR/store.xml">
```







donde

- `wcs.dtd` hace referencia al archivo DTD que contiene datos definidos fuera de su grupo de elementos de base de datos. Este archivo, que se proporciona con WebSphere Commerce, también resuelve y define el elemento raíz utilizado en el archivo XML de grupo de elementos de base de datos.
- `NonStoreForeignKeys.dtd` hace referencia al archivo DTD que define elementos distintos del elemento raíz. Este archivo contiene todas las declaraciones y definiciones de referencias de entidad XML para las dependencias externas fuera del grupo de elementos de base de datos. Por ello, los archivos XML contienen referencias a valores de clave externa que no se crean como parte del grupo de elementos de base de datos y que deben cargarse en la base de datos antes que este grupo.

Nota: Asegúrese de que la vía de acceso esté identificada correctamente. En este ejemplo, el archivo está en el mismo directorio que el archivo DTD de grupo de elementos de base de datos.

- `store.xml`, `en_US_store.xml` y `fr_FR_store.xml` son las entidades de referencia externa utilizadas en el archivo XML de grupo de elementos de base de datos. Para utilizar la referencia, siga el convenio de referencia de entidad: `&nombre_alias`;
- `store.xml` hace referencia al archivo de datos del grupo del que se cargan los elementos de base de datos. Este nombre cambiará para coincidir con

los archivos de elementos de base de datos ya creados para cada grupo. Tenga en cuenta que los archivos XML específicos del entorno nacional se encuentran bajo el siguiente directorio:

-  *unidad:\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data\entorno nacional*
-  *unidad:\Archivos de
programa\WebSphere\CommerceServer\samplstores\
nombre tienda ejemplo\data\entorno nacional*
-  */usr/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data/entorno nacional/*
-   */opt/WebSphere/CommerceServer/samplstores/
nombre tienda ejemplo/data/entorno nacional/*
-  */QIBM/ProdData/WebCommerce/samplstores/
nombre tienda ejemplo/data/entorno nacional/*

- los archivos *vía de acceso_store.xml* son necesarios para cada idioma que soporte la tienda, y se encuentran bajo los directorios anteriores. Si la tienda es monolingüe, entonces solamente hará referencia a un archivo. Si la tienda soporta más de un idioma, entonces necesitará un archivo específico del entorno nacional para cada idioma. El extracto anterior supone que la tienda soporta los idiomas inglés y francés.

6. Cada grupo de elementos de base de datos necesita información definida fuera de su dominio o su conjunto de datos, puesto que cada grupo puede tener dependencias externas. Puede proporcionar estos datos en un archivo DTD. Por ejemplo, el grupo de elementos de base de datos de tienda tiene las dependencias externas siguientes:

```
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID,
bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID
```

Al cargar un grupo de datos o el conjunto entero de datos de la tienda, se deberán definir las dependencias externas siguientes de la base de datos de WebSphere Commerce. Para utilizar estos datos, siga la referencia de entidad XML correspondiente. Por ejemplo, para utilizar los datos definidos por la entidad *ffmcenter_id*, deberá escribir *&ffmcenter_id*; en el archivo XML. Utilizando como guía el ejemplo siguiente de grupo de elementos de base de datos de tienda, puede crear su archivo DTD basándose en este extracto, denominado *NonStoreForeignKeys.dtd*:

```
<!ENTITY es_ES "-1">
<!ENTITY fr_FR "-2">
<!ENTITY de_DE "-3">
<!ENTITY it_IT "-4">
<!ENTITY es_ES "-5">
<!ENTITY pt_BR "-6">
<!ENTITY zh_CN "-7">
<!ENTITY zh_TW "-8">
<!ENTITY ko_KR "-9">
<!ENTITY ja_JP "-10">
<!ENTITY MEMBER_ID "-2000">
<!ENTITY ffmcenter_id "10001">
```



donde

- *MEMBER_ID* es el número de referencia interno que identifica al propietario de la tienda.







- `ffmcenter` es el número de referencia del centro de despacho de pedidos de la tienda. Puesto que la tienda puede utilizar más de un centro de despacho de pedidos, se puede definir más de uno en el archivo `NonStoreForeignKeys.dtd`.
- *entorno nacional* es el número de referencia de WebSphere Commerce para cada entorno nacional (que se identifica mediante el país o región y el idioma). Los valores están ubicados en la tabla de base de datos `LANGUAGE`.

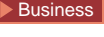
Nota: Si está dividiendo un archivador de tienda existente en grupos de elementos de base de datos, asegúrese de que todas las referencias, por ejemplo, al alias `@ffmcenter_id` se sustituyen por su referencia de entidad correspondiente: `&ffmcenter_id`;

7. Una vez que se hayan creado todos los archivos de datos, ejecute el mandato `IDResolve` para el archivo XML de grupo de elementos de base de datos para resolver los datos, tal como se describe en el apartado “Mandato ID Resolve” en la página 245.
8. Ejecute el mandato `Load` en el archivo de datos resueltos tal como se describe en el apartado “Mandato Load” en la página 253. Para comprobar el proceso de carga, consulte los archivos de anotaciones:

-  `idresgen.db2.log` y `massload.db2.log`
-  `idresgen.oracle.log` y `massload.oracle.log`

Los archivos de anotaciones se encuentran bajo:

-  `unidad:\WebSphere\CommerceServer\logs\`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\logs\`
-  `/usr/WebSphere/CommerceServer/logs/`
-   `/opt/WebSphere/CommerceServer/logs/`
-  `/QIBM/ProdData/WebCommerce/logs/`

9.  Ejecute el mandato `AccountImport` tal como se describe en el apartado “Publicación de elementos de cuenta de negocio” en la página 294.
10. Si es aplicable, publique los contratos tal como se describe en el apartado “Publicación de elementos de contrato” en la página 295.
11. Complete las tareas del apartado “Publicación de elementos de escaparate y archivos de configuración de tienda copiándolos en WebSphere Commerce Server” en la página 298

Capítulo 29. Publicación de cuentas de negocio y contratos

Loader package no puede cargar algunos de los elementos de base de datos de la tienda, (cuentas de negocio y contratos). Puede publicar estos elementos de base de datos utilizando los Servicios de tienda o desde la línea de mandatos, como parte de la publicación de una opción de tienda completa, como se describe en el Capítulo 26, “Publicación de una tienda completa” en la página 227, o puede publicar cuentas de negocio y contratos utilizando los mandatos correspondientes. Estos mandatos son los siguientes:

- **AccountImport**— Crea cuentas de negocio a partir del archivo `businessaccount.xml` del archivador de tienda.
- **ContractImportApprovedVersion**— Crea un contrato del archivo `contract.xml`. Si el contrato está en estado activo, el mandato crea y despliega el contrato. Aunque el archivo `contract.xml` contenga más de un contrato, sólo es necesario llamar al mandato una vez.
- **ProductSetPublish**— Sincroniza los datos del conjunto de productos en las tablas de base de datos de conjuntos de productos con el catálogo, antes de crear las cuentas de negocio y los contratos. La publicación a través de los Servicios de tienda o la línea de mandatos llama al mandato `ProductSetPublish` que, a su vez, llama a los mandatos `AccountImport` y `ContractImportApprovedVersion`.

Nota: Para obtener más información sobre estos mandatos, consulte la ayuda en línea de WebSphere Commerce.

Los elementos de cuenta de negocio están incluidos en forma de archivos XML en algunos de los archivadores de tienda de ejemplo proporcionados con WebSphere Commerce. Sin embargo, se recomienda que cree los elementos de cuenta de negocio utilizando las herramientas proporcionadas, en lugar de crear archivos XML para estos elementos. Para obtener más información sobre cómo crear estos elementos utilizando las herramientas proporcionadas, consulte la ayuda en línea de WebSphere Commerce. Las instrucciones para publicar cuentas de negocio se incluyen en las secciones siguientes, en el caso de que elija publicar los archivos XML correspondientes proporcionados con los archivadores de tienda de ejemplo o crear los suyos propios.

Nota: Si no utiliza los Servicios de tienda para publicar las cuentas de negocio o los contratos, para poder publicarlos debe primero publicar los elementos de tienda y de catálogo. Concretamente, necesita los identificadores de tienda y de catálogo, así como el ID de la organización propietaria de la tienda y los ID de las organizaciones compradoras asociadas al contrato, si las hay. Si los términos y condiciones de su contrato no especifican un catálogo, no es necesario publicar un catálogo antes de publicar una cuenta de negocio o un contrato.

Si publica estos elementos utilizando la publicación de los Servicios de tienda o de la línea de mandatos, asegúrese de seleccionar la opción de catálogo o de que la tienda ya tenga un catálogo publicado. Si publica estos elementos utilizando los mandatos correspondientes, asegúrese de que ya ha cargado los elementos listados más arriba en la base de datos.

Publicación de cuentas de negocio y contratos utilizando los Servicios de tienda o la línea de mandatos

Las cuentas de negocio y los contratos pueden publicarse utilizando los Servicios de tienda o el programa de utilidad de publicación desde la línea de mandatos. Para publicar las cuentas de negocio o los contratos utilizando los Servicios de tienda o la línea de mandatos, los elementos deben estar empaquetados en formato de archivador de tienda. Para obtener más información sobre cómo empaquetar los elementos de escaparate como un archivador de tienda, consulte la Parte 6, “Empaquetado de la tienda” en la página 215.

Mediante la publicación utilizando los Servicios de tienda o la línea de mandatos, puede elegir publicar todos los tipos de elementos del archivador de tienda (incluidos los elementos de escaparate, elementos de datos de tienda y paquetes de recursos) o puede elegir publicar sólo un tipo de elementos. Para obtener instrucciones paso a paso sobre cómo publicar elementos utilizando los Servicios de tienda o la línea de mandatos, consulte la ayuda en línea de WebSphere Commerce.







Publicación de cuentas de negocio y contratos utilizando mandatos

Si prefiere no empaquetar los elementos como un archivador de tienda, puede publicar las cuentas de negocio y los contratos utilizando los mandatos correspondientes:

- **AccountImport**— Crea cuentas de negocio a partir del archivo `businessaccount.xml` del archivador de tienda.
- **ContractImportApprovedVersion**— Importa un contrato aprobado o activo, de un archivo XML a WebSphere Commerce Server. Antes de importar el contrato, el mandato comprueba que el mandato que se importa contenga los términos y condiciones necesarios, y que sea un contrato válido.
- **ProductSetPublish**— Sincroniza los datos del conjunto de productos en las tablas de base de datos de conjuntos de productos con el catálogo, antes de crear las cuentas de negocio y los contratos. La publicación a través de los Servicios de tienda o la línea de mandatos llama al mandato `ProductSetPublish` que, a su vez, llama a los mandatos `AccountImport` y `ContractImportApprovedVersion`.

Publicación de elementos de cuenta de negocio

Para publicar los elementos de cuenta de negocio, realice lo siguiente:

1. Utilizando la Consola de administración, actualice el registro de vistas o reinicie la instancia de WebSphere Commerce. Para obtener más información, consulte la ayuda en línea de WebSphere Commerce.
2. Copie `businessaccount.xml` en el directorio siguiente:
 -  `unidad:\WebSphere\CommerceServer\xml\trading`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\trading`
 -  `/usr/WebSphere/CommereServer/xml/trading`
 -  `/opt/WebSphere/CommerceServer/xml/trading`
 -  `/opt/WebSphere/CommerceServer/xml/trading`
 -  `/QIBM/UserData/WebCommerce/instances/instancename/xml/trading`
3. Abra `businessaccount.xml` y realice los cambios siguientes:

- Sustituya todas las apariciones de `&STORE_IDENTIFIER`; por el identificador de la tienda.
- Sustituya todas las apariciones de `&MEMBER_IDENTIFIER`; por el nombre distinguido de miembro de la tienda.

Nota: Si trabaja con un archivo `businessaccount.xml` que forma parte de un archivador de tienda creado utilizando los Servicios de tienda, este paso ya se habrá completado.

4. Guarde y cierre el archivo.
5. Abra la Consola de administración. Conéctese como administrador.
6. En un navegador, escriba lo siguiente:
 - `https://nombresistpral:8000/webapp/wcs/stores/servlet/AccountImport?fileName=businessaccount.xml`
&URL=URL a redirigir después de completar satisfactoriamente







Nota: Para obtener más información sobre la sintaxis del mandato y los parámetros, consulte la ayuda en línea de WebSphere Commerce.

Publicación de elementos de contrato

Para publicar los elementos de contrato, realice lo siguiente:


1. Copie `contract.xml` en el directorio siguiente:

Nota: Esta vía de acceso es configurable. A continuación se muestra la vía de acceso por omisión.

-  `unidad:\WebSphere\CommerceServer\xml\trading`
 -  `unidad:\Archivos de programa\WebSphere\CommerceServer\xml\trading`
 -  `/usr/WebSphere/CommerceServer/xml/trading`
 -  `/opt/WebSphere/CommerceServer/xml/trading`
 -  `/opt/WebSphere/CommerceServer/xml/trading`
 -  `/QIBM/UserData/WebCommerce/instancias/nombreinstancia/xml/trading`
2. Abra `contract.xml` y realice los cambios siguientes:
 - Sustituya todas las apariciones de `&STORE_IDENTIFIER`; por el identificador de la tienda.
 - Sustituya todas las apariciones de `&MEMBER_IDENTIFIER`; por el nombre distinguido de miembro de la tienda.
 3. Guarde y cierre el archivo.
 4. Abra la Consola de administración. Conéctese como administrador.
 5. En un navegador, escriba lo siguiente:
 - `https://nombresistpral:número/webapp/wcs/tools/servlet/ContractImportApprovedVersion?fileName=contract.xml`
&targetStoreId=store_id&URL=ContractDisplay
 6. Si la tienda contiene múltiples archivos `contract.xml` (por ejemplo, archivos de contrato específicos del entorno nacional), repita los pasos 1 a 5 para cada archivo `contract.xml`.

Capítulo 30. Publicación de elementos de escaparate y de archivos de configuración de tienda

La publicación de los elementos de escaparate, los archivos HTML y JSP, los archivos de propiedades o paquetes de recursos, y las imágenes y los gráficos que crean las páginas de la tienda, forma parte del proceso de creación de una tienda operativa. Puede publicar los elementos de escaparate utilizando los Servicios de tienda o desde la línea de mandatos, como parte de la publicación de una opción de tienda completa, que se describe en el Capítulo 26, "Publicación de una tienda completa" en la página 227, o puede publicar los elementos de escaparate simplemente copiando los elementos en una ubicación específica de WebSphere Commerce Server.

Si publica los archivos JSP contenidos en las tiendas de ejemplo  ToolTech y NewFashion, y piensa configurar la tienda para utilizar las características de colaboración, también necesitará publicar los archivos de configuración de tienda que forman parte de ese archivador de tienda. Ambas tiendas contienen los archivos de configuración de tienda siguientes:

- tools_properties.zip
- tools_xml.zip
- runtime_xml.zip

Si publica una tienda completa utilizando los Servicios de tienda o desde la línea de mandatos (seleccionando todas las opciones de publicación), los archivos de configuración de tienda también se publicarán. Sin embargo, si elige publicar los elementos de escaparate copiándolos en WebSphere Commerce Server, también tendrá que copiar los archivos de configuración de tienda en WebSphere Commerce Server.

Publicación de elementos de escaparate y de archivos de configuración de tienda utilizando los Servicios de tienda o la línea de mandatos

Puede publicar los elementos de escaparate y los archivos de configuración de tienda utilizando los Servicios de tienda o el programa de utilidad de publicación desde la línea de mandatos.

Nota: Para publicar los archivos de configuración de tienda utilizando los Servicios de tienda o el programa de utilidad de publicación, deberá publicar todos los elementos de tienda, incluidos los elementos Web y los elementos de datos. No puede elegir publicar sólo los archivos de configuración de tienda.

Para publicar los elementos de escaparate y los archivos de configuración de tienda utilizando los Servicios de tienda o la línea de mandatos, los elementos de escaparate y los archivos de configuración de tienda deben estar empaquetados en formato de archivador de tienda. Para obtener más información sobre cómo empaquetar los elementos de escaparate como un archivador de tienda, consulte la Parte 6, "Empaquetado de la tienda" en la página 215.

Si utiliza la publicación mediante los Servicios de tienda o la línea de mandatos, puede elegir publicar todos los tipos de elementos del archivador de tienda (incluidos los elementos de escaparate, elementos de datos de tienda y paquetes de recursos) o puede elegir publicar sólo un tipo de elementos. Para obtener instrucciones paso a paso sobre cómo publicar elementos utilizando los Servicios de tienda o la línea de mandatos, consulte la ayuda en línea de WebSphere Commerce.

Publicación de elementos de escaparate y archivos de configuración de tienda copiándolos en WebSphere Commerce Server

Si prefiere no empaquetar los elementos como un archivador de tienda, puede publicar de todos modos los elementos de escaparate copiándolos directamente en WebSphere Commerce Server. Los elementos Web (HTML, archivos JSP, imágenes y gráficos) deben copiarse en el directorio raíz de documentos de la aplicación Web. Los paquetes de recursos o los archivos de propiedades deben copiarse en la vía de acceso de propiedades de las aplicaciones.



Para copiar los elementos de escaparate y los archivos de configuración de tienda en WebSphere Commerce Server, realice lo siguiente:

1. Copie los archivos JSP, HTML, archivos de inclusión, imágenes y gráficos en el directorio de la tienda (*dir tienda*) en el directorio raíz de documentos de la aplicación Web de tiendas:

-  unidad:\WebSphere\AppServer\installedApps\
WC_Enterprise_App_nombreinstancia.ear\wcstores.war\dir tienda
-  unidad:\Archivos de programa\WebSphere\AppServer\
installedApps\WC_Enterprise_App_nombreinstancia.ear\wcstores.war\
dir tienda
-  /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
-  /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
-  /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda
-  /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/dir tienda

donde *dir tienda* es el valor de la columna DIRECTORY de la tabla de base de datos STORE. Si este valor no existe, puede añadir un valor haciendo lo siguiente: `select directory from store=añadir nombre directorio relativo para publicar elementos de archivo`

2. Copie los paquetes de recursos y los archivos de propiedades en la vía de acceso de propiedades de la aplicación:

-  unidad:\WebSphere\AppServer\installedApps\
WC_Enterprise_App_nombreinstancia.ear\wcstores.war\WEB-
INF\classes\dir tienda
-  unidad:\Archivos de programa\WebSphere\AppServer\
installedApps\WC_Enterprise_App_nombreinstancia.ear\wcstores.war\WEB-
INF\classes\dir tienda







- **AIX** /usr/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
 - **Solaris** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
 - **Linux** /opt/WebSphere/AppServer/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
 - **400** /QIBM/UserData/WebASAdv4/WASinstancename/installedApps/
WC_Enterprise_App_nombreinstancia.ear/wcstores.war/WEB-INF/classes/dirtienda
3. Copie los archivos de configuración de tienda en las ubicaciones definidas en el archivo de configuración de WebSphere Commerce, nombre_instancia.xml. Este archivo está ubicado en el directorio siguiente:

- **NT** unidad:\WebSphere\CommerceServer\instances
nombreinstancia.xml
- **2000** unidad:\Archivos de programa\WebSphere\CommerceServer
instances\nombreinstancia.xml
- **AIX** /usr/WebSphere/CommerceServer/instances/
nombreinstancia.xml
- **Solaris** /opt/WebSphere/CommerceServer/instances/
nombreinstancia.xml
- **Linux** /opt/WebSphere/CommerceServer/instances/
nombreinstancia.xml
- **400** /QIBM/UserData/WebCommerce/instances/
nombreinstancia.xml

Los archivos de configuración de la tienda se copian en las ubicaciones siguientes:

- runtime_xml.zip se copia en StoresXMLPath. Esta vía de acceso está definida en el archivo de configuración de WebSphere Commerce, nombre_instancia.xml.
 - tools_properties.zip se copia en ToolsStoresPropertiesPath. Esta vía de acceso está definida en el archivo de configuración de WebSphere Commerce, nombre_instancia.xml.
 - tools_xml.zip se copia en ToolsStoresXMLPath. Esta vía de acceso está definida en el archivo de configuración de WebSphere Commerce, nombre_instancia.xml.
4. Inicie la tienda utilizando uno de los siguientes métodos:
- Utilice el mandato StoreCatalogDisplay:
StoreCatalogDisplay?storeId=idTienda&catalogId=idCatálogo&langId=idIdioma
donde
 - idTienda es el valor que se encuentra en la columna STORE_ID de la tabla de base de datos STORE,
 - idCatálogo es el valor que se encuentra en la columna CATALOG_ID de la tabla de base de datos CATALOG,

- idIdioma es el valor de la columna LANGUAGE_ID de la tabla de base de datos LANGUAGE para un entorno nacional específico. Para obtener una lista de los valores de WebSphere Commerce por omisión, consulte la tabla de base de datos LANGUAGE.
- Si la tienda se basa en una tienda de ejemplo de WebSphere Commerce, configure el URL de la tienda editando el archivo `index.jsp` bajo:

-  `unidad:\WebSphere\CommerceServer\wc.ear\wcstores.war\dirienda`
-  `unidad:\Archivos de programa\WebSphere\CommerceServer\wc.ear\wcstores.war\dirienda`
-  `/usr/WebSphere/CommerceServer/wc.ear/wcstores.war/dirienda`
-   `/opt/WebSphere/CommerceServer/wc.ear/wcstores.war/dirienda`
-  `/QIBM/ProdData/WebCommerce/wc.ear/wcstores.war/dirienda`

Añada los valores correctos para los parámetros siguientes:

- nombresistpral es el nombre totalmente calificado de la máquina de WebSphere Commerce,
- idTienda es el valor que se encuentra en la columna STORE_ID de la tabla de base de datos STORE,
- idCatálogo es el valor que se encuentra en la columna CATALOG_ID de la tabla de base de datos CATALOG,
- idIdioma es el valor de la columna LANGUAGE_ID de la tabla de base de datos LANGUAGE para un entorno nacional específico. Para obtener una lista de los valores de WebSphere Commerce por omisión, consulte la tabla de base de datos LANGUAGE.

Para ver la tienda en un navegador, inicie el siguiente URL: `http://nombresistpral/webapp/wcs/stores/servlet/dirienda/index.jsp`

Parte 8. Adición de características de WebSphere Commerce a la tienda

Para añadir ciertas características disponibles en WebSphere Commerce a la tienda, debe completar algunos pasos manualmente. Los capítulos de esta sección describen cómo se añaden las siguientes características a la tienda:

- Capítulo 31, “Adición del servicio de atención al cliente a la tienda” en la página 303
- Capítulo 32, “Adición de zonas de e-marketing a la tienda” en la página 317

Capítulo 31. Adición del servicio de atención al cliente a la tienda

La característica de atención al cliente de WebSphere Commerce proporciona soporte para el servicio al cliente en tiempo real mediante una interfaz de texto síncrona que utiliza el servidor Lotus Sametime. Cuando está habilitado el servicio de atención al cliente en la tienda, el cliente puede entrar en la tienda, pulsar un enlace y conectarse con un representante de servicio al cliente (CSR). A continuación, el cliente puede comunicarse con un CSR a través de Internet.

Nota: Este capítulo describe cómo activar el servicio de atención al cliente en la tienda. Sin embargo, para poder habilitar el servicio de atención al cliente en la tienda, deberá instalar primero un servidor Sametime y configurarlo para que funcione con WebSphere Commerce. Para obtener más información, consulte la publicación *WebSphere Commerce, Guía de software adicional*. También deberá registrar los CSR en la Consola de administración para permitirles utilizar el servicio de atención al cliente. Para obtener más información sobre esta tarea, así como sobre los conceptos generales del servicio de atención al cliente y cómo emplea un CSR el servicio de atención al cliente, consulte la ayuda en línea de WebSphere Commerce.

Nota:

Puede habilitar el servicio de atención al cliente en la tienda de un modo fácil y rápido utilizando los Servicios de tienda, si crea la tienda basándose en una de las tiendas de ejemplo siguientes:  ToolTech y NewFashion. Después de publicar la tienda utilizando los Servicios de tienda, seleccione la vista Tiendas, seleccione la tienda, a continuación seleccione **Configurar** y habilite las características de servicio de atención al cliente. Para obtener instrucciones detalladas, consulte la ayuda en línea de WebSphere Commerce.

Sin embargo, si no crea la tienda utilizando como base una tienda de ejemplo, tendrá que realizar algunas tareas para habilitar el servicio de atención al cliente en su tienda. El resto de este capítulo describe los conceptos y los pasos necesarios para activar el servicio de atención al cliente en una tienda que no esté basada en uno de los ejemplos.

Nota: Las tiendas de ejemplo  ToolTech y NewFashion muestran cómo debe implementarse el servicio de atención al cliente y proporcionan el código que se puede utilizar en la tienda para habilitar el servicio de atención al cliente. Este capítulo hace referencia a estas dos tiendas de ejemplo para ilustrar cómo puede activar el servicio de atención al cliente en la tienda. Cuando lea este capítulo, asegúrese de tener la versión más reciente de las tiendas de ejemplo (disponibles en el sitio Web del producto WebSphere Commerce).

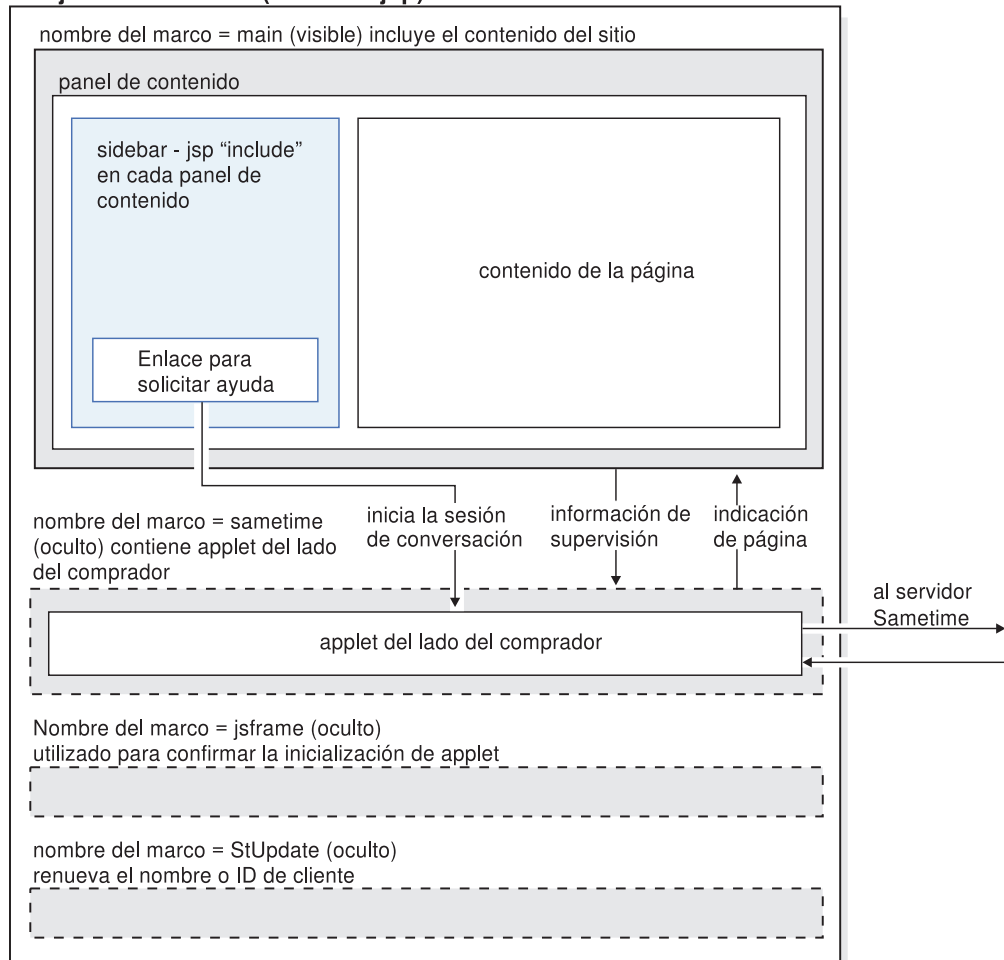
Descripción del servicio de atención al cliente en una tienda

Cuando un cliente selecciona el enlace del servicio de atención al cliente, por ejemplo **Conversación en directo con Asistencia al cliente**, en una tienda que tiene activado el servicio de atención al cliente, se inicia un applet que contiene la ventana para la conversación. Este applet se ejecuta dentro de un conjunto de

marcos ocultos que no interfiere con el diseño externo del sitio. Cuando se inicia el applet, se conecta con un servidor Lotus Sametime.

El diagrama siguiente muestra el diseño del conjunto de marcos.

conjunto de marcos (en index.jsp)



El conjunto de marcos incluye cuatro marcos:

- Main: el marco que contiene el contenido de la tienda, incluidos los archivos que crean las páginas de la tienda, es decir, los archivos que crean los archivos de cabecera, pie de página, el texto de la página y los archivos sidebar. Los visitantes de la tienda pueden ver el contenido de este marco. Tenga en cuenta que el marco principal (main) contiene las conexiones siguientes con el marco Sametime: un enlace con el servicio de atención al cliente e información de supervisión. La información de supervisión se describe detalladamente en el apartado "Supervisión de los clientes que utilizan el servicio de atención al cliente" en la página 306.
- Sametime: el marco que contiene el applet del servicio de atención al cliente. Este marco no está visible para los visitantes de la tienda. Sin embargo, cuando un cliente pulsa el enlace para iniciar el applet, el cliente verá la ventana del servicio de atención al cliente. Este marco también pasa información al marco principal (main), mediante la característica de indicación de página.
- jsframe: el marco que confirma que el applet se ha cargado correctamente. El contenido de este marco no se muestra a los clientes.
- StUpdate: El marco renueva el nombre o ID de cliente.

Utilización del conjunto de marcos

Al iniciar el applet de servicio al cliente en un conjunto de marcos se separa el código del applet del código de las páginas de la tienda. Como muestra el diagrama anterior, las páginas de la tienda están contenidas en el marco principal del conjunto de marcos, mientras que el código del applet está en el marco Sametime. Al separar el código del applet de las páginas de la tienda se disminuye el tráfico de la red, ya que el applet solamente se baja una vez, cuando se inicia por primera vez el conjunto de marcos. Si el applet del servicio de atención al cliente no formara parte del conjunto de marcos tendría que estar en todas las páginas de la tienda y sería necesario bajarla cada vez que se accediera a una página nueva.

Al utilizar un conjunto de marcos se puede mantener la conexión con el servidor Sametime. Si el applet formara parte de cada página y no del conjunto de marcos, se crearía una nueva sesión Sametime cada vez que el cliente accediera a una página nueva. Dado que el applet del servicio de atención al cliente se conecta al servidor Sametime de forma anónima, crear una sesión nueva cada vez que un cliente accede a una página nueva no permitiría hacer un seguimiento de las actividades de los clientes en la tienda. Al utilizar el conjunto de marcos, se mantiene la sesión Sametime original del cliente y las actividades del cliente se devuelven al servidor Sametime a medida que cambian los atributos.

Aspectos relacionados con la utilización de conjuntos de marcos

Aunque el método recomendado para implementar el servicio de atención al cliente en la tienda es utilizar un conjunto de marcos, debe tener en cuenta los aspectos siguientes relacionados con la utilización de conjuntos de marcos:

- Un sólo punto de entrada: los clientes solamente pueden utilizar el servicio de atención al cliente si navegan por la tienda dentro de la infraestructura. Del mismo modo, los representantes del servicio al cliente solamente pueden supervisar el movimiento de los clientes a través del conjunto de marcos. Para garantizar que los clientes naveguen por la tienda mediante el conjunto de marcos, deben acceder al sitio mediante un sólo punto de entrada, por ejemplo, a través de la página de presentación de la tienda (en el caso de las tiendas de ejemplo, `index.jsp`). Si un cliente accede a la tienda a través de otra página, por ejemplo, a través de una página de catálogo, no estará en el conjunto de marcos.
- Marcadores (bookmarks): cuando los clientes utilicen el conjunto de marcos solamente podrán obtener los marcadores del URL principal del sitio, no de las páginas individuales.
- Renovación: cuando un cliente esté en el conjunto de marcos y pulse renovar, se le devolverá a la dirección del marco principal, como se ha codificado en el conjunto de marcos, por ejemplo, `index.jsp`.
- Ajuste del tamaño de la ventana del navegador: Si un cliente ajusta el tamaño de la ventana del navegador mientras está en el conjunto de marcos, puede que el navegador vuelva a cargar automáticamente la dirección de entrada. Si se vuelve a cargar la dirección de entrada, puede que termine la conexión con el servidor Sametime. En esta situación, los diferentes navegadores se comportan de forma distinta.
- Seguridad: Cuando un cliente está examinando un sitio a través de un conjunto de marcos, cada marco individual, así como el conjunto de marcos (el URL de la barra de ubicación), mantiene su propia conexión, ya sea insegura (`http`, por omisión el puerto 80) ya sea segura (`https`, por omisión el puerto 443). Si un cliente está examinando la tienda a través de una conexión insegura, todos los marcos del conjunto de marcos están en HTTP. En este caso no hay aspectos relacionados con SSL. Sin embargo, si el cliente navega a una página segura (por

ejemplo, la página de registro), el marco principal (main) del conjunto de marcos se conmutará a HTTPS, mientras el resto de los marcos siguen siendo no seguros (http). En este caso, el cliente no podrá iniciar el applet del servicio de atención al cliente. El navegador no autorizará el inicio del applet, porque el applet (seguro, puerto 443) parece proceder de un servidor diferente del URL de la barra de ubicación del navegador (HTTP, puerto 80). El applet no se podrá iniciar hasta que el conjunto de marcos entero vuelva a ser seguro, es decir, hasta que el URL de la barra de ubicación apunte a HTTPS. Para que todo el conjunto de marcos vuelva a ser seguro, tiene las opciones siguientes:

- Redirigir todo el conjunto de marcos a una conexión segura. Sin embargo, al hacerlo finalizará cualquier conversación que esté en proceso en ese momento, ya que el applet pasa a una nueva conexión Sametime.
- Redirigir todo el conjunto de marcos a una conexión segura cuando se entre en el sitio. Esto disminuye ligeramente el rendimiento pero aumenta la seguridad y la confidencialidad de la sesión del cliente.

Un método de redireccionamiento del conjunto de marcos a una conexión segura consiste en añadir el código siguiente al archivo index.html:

```
- <html>
  <head>
    <META HTTP-EQUIV=Refresh CONTENT="0;URL=https://sistpral/webapp/wcs/stores/
    servlet/NewFashion/index.jsp>
  </head>
</html>
</head>
```

Supervisión de los clientes que utilizan el servicio de atención al cliente

El servicio de atención al cliente le permite supervisar qué clientes se comunican con el representante del servicio al cliente de la tienda de uno de los modos siguientes:

- Obteniendo el nombre o el ID del cliente
- Determinando en qué página está navegando el cliente
- Realizando un seguimiento de los artículos del carro de la compra

Para obtener esta información, se añade el código personalizado a las páginas de la tienda. Los apartados siguientes describen cómo se implementan estas características de supervisión en las tiendas de ejemplo.

Obtención del nombre o el ID del cliente

Cuando se inicia el applet del servicio de atención al cliente y el representante del servicio de atención al cliente (CSR) está conectado, éste puede identificar quién está utilizando el applet según el nombre o el ID del comprador. Las tiendas de ejemplo incluyen un código especializado que funciona con el applet del servicio de atención al cliente para determinar el nombre o el ID del comprador. Este código determina si se trata de un cliente invitado, un cliente invitado que tiene artículos en un carro de la compra o un cliente registrado, a continuación, asigna un nombre o ID al cliente y lo pasa al applet del servicio de atención al cliente. A continuación, estos nombres se muestran al CSR. Por ejemplo, si el cliente es un comprador invitado que todavía no ha colocado nada en el carro de la compra, se asigna al cliente un ID generado con el ID de comprador -1002. Si el cliente es un comprador invitado que tiene artículos en el carro de la compra, se mostrará el ID del comprador y si el cliente está registrado, se visualizarán su nombre y sus apellidos.

Las tiendas de ejemplo obtienen el nombre o ID del cliente añadiendo el código siguiente al archivo de cabecera de la tienda que renueva el marco StUpdate. Este código se incluye en header.jsp de NewFashion y en NavHeader.jsp de

 ToolTech.

Nota: Cada vez que un cliente navega por una página nueva de la tienda, se renueva el nombre o el ID del cliente.


```
<script language="javascript">
  if (typeof top.updateStInfo == 'function')
    top.updateStInfo();
</script>
```

El código anterior renueva lo siguiente en el marco StUpdate.

```
//set Customer Name for LiveHelp if user is registered.

if (userRegistrationDataBean.findUser()) {
if (userRegistrationDataBean.getLastName() != null &&
userRegistrationDataBean.getLastName().length() > 0) {
  if(cmdcontext != null) {
    Long uid = cmdcontext.getUserId();
    String customerName = "";
    if (locale.toString().equals("ja_JP")||locale.toString().equals("ko_KR")
||locale.toString().equals("zh_CN")||locale.toString().equals("zh_TW"))
    {
      customerName = "" + userRegistrationDataBean.getLastName() + " "
+ userRegistrationDataBean.getFirstName();
    }
    else {
      customerName = "" + userRegistrationDataBean.getFirstName() + " "
+ userRegistrationDataBean.getLastName();
    }
  }
  else {
    customerName=userRegistrationDataBean.getUserId();
    if (customerName.equals("-1002"))
      customerName="";
    customer_name=customer_name.trim();
  }
}
```

En la página Desconexión de la tienda de ejemplo se incluye más código del cliente, que establece el nombre del cliente en un ID generado y restablece el número de artículos del carro de la compra a cero. La página Desconexión de

NewFashion es LoginForm.jsp. En  ToolTech es Logoff.jsp. El código del cliente es el siguiente:

```
<HTML>
<HEAD>
<SCRIPT language="javascript">
  if (typeof parent.setCustomerName == 'function')
    parent.setCustomerName (parent.WCSGUESTID, '')
  if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>
```

Determinación de la página por la que navega el cliente

El servicio de atención al cliente también permite a los CSR determinar por qué página de la tienda están navegando actualmente los clientes. Las tiendas de ejemplo determinan en qué páginas están los clientes añadiendo el código siguiente al archivo de cabecera (header.jsp en NewFashion y NavHeader.jsp en

 ToolTech):

```

<%
//Determine Page Type for LiveHelp
String headerType = (String) request.getAttribute("liveHelpPageType");
if (headerType==null)
    headerType = "";
%>

<script language="javascript">
<%
    String pname = request.getRequestURI();
    int indpn = pname.lastIndexOf('/');
    indpn = pname.lastIndexOf('/', indpn-1);
    if(indpn != -1)
        pname = pname.substring(indpn+1);

    //Determine if this is a personal page or not
    if (headerType.equals("personal") ) {
%>
        if (typeof parent.setPageParams == 'function')
            parent.setPageParams('PERSONAL_URL', '<%=pname%>');
<% } else { %>
        if (typeof parent.setPageParams == 'function')
            parent.setPageParams(location.href, '<%=pname%>');
<% } %>
</script>

```

Si una página no utiliza el archivo de cabecera, se incluye el archivo StHeader1.jsp en la página. El archivo StHeader1.jsp contiene el mismo código que se añade al archivo de cabecera de la tienda.

Para mantener la confidencialidad del cliente, el CSR no debe tener acceso a determinadas páginas. Por ejemplo, es posible que un CSR no tenga acceso a una página de campaña, a una página que incluye un precio determinado por contrato o a una página que incluye el ID de usuario, por ejemplo, la página del listín. Estas páginas se marcan como personales. En las tiendas de ejemplo, las páginas siguientes están marcadas como personales:

- NewFashion
 - AddressBookForm.jsp
 - AllocationCheck.jsp
 - edit_registration.jsp
 - emptyshoppingcart.jsp
 - interestItemDisplay.jsp
 - myaccount.jsp
 - orderItemDisplay.jsp
 - OrderDisplayPending.jsp
 - ResultList.jsp
 - shoppingcart.jsp
 - TrackOrderStatus.jsp
- Business ToolTech
 - Address.jsp
 - Addressbook.jsp
 - AddToExistReqList.jsp
 - AdvancedSearch.jsp
 - AllocationCheck.jsp

- CatalogMainDisplay.jsp
- CatalogItemDisplay.jsp
- CatalogTopCategoriesDisplay.jsp
- Confirmation.jsp
- OrderDisplayPending.jsp
- OrderItemDisplay.jsp
- OrderDetail.jsp
- QuickOrder.jsp
- RequisitionListCreate.jsp
- RequisitionListDetailDisplay.jsp
- RequisitionListDisplay.jsp
- RequisitionListUpdate.jsp
- Result List.jsp
- Shipping.jsp
- shoppingcart.jsp
- TrackOrderStatus.jsp
- UserAccount.jsp
- UserRegistrationUpdate.jsp

Para poder marcar las páginas como personales, es decir, como no disponibles para el CSR, las tiendas de ejemplo incluyen el código siguiente en la página, justo antes de incluir la cabecera.

```
<%
// Establecer tipo de cabecera para esta JSP para servicio de atención
// al cliente. Debe establecerse antes de Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>

<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>" flush="true"/>
```

Nota: Aunque un CSR no puede ver el contenido de una página marcada como personal, el CSR puede ver el URL de dicha página.

Seguimiento del número de artículos del carro de la compra

El servicio de atención al cliente permite a los CSR realizar un seguimiento de la cantidad de artículos que un cliente tiene en el carro de la compra en cualquier momento. Las tiendas de ejemplo obtienen el número de artículos del carro de la compra de las ubicaciones siguientes:

- Página de carro de la compra (NewFashion: shoppingcart.jsp,
Business ToolTech:ShoppingCart.jsp)
- Página de carro de la compra vacío (NewFashion: emptyshopcart.jsp,
Business ToolTech:EmptyOrder.jsp)
- Página de confirmación de pedido (NewFashion: confirmation.jsp,
Business ToolTech:confirmation.jsp)
- Página de desconexión (Business ToolTech:Logoff.jsp)

Nota: Aunque el carro de la compra está diseñado como una página personal, los CSR pueden hacer el seguimiento del número de artículos del carro de la

compra. Utilizando este método no pueden ver qué contiene el carro de la compra, sólo el número de artículos que hay en él. Sin embargo, un CSR puede ver el contenido del carro de la compra utilizando el botón **Ver carro de la compra**. Para obtener más información, consulte la ayuda en línea de WebSphere Commerce.

Las tiendas de ejemplo determinan el número de artículos del carro de la compra añadiendo el código siguiente a las páginas anteriores:

- En primer lugar, se define una variable int

```
int liveHelpShoppingCartItems= 0;
```
- A continuación, se utiliza la siguiente línea de código para agregar la cantidad a liveHelpShoppingCartItems siempre que se añada un orderitem al carro de la compra:

```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```
- A continuación, se añade el código siguiente al final de la página para establecer el nombre del cliente en el ID del comprador invitado y para obtener el número de artículos del carro de la compra del cliente.

```
<script language="javascript">  
if (typeof parent.setShoppingCartItems == 'function')  
parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);  
</script>
```

El código siguiente se utiliza en la página de carro de la compra vacío y en la página de confirmación de pedido para restablecer el número de artículos del carro a cero:

```
<script language="javascript">  
if (typeof parent.setShoppingCartItems == 'function')  
parent.setShoppingCartItems(0);  
</script>
```

Adición del servicio de atención al cliente a la tienda

Para añadir el servicio de atención al cliente a una tienda que no esté basada en un ejemplo, efectúe lo siguiente:

Parte 1: Instalación de requisitos previos

Para que el servicio de atención al cliente funcione en la tienda, deberá realizar lo siguiente:







- Instalar un servidor Sametime. Para obtener más información, consulte la publicación *WebSphere Commerce, Guía de software adicional*.
- Instalar el paquete de integración de Sametime de WebSphere Commerce. Para obtener más información, consulte la publicación *WebSphere Commerce, Guía de software adicional*.
- Detener la instancia de WebSphere Commerce, habilitar Sametime en el Gestor de configuración y, a continuación, reiniciar la instancia. Para obtener más información, consulte la publicación *WebSphere Commerce, Guía de software adicional*.
- Crear un CSR y registrar el CSR para el servicio de atención al cliente utilizando la Consola de administración. Para obtener más información, consulte la ayuda en línea de WebSphere Commerce.

Parte 2: Copia de los archivos de integración del servicio de atención al cliente de la tienda de ejemplo

Las tiendas de ejemplo NewFashion y ToolTech incluyen los archivos siguientes que se utilizan para integrar el servicio de atención al cliente a la tienda:

- `Sametime.js`: Contiene las funciones JavaScript que se incluyen para todos los marcos. Se llama a las funciones de este archivo con un prefijo padre de las páginas del marco principal, por ejemplo, `parent.setCustomerName`.
- `StBlank.jsp`: Archivo JSP vacío.
- `StFrame.jsp`: Contiene funciones JavaScript e incorpora el applet de servicio al cliente para el escaparate.
- `StReadyJS.jsp`: Indica que el applet se ha cargado correctamente.
- `StHeader1.jsp`: Archivo de cabecera que pasa un parámetro al applet indicando si la página que incluye esta cabecera es una página personal o no.
- `StUpdate.jsp`: Actualiza la información de nombre y el ID del cliente.

Para copiar los archivos de integración Sametime de la tienda de ejemplo a su tienda, efectúe lo siguiente:

1. Localice el archivo del archivador de tienda de la tienda NewFashion o de la tienda ToolTech. Los archivos del archivador de tienda están situados en el directorio siguiente:
 -  unidad:\WebSphere\CommerceServer\samlestores
 -  unidad:\Archivos de programa\WebSphere\CommerceServer\samlestores
 -  /usr/WebSphere/CommerceServer/samlestores
 -  /opt/WebSphere/CommerceServer/samlestores
 -  /opt/WebSphere/CommerceServer/samlestores
 -  /qibm/proddata/WebCommerce/samlestores
2. Abra la carpeta ToolTech o NewFashion, luego seleccione un archivador de tienda ToolTech o NewFashion.
3. Abra el archivador de tienda utilizando WinZip o una herramienta similar.
4. Localice el archivo `webapp.zip`. Ábralo utilizando WinZip o una herramienta similar.
5. Seleccione los archivos siguientes:
 - `Sametime.js`
 - `StBlank.jsp`
 - `StFrame.jsp`
 - `StReadyJS.jsp`
 - `StHeader1.jsp`
 - `StUpdate.jsp`
6. Extraiga los archivos en el directorio que contiene los elementos web de la tienda.

Nota: `StHeader1.jsp` está en un directorio include de los archivadores de la tienda de ejemplo. Si tiene un directorio diferente para los archivos que incluye en las páginas de la tienda, guarde `StHeader1.jsp` en dicho directorio. Si no es así, guárdelo en el mismo directorio que los demás archivos de la tienda.

Parte 3: Adición del conjunto de marcos a la tienda

Como se ha descrito en el apartado "Utilización del conjunto de marcos" en la página 305, el applet del servicio de atención al cliente se ejecuta en un conjunto de marcos. Este conjunto de marcos se debe añadir a la página de presentación de la tienda o a la página que con mayor probabilidad utilizarán los clientes para entrar en la tienda. Para añadir el conjunto de marcos a la tienda, efectúe lo siguiente:

1. Determine qué página es el punto de entrada de la tienda.
2. Abra el archivo del archivador de tienda para ToolTech o NewFashion. Los archivos del archivador de tienda están situados en el directorio siguiente:
 - **NT** unidad:\WebSphere\CommerceServer\samplstores
 - **2000** unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
 - **AIX** /usr/WebSphere/CommerceServer/samplstores
 - **Solaris** /opt/WebSphere/CommerceServer/samplstores
 - **Linux** /opt/WebSphere/CommerceServer/samplstores
 - **400** /qibm/proddata/WebCommerce/samplstores
3. Abra el archivador de tienda utilizando WinZip o una herramienta similar.
4. Localice el archivo webapp.zip. Ábralo utilizando WinZip o una herramienta similar.
5. Abra el archivo index.jsp.
6. Copie el código siguiente:

```
<script src="<%= "Sametime.js"%></script>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"DTD/xhtml1-transitional.dtd">
<html>
<head>
<title></title>
<script language="javascript">
var MainPageURL="";
if (MainPageURL=="")
MainPageURL="/webapp/wcs/stores/servlet/Logoff?storeId=<%=storeId>
&langId=<%=langId>&URL=LogonForm?
storeId=<%=storeId>&catalogId=<%=catalogId%>";
function loadFrame()
{
main.document.location.href=MainPageURL;
}
</script>
</head>
<%
String sBasePath="/webapp/wcs/stores/servlet/";
String sSametimeUrl="StFrame.jsp?storeId="+storeId;
String sBlankUrl="StBlank.jsp?storeId="+storeId;
String sUpdateUrl="StUpdate.jsp?storeId="+storeId;
%>
<FRAMESET border=0 frameBorder=0 ROWS="100%,1,1,1,1,1" onLoad="loadFrame();">
<FRAME NAME="main"
SRC="javascript:top.loadFrame();" MARGINWIDTH=0 SCROLLING="Auto"
FRAMEBORDER="no" noresize>
<FRAME NAME="JSFrame"
SRC="<%=sBlankUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
<FRAME NAME="sametime" SRC="<%=sSametimeUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
```

```
<FRAME NAME="StUpdate" SRC="<%=sUpdateUrl%>" MARGINWIDTH=0 SCROLLING="no"
FRAMEBORDER="no" noresize>
</FRAMESET>
</html>
```

Nota: En el ejemplo anterior de la tienda de ejemplo ToolTech, el fuente (SRC) del marco principal ("main") es un mandato a otra página. Dependiendo del modo en que esté configurada la tienda, un SRC puede ser un mandato, un archivo JSP o un archivo HTML.

7. Pegue el código que ha copiado en el paso 6 en la página que ha designado como el punto de entrada de la tienda. Efectúe los cambios necesarios en el fuente (SRC) del marco principal.
8. Guarde el archivo.

Parte 4: Adición del código para obtener el nombre o ID del cliente

Para que el CSR pueda visualizar el nombre o el ID del comprador que utiliza el servicio de atención, efectúe lo siguiente:

1. Revise la información del apartado "Obtención del nombre o el ID del cliente" en la página 306.
2. Determine de dónde piensa obtener el nombre o el ID de comprador. Por ejemplo:
 - Del punto de entrada a la tienda
 - Del registro nuevo o actualizado
 - De la desconexión
 - De la cabecera
3. Después de determinar de dónde piensa obtener el nombre o el ID del comprador, determine si estas páginas incluyen el archivo de cabecera principal de la tienda. Si no es así, determine si añadirá el código necesario a la propia página o si incluirá un archivo de cabecera.
4. Desde el archivo webapp.zip del archivador de tienda NewFashion o desde el archivador de tienda ToolTech, abra uno de los archivos siguientes:
 - NewFashion: header.jsp
 - ToolTech: NavHeader.jsp

Nota: Ambos archivos están situados en el directorio include del archivo webapp.zip.

5. Copie el código siguiente:

```
<script language="javascript">
  if (typeof top.updateStInfo == 'function')
    top.updateStInfo()
</script>
```
6. Pegue el código que ha copiado en el paso 5 en el archivo de cabecera de la tienda o directamente en las páginas correspondientes.

Nota: Si lo prefiere, puede incluir el archivo StHeader1.jsp en la página adecuada, en lugar de copiar el código anterior en el archivo de cabecera de la tienda o directamente en el archivo.

7. Guarde los archivos.

8. (Opcional) Para restablecer el ID de cliente de la página de desconexión (por ejemplo, restablecer el ID de cliente a -1002) y para restablecer el número de artículos del carro de la compra a cero, añada el código siguiente al archivo de desconexión:

```
<HTML>
<HEAD>
<SCRIPT language="javascript">
  if (typeof parent.setCustomerName == 'function')
    parent.setCustomerName (parent.WCSGUESTID, '')
  if (typeof parent.setShoppingCartItems == 'function')
    parent.setShoppingCartItems(0);
</SCRIPT>
</HEAD>
</HTML>
```

Parte 5: Adición de código para determinar qué página está examinando el cliente

Para determinar por qué página está navegando el cliente, efectúe lo siguiente:

1. Incluya el archivo StHeader1.jsp en el archivo de cabecera de la tienda, por ejemplo:

```
<%@ include file="StHeader1.jsp" %>
```

2. Añada el código siguiente a cualquier página que deba marcarse como personal y a la que, por lo tanto, no pueda acceder el CSR:

```
<%
// Establecer tipo de cabecera para esta JSP para servicio de atención
// al cliente. Debe establecerse antes de Header.jsp
request.setAttribute("liveHelpPageType", "personal");
%>
```

```
<%
String incfile;
incfile = includeDir + "Header.jsp";
%>
<jsp:include="<%=incfile%>" flush="true"/>
```

3. Añada el código siguiente a cualquier página que no utilice una cabecera y no deba marcarse como personal:

```
<%
// Establecer tipo de cabecera para esta JSP para servicio de atención
// al cliente. Debe establecerse antes de StHeader1.jsp
request.setAttribute("liveHelpPageType", "personal");

String incfile;
incfile = includeDir + "StHeader1.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>
```

Parte 6: Adición de código para hacer el seguimiento del número de artículos del carro de la compra

Para que los CSR puedan realizar un seguimiento de los artículos que un cliente tiene en el carro de la compra, efectúe lo siguiente:

1. Revise la información del apartado “Seguimiento del número de artículos del carro de la compra” en la página 309.
2. Determine dónde piensa realizar el seguimiento de los artículos del carro de la compra. Por ejemplo:
 - Página del carro de la compra
 - Página del carro de la compra vacío

- Página de confirmación de pedidos
 - Página de desconexión
3. En las páginas en las que piensa realizar el seguimiento de los artículos del carro de la compra, efectúe lo siguiente:
 - a. Defina una variable int:


```
int liveHelpShoppingCartItems= 0;
```
 - b. Para añadir la cantidad a shoppingCartItems cuando se añade un artículo de pedido al carro, añada la línea de código siguiente:


```
liveHelpShoppingCartItems+= orderItem.getQuantityInEJBType().intValue();
```
 - c. Al final de la página, añada el código siguiente para establecer el ID de comprador invitado (-1002) en el ID de comprador real y para obtener el número de artículos del carro de la compra:


```
<script language="javascript">
if (typeof parent.setCustomerName == 'function')
  parent.setCustomerName(<%=cmdcontext.getUserId()%>, parent.CustomerName);
if (typeof parent.setShoppingCartItems == 'function')
  parent.setShoppingCartItems(<%=liveHelpShoppingCartItems%>);
</script>
```
 - d. Si piensa realizar un seguimiento de los datos de la página del carro de la compra vacío y de la página de confirmación de pedidos, añada el código siguiente a dichas páginas para restablecer el valor del carro de la compra a cero:


```
<script language="javascript">
if (typeof parent.setShoppingCartItems == 'function')
  parent.setShoppingCartItems(0);
</script>
```

Parte 7: Adición de un enlace al servicio de atención al cliente

Para que los clientes puedan acceder al servicio de atención al cliente, efectúe lo siguiente:

1. Determine dónde desea colocar el enlace con el servicio de atención al cliente. Por ejemplo, es posible que desee colocar el enlace en una barra de navegación, de modo que los clientes puedan verlo siempre o en determinadas páginas de la tienda.
2. Copie el código siguiente en las páginas que contendrán el enlace:



```
<a href="javascript:if((parent.sametime != null)) top.interact();">
<%=infashiontext.getString("LiveHelp")%></a>
```

Parte 8: Cambio de mensajes que se muestran al cliente

Los mensajes que se muestran a un cliente cuando éste se conecta inicialmente a un CSR, por ejemplo "Hola, ¿en qué puedo ayudarle?" o "Nuestro horario de oficina es de 9 a 21", se almacenan en el archivo de propiedades del servidor Sametime. Los archivos de propiedades se dividen en dos tipos de archivos: Customer.properties y Agent.properties. El archivo Customer.properties contiene mensajes que se muestran al cliente, mientras que el archivo Agent.properties contiene información que se muestra al CSR. Estos dos archivos también tienen archivos correspondientes específicos de entorno nacional, por ejemplo Customer_de_DE.properties y Agent_de_DE.properties, para cada entorno nacional instalado en la instancia de WebSphere Commerce.

Para cambiar los mensajes de estos archivos, realice lo siguiente:

1. Localice los archivos de propiedades en el servidor Sametime. Por omisión, estos archivos de propiedades están ubicados en el directorio siguiente:

-  NT unidad:\Sametime\Data\domino\html\wc\properties
2. Realice los cambios necesarios.
 3. Cierre el archivo y guárdelo.

Capítulo 32. Adición de zonas de e-marketing a la tienda

Las zonas de e-marketing reservan espacio en las páginas de tienda en las que se visualiza el contenido de marketing personalizado para iniciativas de campaña. Cuando un cliente solicita una página, las zonas de e-marketing presentes en la página se comunican con el servidor de normas para procesar el código basado en normas asociado con la zona. Cada zona de e-marketing tiene una o más iniciativas de campaña asociadas a ella. Para obtener más información sobre campañas e iniciativas de campaña, consulte el Capítulo 12, "Elementos de campaña" en la página 119 y la ayuda en línea de WebSphere Commerce.

Para que las iniciativas de campaña se visualicen correctamente en las páginas de la tienda, debe añadirse una zona de e-marketing al archivo JSP y, a continuación, registrarla en la base de datos utilizando WebSphere Commerce Accelerator. Este capítulo describe cómo añadir zonas de e-marketing a los archivos JSP de la tienda. Para obtener más información sobre cómo registrar la zona de e-marketing en la base de datos utilizando WebSphere Commerce Accelerator, consulte la ayuda en línea de WebSphere Commerce.

Zona de e-marketing

A continuación se muestra un ejemplo de una zona de e-marketing.

```
<!-- =====
/**-----
/** El ejemplo siguiente se proporciona "TAL CUAL".
/**
/** IBM lo ofrece como un simple ejemplo y no se ha probado exhaustivamente
/** bajo todas las condiciones. Por tanto, IBM no puede garantizar su
/** fiabilidad, operatividad o funcionalidad.
/**
/** Este ejemplo puede incluir los nombres de individuos, empresas, marcas
/** o productos para ilustrar conceptos de la forma más completa posible.
/** Todos estos nombres son ficticios y cualquier similitud con nombres
/** y direcciones utilizados por una persona o una empresa real es
/** puramente casual
/**-----
/**
=====-->
<%
/**
 * INICIO - el código siguiente debe existir sólo una vez por página; inicializa
 *           el contexto de mandatos y el bean de datos de tienda.
*/

// crear el bean de tienda para obtener el directorio de tienda
String collateralPath = "/webapp/wcs/stores/";
com.ibm.commerce.command.CommandContext emsCommandContext =
(com.ibm.commerce.command.CommandContext) request.getAttribute(
ECCConstants.EC_COMMANDCONTEXT);
com.ibm.commerce.common.beans.StoreDataBean storeDataBean =
new com.ibm.commerce.common.beans.StoreDataBean();
storeDataBean.setStoreId(emsCommandContext.getStoreId().toString());
com.ibm.commerce.beans.DataBeanManager.activate(storeDataBean, request);
if (storeDataBean.getDirectory() != null) {
    collateralPath += storeDataBean.getDirectory() + "/";
}
%>
<!-- =====
// El siguiente formulario HTML somete la petición de la zona de e-marketing al
// mandato ClickInfo que captura las estadísticas de campaña y redirige a la
// ubicación especificada por el parámetro URL.
```

```

===== -->
<form name="storeEmsForm" method="POST" action="/webapp/wcs/stores/servlet/ClickInfo">
  <input type="hidden" name="evtype">
  <input type="hidden" name="mpe_id">
  <input type="hidden" name="intv_id">
  <input type="hidden" name="URL">
</form>
<%
/**
 * FIN - el código siguiente debe existir sólo una vez por página; inicializa
 *       el contexto de mandatos y el bean de datos de tienda.
 */
%>
<%
/**
 * INICIO - el código siguiente puede utilizarse para desactivar múltiples zonas
 *          de e-marketing en la página. Antes de utilizarlo, personalice el
 *          nombre de instancia EMarketingSpot correspondiente y el nombre de la
 *          zona de e-marketing. Copie este código si necesita más de una zona,
 *          no utilice el mismo nombre de zona.
 */

// crear la zona de e-marketing
com.ibm.commerce.marketing.beans.EMarketingSpot eMarketingSpot =
new com.ibm.commerce.marketing.beans.EMarketingSpot();

// IMPORTANTE - establezca aquí el nombre correcto
eMarketingSpot.setName("nombreZonaMarketing");

// el número máximo de copias de productos/categorías/anuncios que se muestran
// a través de esta zona de e-marketing puede establecerse aquí
eMarketingSpot.setMaximumNumberOfCatalogEntries(20);
eMarketingSpot.setMaximumNumberOfCategories(20);
eMarketingSpot.setMaximumNumberOfCollateral(20);

// crear instancia de bean
com.ibm.commerce.beans.DataBeanManager.activate(eMarketingSpot,
request);
%>

<%
// El bloque siguiente se utiliza para visualizar los anuncios asociados a esta
// zona de e-marketing. Se puede hacer referencia al enlace de URL definido con un
// anuncio mediante el sometimiento del formulario HTML que se adjunta más arriba.

if (eMarketingSpot.getCollateral() != null && eMarketingSpot.getCollateral().
length > 0) {
%>
<TABLE>
<% for (int i=0; i eMarketingSpot.getCollateral().length; i++) { %>
<TR>
<% if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Image")) { %>
<TD>
  A HREF="javascript:document.storeEmsForm.evtype.value='CpgnClick'
;document.storeEmsForm.mpe_id.value=' <%= eMarketingSpot.getId() %>
';document.storeEmsForm.intv_id.value='<%= eMarketingSpot.getCollateral
()[i].getInitiativeId() %>';document.storeEmsForm.URL.value='
<%= eMarketingSpot.getCollateral()[i].getURLLink() %>';
document.storeEmsForm.submit();">
  IMG SRC=" <%= collateralPath + eMarketingSpot.getCollateral()[i].getLocation() %>">
  </A>
</TD>
<TD>
  <%= eMarketingSpot.getCollateral()[i].getMarketingText() %>
</TD>
<% } else if (eMarketingSpot.getCollateral()[i].getTypeName().equals("Flash")) { %>
<TD>
  EMBED src=" <%= collateralPath + eMarketingSpot.getCollateral()[i].getLocation()
%>" quality=high bgcolor=#FFFFFF WIDTH=120 HEIGHT=90
TYPE="application/x-shockwave-flash"> </EMBED>
</TD>
<TD>

```



```

        A HREF="javascript:document.storeEmsForm.evtype.value='CpgnClick'
;document.storeEmsForm.mpe_id.value='<% = eMarketingSpot.getId()
%>';document.storeEmsForm.intv_id.value='<% = eMarketingSpot.getCollateral()
[i].getInitiativeId() %>';document.storeEmsForm.URL.value='
<% = eMarketingSpot.getCollateral()[i].getUrlLink() %>';
document.storeEmsForm.submit();">
        <%= eMarketingSpot.getCollateral()[i].getMarketingText() %>
    </A>
</TD>
<% } %>
</TR>
<% } %>
</TABLE>
<% } %>

<%
// El bloque siguiente se utiliza para visualizar las categorías asociadas a esta
// zona de e-marketing. Se puede hacer referencia a la página de visualización de
// la categoría que muestra la categoría seleccionada en la campaña, mediante
// el sometimiento del formulario HTML que se adjunta más arriba.

if (eMarketingSpot.getCategories() != null && eMarketingSpot.getCategories().length > 0) {
%>
<TABLE>
<% for (int i=0; i eMarketingSpot.getCategories().length; i++) { %>
<TR>
<TD>
        A HREF="/webapp/wcs/stores/servlet/ClickInfo?evtype=CpgnClick
&mpe_id=<% = eMarketingSpot.getId() %>&intv_id=<% = eMarketingSpot.
getCategories()[i].getInitiativeId()
%>&URL=/webapp/wcs/stores/servlet/CategoryDisplay&
<%= ECConstants.EC_STORE_ID %>=<%= eMarketingSpot.getStoreId().toString() %>&
<% = ECConstants.EC_CATEGORY_ID %>=<% =
eMarketingSpot.getCategories()[i].getCategoryId() %>&<%
= ECConstants.EC_CATALOG_ID %>=<% = eMarketingSpot.getCategories()[i].getCatalogId() %>
&<% = ECConstants.EC_LANGUAGE_ID %>=<% = eMarketingSpot.getCategories()[i].getLanguageId().toString() %>">
        <%= eMarketingSpot.getCategories()[i].getDescription
(eMarketingSpot.getCategories()[i].getDescription
(emsCommandContext.getLanguageId()).getName() %>
    </A>
</TD>
<TD><% = eMarketingSpot.getCategories()[i].getDescription
(emsCommandContext.getLanguageId()).getLongDescription() %> </TD>
</TR>
<% } %>
</TABLE>
<% } %>

<%
// El bloque siguiente se utiliza para visualizar los productos asociados a esta
// zona de e-marketing. Se hará referencia a la página de visualización del
// producto que muestra el producto seleccionado en la campaña, mediante
// el sometimiento del formulario HTML que se adjunta más arriba.

if (eMarketingSpot.getCatalogEntries() != null &&
eMarketingSpot.getCatalogEntries().length > 0) {
%>
<TABLE>
<% for (int i=0; i eMarketingSpot.getCatalogEntries().length; i++) { %>
<TR>
<TD>
        A HREF="/webapp/wcs/stores/servlet/ClickInfo?evtype=CpgnClick
&mpe_id=<% = eMarketingSpot.getId() %>&intv_id=<% =
eMarketingSpot.getCatalogEntries()[i].getInitiativeId()
%>&URL=/webapp/wcs/stores/servlet/ProductDisplay
&<%= ECConstants.EC_STORE_ID %>=<%= eMarketingSpot.getStoreId().toString()
%>&<%= ECConstants.EC_PRODUCT_ID %>=<% =
eMarketingSpot.getCatalogEntries()[i].getCatalogEntryID() %>&<%
= ECConstants.EC_LANGUAGE_ID %>=<% = eMarketingSpot.getCatalogEntries()[i].getLanguageId().toString() %>">
        IMG SRC="<%= collateralPath + eMarketingSpot.getCatalogEntries()
[i].getDescription(emsCommandContext.getLanguageId())
.getThumbnail() %>" ALT="<%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription() %>" BORDER=0 WIDTH=60>

```

```

        </A>
    </TD>
    <TD><%= eMarketingSpot.getCatalogEntries()[i].getDescription
(emsCommandContext.getLanguageId()).getShortDescription() %> </TD>
</TR>
<% } %>
</TABLE>
<% } %>

<% /**
 * FIN - el código siguiente se utiliza para desactivar varias zonas de e-marketing
 * en la página. Antes de utilizarlo personalice el nombre de zona de
 * e-marketing. Copie este código si necesita más de una zona, no utilice el
 * mismo nombre de zona.
 */
%>

```

La zona de e-marketing anterior soporta tres tipos de iniciativas de campaña:

- Recomendación de productos
- Recomendación de categorías
- Anuncios publicitarios informativos

Nota: Para obtener información más detallada sobre cada una de estas iniciativas, consulte el Capítulo 12, “Elementos de campaña” en la página 119.

Bean e-MarketingSpot

Las zonas de e-marketing utilizan el bean e-MarketingSpot para devolver el resultado de las iniciativas de campaña que están planificadas actualmente en la zona. El uso de distintas propiedades del bean le permite personalizar la zona de e-marketing y la iniciativa de campaña correspondiente. Para obtener más información sobre el bean e-MarketingSpot y sus propiedades, consulte la ayuda en línea de WebSphere Commerce.

Adición de una zona de e-marketing a las páginas de la tienda

Para añadir una zona de e-marketing a las páginas de la tienda, haga lo siguiente:

1. Determine en qué archivos JSP se visualizará la zona. La zona se puede añadir a varios archivos JSP.
2. Determine en qué parte del archivo JSP colocará la zona.
3. Copie la zona de e-marketing de ejemplo del apartado “Zona de e-marketing” en la página 317.
4. Péguela en la ubicación que desee de su(s) archivo(s) JSP.
5. Personalice la zona de e-marketing de ejemplo para que se adapte al diseño de su(s) archivo(s) JSP.
6. En el código de la zona de e-marketing, ponga un nombre a la zona de e-marketing.

Nota: Los nombres de las zonas de e-marketing deben ser descriptivos, de forma que incluyan su ubicación, por ejemplo AnuncioPáginaPresentación o RecomendaciónPáginaPasarPorCaja. Esto ayuda a reducir la confusión acerca de dónde aparecerá y del contenido que deberá tener. Si es necesario, se pueden añadir números al nombre para establecer diferencias entre dos Zonas de e-marketing que aparecen en la misma página. Los nombres de zona de e-marketing deben ser

identificadores válidos de Java. Debe utilizar este mismo nombre al registrar la zona de e-marketing en la base de datos utilizando WebSphere Commerce Accelerator.

7. Si necesita más de una zona de e-marketing por archivo JSP, repita los pasos 2 a 6. Cuando añada otra zona de e-marketing a un archivo JSP, asegúrese de que copia toda la segunda sección de la zona de e-marketing de ejemplo que se proporciona en "Zona de e-marketing" en la página 317 y pegue la segunda sección entera de nuevo en el archivo JSP. A continuación, cambie el nombre de la zona de e-marketing en el archivo JSP.
8. Registre la zona de e-marketing en la base de datos utilizando WebSphere Commerce Accelerator. Para obtener instrucciones más detalladas, consulte la ayuda en línea de WebSphere Commerce.

Nota:

- a. Si piensa añadir el ID de tienda, el ID de catálogo o el ID de idioma al URL utilizando el siguiente convenio, "langId=<%= languageId %>", tenga en cuenta que el archivo JSP en el que se incluye la zona de e-marketing debe poner a disposición el ID adecuado. Los ID también pueden recuperarse a través del contexto de mandatos, por ejemplo, `getCommandContext().getLanguageId()`.
- b. Como resultado de la estructura de la tienda de ejemplo NewFashion, en las zonas de e-marketing sólo se pueden recomendar productos, no artículos en tiendas basadas en la tienda de ejemplo NewFashion.
- c. El parámetro de URL, `CatalogDisplay` debe empezar con "&" en lugar de "?" porque el código no hace referencia al mandato directamente.

Parte 9. Apéndices

Apéndice A. Descripción de UML

UML (Unified Modeling Language) es un lenguaje gráfico estándar para representar diferentes elementos de diseño de software. Los ejemplos siguientes son algunos de los elementos más comunes de UML. Para obtener más detalles sobre las especificaciones formales, consulte <http://www.rational.com> y <http://www.omg.org>.

Los diagramas UML constan de los elementos siguientes:

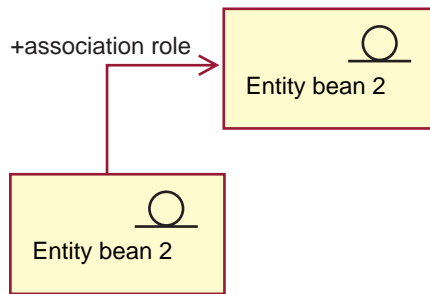
- **Recuadros:** Los recuadros representan clases de objetos. Los nombres de clase aparecen en la parte superior del recuadro. Los atributos, si se muestran, aparecen debajo del nombre de clase. El nombre de clase y los atributos están separados por una línea.
- **Líneas:** Las líneas representan las posibles relaciones entre objetos de dos clases. Los objetos de la clase de un extremo de la línea pueden "asociarse" con objetos de la otra clase.
- **Rombos sólidos:** Los rombos sólidos en el extremo de una línea indican contención por valor. Los objetos de la clase del otro extremo de la línea forman parte de un objeto, y sólo uno, de la clase a la que toca el rombo.
- **Rombos abiertos:** Los rombos abiertos en el extremo de una línea indican contención por referencia. Los objetos en el extremo de la línea donde hay el rombo pueden considerarse como objetos de agrupación de la clase del otro extremo de la línea.
- **Números de cardinalidad:** Aparecen en el extremo de las líneas de relación para indicar una restricción de cardinalidad. La tabla siguiente resume las restricciones de cardinales:

Número cardinal	Tipo de relación
1	uno y sólo uno
0..1	cero o uno
0..n	cero o más
1..n	uno o más

Si no se muestra una restricción de cardinales, se supone que la cardinalidad es 0..n, a menos que aparezca un rombo sólido en el extremo de una línea de relación. En tal caso, deberá ser 1.

- **Signos más:** Los signos más que aparecen en el extremo de las líneas de relación indican que el objeto de la clase del extremo de la línea desempeña un rol en la relación. El texto que va a continuación del signo más indica el rol del objeto en la relación.
- **Flechas:** Las flechas en el extremo de una línea de relación indican que la dirección de la relación entre dos objetos sigue la dirección de la flecha. La ausencia de flechas en una línea de relación indica que la dirección de la relación entre los objetos es normalmente en ambas direcciones.

Los diagramas siguientes ilustran los conceptos mencionados anteriormente.



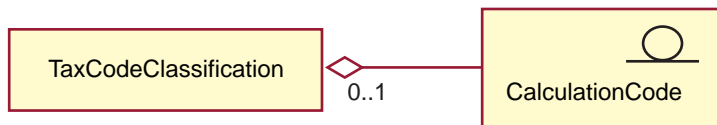
Este diagrama muestra dos beans de entidad con el símbolo estereotipo que indica un bean al que se denomina Enterprise Java Bean. Existe una asociación unidireccional del primer bean al segundo bean de entidad. El signo más va seguido de texto que describe qué rol desempeña el bean de Entidad 2 en la asociación.



En este diagrama, una Entidad de tienda tiene un solo y único propietario, que es un Miembro. Un Miembro puede tener cero o más Entidades de tienda. El signo más indica que el miembro desempeña un rol en la relación. En este caso, el Miembro es el propietario de la Entidad de tienda. La flecha indica que normalmente se puede encontrar el propietario de una entidad de tienda solicitando el propietario a la entidad de tienda y no solicitando a un miembro todas las entidades de tienda de las que es propietario.



En este diagrama, un Artículo de pedido siempre forma parte de un solo pedido. Un pedido tiene cero o más Artículos de pedido.



Este diagrama indica que un Código de cálculo se agrupa mediante cero o una Clasificación de códigos de impuestos y una Clasificación de códigos de impuestos agrupa cero o más Códigos de cálculo.

Apéndice B. Creación de los datos

Antes de crear los datos de la tienda con formato de archivo XML, efectúe lo siguiente:

- Determine el orden de la información que está creando. La información de cada uno de los capítulos de datos de la tienda le recomienda el orden en que debe crear los datos, pero cuando cree los archivos XML recuerde que la información de una tabla padre debe ser anterior a la información de una tabla hijo.
- Determine cómo desea utilizar la tienda. Si va a crear un archivador de tienda de ejemplo (.sar), que es un archivo que se copia y utiliza como base sobre la que se crean tiendas nuevas, necesitará crear datos ligeramente diferentes a los que necesitaría si fuera a crear una tienda de ejemplo. Para obtener más información, consulte “Creación de datos para tiendas de ejemplo”.

Creación de datos para tiendas de ejemplo

Los datos de los archivadores de tienda de ejemplo tienen el formato de archivos XML válidos para Loader package. Los archivos XML del archivador de tienda están destinados a ser portables y no deben contener claves primarias generadas que sean específicas de una instancia determinada de la base de datos. En lugar de ello, utilizan alias internos que ID Resolver resuelve en el momento de la publicación. La utilización de estos convenios permite copiar y publicar los archivadores de tienda de ejemplo varias veces.









No es necesario que siga estos convenios cuando cree datos de tienda para su tienda con formato de archivos XML, a no ser que piense crear un archivador de tienda que se utilizará para generar varias tiendas o que desee crear un archivador de tienda portable, es decir un archivador de tienda que pueda publicarse en otra instancia de WebSphere Commerce.

Por lo tanto, los archivadores de tienda de ejemplo utilizan los convenios siguientes:

- & como en `member_id="&MEMBER_ID;"`. El convenio &XXX; es una macro DTD (conocida en XML como entidad).

Nota: Debe definir MEMBER_ID como una macro DTD cuando cree un archivador de tienda de ejemplo.

WebSphere Commerce define un conjunto de macros en el archivo siguiente:

-  unidad:\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd
-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar\DBLoadMacros.dtd
-  /usr/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd
-  /opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd
-  /opt/WebSphere/CommerceServer/xml/sar/DBLoadMacros.dtd
-  /qibm/proddata/WebCommerce/xml/sar/DBLoadMacros.dtd

Las macros como en_US y es_ES se establecen en los ID de idioma apropiados. Por ejemplo:

```
<!ENTITY es_ES "-1">
```

La información se especificará utilizando las herramientas de Servicios de tienda. Por ejemplo, el usuario selecciona MEMBER_ID en la página Crear archivador de tienda de los Servicios de tienda. La macro MEMBER_ID es un espacio reservado para el ID del miembro que es propietario de la tienda. Cuando se crea un archivador de tienda, se selecciona un miembro para que sea el propietario de la tienda. La macro MEMBER_ID se establece en el ID del miembro. Por ejemplo, si selecciona el ID de miembro -2000, entonces MEMBER_ID se establece en -2000 como se indica a continuación:

```
<!ENTITY MEMBER_ID "-2000">
```

- @ como en `ffmcenter_id="@ffmcenter_id_1"`. El uso del símbolo @ se conoce como resolución de alias interno. ID Resolver es un programa de utilidad de Loader package que genera identificadores para elementos XML que los necesitan. Una de las técnicas que ID Resolver utiliza es la resolución de alias internos. Al utilizar la resolución de alias interno, la clave primaria (identificador) se sustituye por un alias en el documento XML. Este alias se utiliza luego en cualquier otra parte del archivo XML para hacer referencia a ese elemento. De este modo, no es necesario conocer los índices exclusivos necesarios para poder crear el archivo XML. Durante la publicación en los Servicios de tienda o utilizando Loader package, ID Resolver sustituye el símbolo @ por un valor exclusivo. Vea los ejemplos siguientes de un archivo XML:

- ID Resolver previo

```
<catalog
  catalog_id="@catalog_id_1"
  member_id="&MEMBER_ID;"
  identifier=InFashion"
  description="Catálogo de InFashion"/>
```

- ID Resolver posterior

```
<catalog
  catalog_id="10001"
  member_id="-2000"
  identifier=InFashion"
  description="Catálogo de InFashion"/>
```

donde 10001 es el ID exclusivo asignado por ID Resolver y -2000 es el ID de miembro seleccionado por el usuario en los Servicios de tienda. El archivo XML resultante se carga utilizando Loader package. Si se ejecutan los archivos mediante ID Resolver, se asegura que se puedan crear muchas tiendas a partir de un conjunto individual de archivos XML.

Servicios de tienda y tiendas de ejemplo

Las opciones **Nuevo** y **Crear archivador de tienda** de los Servicios de tienda dependen de los convenios descritos anteriormente. Si desea utilizar la tienda como una plantilla para crear otras tiendas mediante los Servicios de tienda, deberá seguir estos convenios para crear elementos de datos.

Sin embargo, si no va a crear un archivador de tienda de ejemplo, puede editar y publicar un archivador de tienda que no siga estos convenios utilizando las herramientas de los Servicios de tienda.

Apéndice C. sarinfo.xml

Todo archivador de tienda debe contener un archivo `sarinfo.xml`. Este archivo, que se conoce como descriptor, contiene información acerca del archivador de tienda que se utiliza cuando se publica un archivador de tienda, incluidos los nombres de los ZIP de los elementos de archivos y los archivos XML de la base de datos de la tienda, en el orden en que se publican. Si un archivador de tienda incluye archivos en varios idiomas, el archivo `sarinfo.xml` también incluye dicha información y determina el orden en el que se publica cada archivo de idioma.

Nota: El orden en que se publican los elementos de datos es importante ya que algunos elementos de datos se deben publicar antes que otros. Como resultado, el orden de los elementos, tal y como se especifica en el archivo `sarinfo.xml`, deberá coincidir con el orden de los elementos especificados en los archivos `sarinfo.xml` de las tiendas de ejemplo. Los archivos de archivador de tienda de ejemplo se encuentran en el directorio siguiente:

- ▶ NT unidad:\WebSphere\CommerceServer\samplstores
- ▶ 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
- ▶ AIX /usr/WebSphere/CommerceServer/samplstores
- ▶ Solaris /opt/WebSphere/CommerceServer/samplstores
- ▶ Linux /opt/WebSphere/CommerceServer/samplstores
- ▶ 400 /qibm/proddata/WebCommerce/samplstores

Para ver el contenido del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo `sarinfo.xml` está ubicado en el directorio SAR-INF.

Ejemplo de sarinfo.xml

El ejemplo siguiente es el archivo `sarinfo.xml` de ToolTech. Para obtener más información sobre los elementos, atributos y valores de atributos, consulte la información siguiente. Para obtener más información sobre las especificaciones XML de un archivador de tienda, consulte `sarinfo.dtd` en el directorio siguiente:

- ▶ NT unidad:\WebSphere\CommerceServer\xml\sar
- ▶ 2000 unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
- ▶ AIX /usr/WebSphere/CommerceServer/xml/sar
- ▶ Solaris /opt/WebSphere/CommerceServer/xml/sar
- ▶ Linux /opt/WebSphere/CommerceServer/xml/sar
- ▶ 400 /qibm/proddata/WebCommerce/xml/sar

```
<?xml version = "1.0"?>
<!DOCTYPE sarinfo SYSTEM "sarinfo.dtd">
<sarinfo complete-store="yes" multi-language="yes" version="1.0">

<store-info asset-name="store"/>

<file name="webapp.zip" type="zip">
```

```

<asset fragmented="no" name="webapp">
<file name="webapp.zip" type="zip">
<display-name>Nombre de visualización de mi aplicación web</display-name>
<description>Mi aplicación web</description>
</file>
</asset>

<asset fragmented="no" name="properties">
<file name="properties.zip" type="zip" />
</asset>

<asset fragmented="no" name="dbloadmacros">
<file name="data/DBLoadMacros.dtd" type="dtd"/>
</asset>

<asset fragmented="no" name="fulfillment">
<file name="data/fulfillment.dtd" type="dtd"/>
<file name="data/fulfillment.xml" priority="1" type="db-load"/>
<file name="data/en_US/fulfillment.xml" priority="31" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/fulfillment.xml" priority="31" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="store">
<file name="data/store.dtd" type="dtd"/>
<file name="data/store.xml" priority="2" type="db-load"/>
<file name="data/en_US/store.xml" priority="3" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/store.xml" priority="3" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="catalog">
<file name="data/catalog.dtd" type="dtd"/>
<file name="data/catalog.xml" priority="4" type="db-load"/>
<file name="data/en_US/catalog.xml" priority="5" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/catalog.xml" priority="5" type="db-load">
<locale>es_ES</locale>
</asset>

<asset fragmented="yes" name="tax">
<file name="data/tax.dtd" type="dtd"/>
<file name="data/tax.xml" priority="6" type="db-load"/>
<file name="data/en_US/tax.xml" priority="7" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/tax.xml" priority="7" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="taxfulfill">
<file name="data/taxfulfill.dtd" type="dtd"/>
<file name="data/taxfulfill.xml" priority="8" type="db-load"/>
</asset>

<asset fragmented="yes" name="shipping">
<file name="data/shipping.dtd" type="dtd"/>

```

```

<file name="data/shipping.xml" priority="9" type="db-load"/>
file name="data/en_US/shipping.xml" priority="10" type="db-load">
<locale>en_US</locale>
</file>
file name="data/es_ES/shipping.xml" priority="10" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="shippingfulfill">
<file name="data/shipfulfill.dtd" type="dtd"/>
<file name="data/shipfulfill.xml" priority="11" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog">
<file name="data/store-catalog.dtd" type="dtd"/>
<file name="data/store-catalog.xml" priority="12" type="db-load"/>
</asset>

<asset fragmented="no" name="storefulfill">
<file name="data/storefulfill.dtd" type="dtd"/>
<file name="data/storefulfill.xml" priority="13" type="db-load"/>
</asset>

<asset fragmented="yes" name="offering">
<file name="data/offering.dtd" type="dtd"/>
<file name="data/offering.xml" priority="14" type="db-load"/>
</asset>

<asset fragmented="no" name="command">
<file name="data/command.dtd" type="dtd"/>
<file name="data/command.xml" priority="16" type="db-load"/>
</asset>

<asset fragmented="yes" name="currency">
<file name="data/currency.dtd" type="dtd"/>
<file name="data/currency.xml" priority="17" type="db-load"/>
<file name="data/en_US/currency.xml" priority="18" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/currency.xml" priority="18" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="campaign">
<file name="data/campaign.dtd" type="dtd"/>
<file name="data/campaign.xml" priority="20" type="db-load"/>

<file name="data/en_US/campaign.xml" priority="24" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/campaign.xml" priority="24" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="store-catalog-tax">
<file name="data/store-catalog-tax.dtd" type="dtd"/>
<file name="data/store-catalog-tax.xml" priority="21" type="db-load"/>
</asset>

<asset fragmented="no" name="store-catalog-shipping">
<file name="data/store-catalog-shipping.dtd" type="dtd"/>
<file name="data/store-catalog-shipping.xml" priority="22" type="db-load"/>
</asset>

```

```

<asset fragmented="no" name="store-defaults">
<file name="data/store-defaults.dtd" type="dtd"/>
<file name="data/store-defaults.xml" priority="23" type="db-load"/>
</asset>

<asset fragmented="no" name="consistency_check">
<file name="data/sarrule.dtd" type="dtd"/>
<file name="data/sarrule.xml" priority="25" type="config"/>
</asset>

<asset fragmented="no" name="payment">
<file name="data/es_ES/paymentinfo.xml" type="config"/>
<file name="data/paymentinfo.dtd" type="dtd"/>
</asset>

<asset fragmented="yes" name="policy">
<file name="data/businesspolicy.dtd" type="dtd"/>
<file name="data/businesspolicy.xml" priority="26" type="db-load"/>
<file name="data/en_US/businesspolicy.xml" priority="27" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/businesspolicy.xml" priority="27" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="no" name="organization">
<file name="data/organization.dtd" type="dtd"/>
<file name="data/organization.xml" priority="28" type="db-load"/>
</asset>

asset fragmented="no" name="businessaccount">
<file name="data/businessaccount.xml" type="xml"/>
/asset>
asset fragmented="yes" name="contract">
<file name="data/contract.xml" priority="1" type="xml"/>
<file name="data/en_US/contract.xml" priority="2" type="xml">
<locale>en_US</locale>
</file>
file name="data/es_ES/contract.xml" priority="2" type="xml">
<locale>es_ES</locale>
</file>
</asset>

<asset fragmented="yes" name="accesscontrol">
<file name="data/accesscontrol.dtd" type="dtd"/>
<file name="data/accesscontrol.xml" priority="29" type="db-load"/>
<file name="data/en_US/accesscontrol.xml" priority="30" type="db-load">
<locale>en_US</locale>
</file>
<file name="data/es_ES/accesscontrol.xml" priority="30" type="db-load">
<locale>es_ES</locale>
</file>
</asset>

<!-- next priority should be 32 -->

</sarinfo>

```

donde

sarinfo es toda la información que contiene el archivo sarinfo.xml. Los atributos del diagrama siguiente contienen información general acerca del archivador de tienda.

Nombre del atributo	Valor(es) del atributo
multi-language (necesario)	Determina si se dará soporte a varios idiomas en este archivador de tienda: <ul style="list-style-type: none"> • yes • no
complete-store (necesario)	Determina si el archivador de tienda incluye los elementos necesarios para una tienda completa: <ul style="list-style-type: none"> • yes • no
version	La versión del archivador de tienda. Por ejemplo: 1.0, 1.1
display-name	El nombre del archivador de tienda.
description	Una descripción breve del archivador de tienda.
standard-schema (necesario)	Si la tienda sigue el esquema de base de datos de WebSphere Commerce estándar: <ul style="list-style-type: none"> • yes • no
store-info asset-name (necesario)	El elemento que funciona como contenedor del archivador de tienda. Toda la información de la tienda se encuentra en los archivos pertenecientes a este elemento. Por ejemplo: store
locale	El entorno nacional o los entornos nacionales a los que da soporte el archivador de tienda. Las variables del entorno nacional (que se listan a continuación) constan del idioma_país: <ul style="list-style-type: none"> • de_DE • en_US • es_ES • fr_FR • it_IT • ja_JP • ko_KR • pt_BR • zh_CN • Zh_TW

asset (obligatorio) Un elemento es una agrupación lógica de archivos relacionados. Por ejemplo, tax (impuesto) puede ser el nombre de un grupo de archivos en el que todos los archivos están relacionados con los impuestos de la tienda.

Nombre del atributo	Valor(es) del atributo
name (necesario)	El nombre del tipo de elemento. Por ejemplo: <ul style="list-style-type: none"> • store • catalog • payment • tax
fragmented (necesario)	Determina si la información del elemento se divide en varios archivos según los idiomas. <ul style="list-style-type: none"> • yes • no

file

Nombre del atributo	Valor(es) del atributo
name (necesario)	El nombre del archivo.
type (necesario)	El tipo de formato del archivo. Por ejemplo: <ul style="list-style-type: none"> • db-load - archivos que se deben cargar en la base de datos • dtd - archivos de definición de tipo de documento • zip - archivo ZIP para elementos de archivos, por ejemplo webapp.zip • config - archivos de configuración
priority	El orden en que se publicarán los archivos en el archivador de tienda. 1,2,3,4 . . . Nota: Si dos archivos comparten la misma prioridad, no importa el orden en que se cargan. Si los archivos se tienen que cargar en un orden determinado, compruebe que tienen asignadas prioridades diferentes.
display-name	El nombre del archivo.
description	Descripción para referencia.
locale	El entorno nacional. Las variables del entorno nacional (que se listan a continuación) constan del idioma_país. <ul style="list-style-type: none"> • de_DE • en_US • es_ES • fr_FR • it_IT • ja_JP • ko_KR • pt_BR • zh_CN • Zh_TW







Apéndice D. sarrule.xml

Todo archivador de tienda contiene un archivo `sarrule.xml` que comprueba la coherencia cuando se publica utilizando los Servicios de tienda. Al publicar, el programa de utilidad de publicación utiliza las normas del archivo `sarrule.xml` para comprobar que el archivador de tienda contiene los elementos web que se listan en los archivos XML.

Ejemplo de `sarrule.xml`







El archivo `sarrule.xml` siguiente es de la tienda de ejemplo ToolTech.

Los archivadores de tienda de ejemplo están en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\samplstores
-  unidad:\Archivos de programa\WebSphere\CommerceServer\samplstores
-  /usr/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /opt/WebSphere/CommerceServer/samplstores
-  /qibm/proddata/WebCommerce/samplstores

Para ver el archivo `sarrule.xml` del archivador de tienda, descomprímalo utilizando un programa ZIP. El archivo `sarrule.xml` está ubicado en el directorio de datos.

El archivo `sarrule.dtd` está en el directorio siguiente:

-  unidad:\WebSphere\CommerceServer\xml\sar
-  unidad:\Archivos de programa\WebSphere\CommerceServer\xml\sar
-  /usr/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /opt/WebSphere/CommerceServer/xml/sar
-  /qibm/proddata/WebCommerce/xml/sar

Nota: Se recomienda utilizar el archivo `sarrule.xml` que se proporciona con las tiendas de ejemplo en su archivador de tienda. Sin embargo, si lo desea puede añadir normas nuevas al archivo `sarrule.xml` existente.

```
<?xml version="1.0"?>
<!DOCTYPE SAR-rules SYSTEM "sarrule.dtd">
<SAR-rules>
  <asset name = "command">
    <check type="webasset registration">
      <rule entry="viewreg" attribute="properties" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
  </asset>
!--
  <asset name = "catalog">
    <check type="webasset registration">
```

```

        <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
-->
    <asset name = "store">
    <check type="webasset registration">
        <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
        <rule entry = "dispcgprel" attribute="pagename" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
    </asset>
</SAR-rules>

```

donde, en la parte

```

<asset name = "command">
    <check type="webasset registration">
        <rule entry="viewreg" attribute="properties" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
    </asset>

```

el programa de utilidad de publicación de los Servicios de tienda comprueba que cada archivo JSP listado en el archivo command.xml exista en los elementos web del archivador de tienda.

en la parte

```

    <asset name = "catalog">
    <check type="webasset registration">
        <rule entry = "catalogdsc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catalogdsc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catentdesc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catentdesc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catgrpdesc" attribute="thumbnail" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "catgrpdesc" attribute="fullimage" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
    </check>
    </asset>

```

el programa de utilidad de publicación de los Servicios de tienda comprueba que cada elemento de catálogo listado en el archivo catalog.xml exista en los elementos web del archivador de tienda.

y en la parte

```

<asset name = "store">
    <check type="webasset registration">
        <rule entry = "dispentrel" attribute="pagename" type="java.lang.String"

```

```
        removeStoreDir="false" file="webapp.zip"/>
    <rule entry = "dispcgpre1" attribute="pagename" type="java.lang.String"
        removeStoreDir="false" file="webapp.zip"/>
</check>
</asset>
```

el programa de utilidad de publicación de los Servicios de tienda comprueba que cada elemento de tienda listado en el archivo `store.xml` exista en los elementos web del archivador de tienda.



Si hay problemas de memoria durante la publicación, suprima el comentario de la norma del elemento de catálogo de `sarrule.xml` antes de intentar realizar la publicación.

Apéndice E. Grupos de elementos de base de datos

Todos los elementos de base de datos de WebSphere Commerce se dividen en grupos para crearse y cargarse. Estos grupos son un conjunto de tablas relacionadas de forma lógica. El orden en el que se organizan estos grupos de elementos de base de datos es importante para la carga de datos, puesto que los objetos deben existir antes de cargar la relación entre los objetos.

Al cargar elementos de base de datos en formato XML para la tienda, puede elegir cargar sólo grupos seleccionados. Estos grupos constan de los elementos de base de datos creados en los capítulos anteriores, por ejemplo los elementos de catálogo o de despacho de pedidos. Antes de cargar grupos de datos como se indica en el apartado “Carga de grupos de elementos de base de datos” en la página 287, realice lo siguiente:

- Determine qué grupo de elementos de base de datos está cargando. Cada grupo contiene dependencias que deben satisfacerse para que los elementos se puedan cargar. Revise la información del apartado “Dependencias de grupos de elementos de base de datos”.
- Asegúrese de haber creado o actualizado los archivos XML para el grupo de elementos de base de datos seleccionado. La información de cada uno de los capítulos de elementos de la tienda le recomienda el orden en el que debe crear los elementos de base de datos, pero cuando cree o actualice archivos XML, recuerde que la información de una tabla padre debe preceder a la información de una tabla hijo.

Dependencias de grupos de elementos de base de datos

Cada grupo de elementos de base de datos obtiene su información de las tablas de base de datos de WebSphere Commerce. Los elementos de base de datos tienen dependencias dentro de su propio grupo. Es decir, un grupo de elementos de base de datos no puede obtener datos en otros archivos XML de un grupo de datos distinto, y cada grupo es independiente menos las claves externas. Sin embargo, si el grupo de elementos de base de datos tiene que hacer referencia a los datos externos definidos en otro grupo, deberá proporcionar estos datos manualmente. Esto significa que los datos de un grupo tienen una *dependencia externa* de los datos definidos fuera de su dominio, es decir, en otro grupo de elementos de base de datos. Las dependencias externas se producen cuando un grupo de elementos de base de datos tiene una relación de clave externa con la clave primaria de una tabla en otro grupo. Para cargar un grupo de elementos de base de datos, deben satisfacerse sus dependencias externas. Para utilizar un ejemplo del diagrama siguiente, una de las dependencias externas del grupo de elementos de base de datos de la tienda es fulfillment.FFMCENTER.FFMCENTER_ID; esta dependencia indica que, para poder cargar el grupo de elementos de base de datos de la tienda, los elementos de base de datos de despacho de pedidos deben ya existir en la base de datos de WebSphere Commerce.

Consulte el diagrama siguiente antes de iniciar el proceso de carga. Cada grupo de elementos de base de datos es dependiente de otras tablas de base de datos, de las que se cargan los datos.

Algunos puntos que se deben recordar:

- Algunas dependencias externas quizá no se satisfagan con un solo grupo. Los elementos generales de base de datos o los elementos a nivel de sitio, utilizados por todas las tiendas, se rellenan automáticamente al crear la instancia en la *rutina de carga* y se pueden acceder de inmediato. Las tablas contenidas en grupos de elementos de base de datos tienen referencias de clave externa a este tipo de datos. Los datos de la rutina de carga están divididos en datos comunes y datos específicos del entorno nacional. Si tiene una tienda multilingüe, debe elegir los datos de rutina de carga comunes y los específicos del entorno nacional. Por ejemplo, necesita los datos de rutina de carga de idioma y de miembros. El proceso de creación de la instancia rellena la tabla LANGUAGE con los idiomas de WebSphere Commerce soportados para su tienda y crea una organización raíz (MEMBER.MEMBER_ID=-2001) y una organización por omisión (MEMBER.MEMBER_ID=-2000). Debe utilizar la organización raíz cuando sea necesario, pero debe crear una organización propietaria de la tienda en lugar de utilizar la organización por omisión. Para obtener más información sobre las organizaciones y su jerarquía, consulte la ayuda en línea de WebSphere Commerce.
- Los archivos que se listan bajo la columna **Dependencias externas** utilizan la siguiente estructura de denominación: *grupo elementos basedatos.tabla basedatos.columna basedatos*. Utilizando el archivo store.STOREENT.STOREENT_ID como ejemplo, los datos se obtienen del grupo de elementos de base de datos store, tabla STOREENT y columna STOREENT_ID. Los nombres de archivo que empiezan por *bootstrap* indican que los datos se han llenado durante la creación de la instancia de WebSphere Commerce.
- Los archivos listados bajo la columna **Dependencias externas** contienen referencias de claves externas a las **Tablas relacionadas**. Primero deben llenarse de datos estas tablas.
- Las tablas se han dividido, sólo a efectos de presentación, para indicar las tablas específicas de entorno nacional que contienen información multilingüe como, por ejemplo, las descripciones de productos.
- Las tablas del diagrama representan los elementos de base de datos de las tiendas de ejemplo de WebSphere Commerce. Las tablas pueden variar de acuerdo con el tamaño, la función y las necesidades de la tienda. Dependiendo de los requisitos de su tienda, asegúrese de incluir todas las tablas de base de datos que contienen los elementos de la tienda, incluso si esa tabla en concreto no está listada a continuación.

Elementos de base de datos de control de acceso		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID (organizaciones raíz y propietaria de la tienda), bootstrap.MEMBER.MEMBER_ID (organizaciones raíz y propietaria de la tienda)	accesscontrol.xml ACACTACTGP, ACACTGRP, ACACTION, ACPOLICY, ACRESCGRY, ACRESGPRES, ACRESGRP	accesscontrol.xml ACACGPDESC, ACACTDESC, ACPOLDESC, ACRSCGDES, ACRESGPDES
Elementos de base de datos de política de negocio		

Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID, store.STOREENT.STOREENT_ID (organización propietaria de la tienda)	businesspolicy.xml POLICY, POLICYCMD	businesspolicy.xml POLICYDESC
Elementos de base de datos de campaña		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
store.STOREENT.STOREENT_ID	campaign.xml CAMPAIGN, COLLATERAL, EMSPOT, STENCALUSG	campaign.xml COLLDESC
Elementos de base de datos de catálogo		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos

bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organización propietaria de la tienda), store.STOREENT.STOREENT_ID, shipping.CALCODE.CALCODE_ID, tax.CALCODE.CALCODE_ID	catalog.xml BASEITEM, CATALOG, CATENTREL, CATENTRY, CATGROUP, CATGRPREL, CATTOGRP, ITEMSPC, ITEMVERSN, RA, RADETAIL, STOREITEM, STORITMFFC, VERSIONSPC offering.xml CATGRPTPC, MGPTRDPSCN, OFFER, OFFERPRICE, TRADEPOSCN storefulfill.xml INVENTORY store-catalog.xml DISPCGPREL, DISPENTREL, STORECAT, STORECENT, STORECGRP store-catalog- shipping.xml CATENTCALCD, CATENTSHIP store-catalog- tax.xml CATENTCALD	catalog.xml ATTRIBUTE, ATTRVALUE, BASEITMDSC, CATALOGDSC, CATENTDESC, CATGRPDESC, PKGATTR, PKGATTRVAL,
Elementos de base de datos de mandato		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
store.STOREENT.STOREENT_ID	command.xml CMDREG, VIEWREG	N/D
Business Elementos de base de datos de contrato		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	
store.STOREENT.STOREENT_ID	Las tablas de base de datos de contrato no se cargan directamente y siguen un proceso diferente de los demás grupos de datos de WebSphere Commerce. Consulte el apartado "Publicación de elementos de contrato" en la página 295 para obtener más información.	

Elementos de base de datos de moneda		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
store.STOREENT.STOREENT_ID	currency.xml CURCVLIST	currency.xml CURCONVERT, CURLIST
Elementos de base de datos de despacho de pedidos		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organización propietaria de la tienda)	fulfillment.xml FFMCENTER, STADDRESS	fulfillment.xml FFMCENTDS
Elementos de base de datos de organización		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID (organizaciones raíz y propietaria de la tienda), bootstrap.MEMBER.MEMBER_ID (organizaciones raíz y propietaria de la tienda)	organization.xml ADDRBOOK, ADDRESS, MBRREL, MEMBER, ORGENCY	N/D
Elementos de base de datos de envío		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID (organización propietaria de la tienda), fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	shipping.xml CALCODE, CALRULE, CRULESCALE, JURST, JURSTGPREL, JURSTGROUP, SHIPMODE, STENCALUSG shipping.xml SHPJCRULE, SHPARRANGE	shipping.xml CALCODEDSC, CALRANGE, CALRLOOKUP, CALSCALE, SHPMODEDSC
Elementos de base de datos de tienda		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos

bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(organización propietaria de la tienda), bootstrap.SETCURR.SETCURR_ID, fulfillment.FFMCENTER.FFMCENTER_ID	store.xml INVADJCODE, RTNREASON, STORE, STOREENT, STORELANG, VENDOR	store.xml FFMCENTDS, INVADJDESC, RTNRSNDESC, STADDRESS, STOREENTDS, STORLANGDS, VENDORDESC
Elementos de base de datos por omisión de tienda		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
shipping.SHIPMODE.SHIPMODE_ID (si es aplicable, primero debe cargarse este archivo), contract.CONTRACT.CONTRACT_ID, store.STOREENT.STOREENT_ID	store-default.xml STOREDEF	N/D
Elementos de base de datos de impuestos		
Dependencias externas	Tablas relacionadas de los archivos XML de elementos de base de datos	Tablas relacionadas específicas de entorno nacional de los archivos XML de elementos de base de datos
bootstrap.LANGUAGE.LANGUAGE_ID, bootstrap.MEMBER.MEMBER_ID(organización propietaria de la tienda), store.STOREENT.STOREENT_ID, fulfillment.FFMCENTER.FFMCENTER_ID, store.STOREENT.STOREENT_ID	tax.xml CALCODE, CALRANGE, CALRLOOKUP, CALRULE, CALSCALE, CRULESCALE, JURST, JURSTGROUP, JURSTGPREL, STENCALUSG, TAXCGRY, TAXJCRULE taxfulfill.xml TAXJCRULE	tax.xml CALCODEDSC, CALSCALEDS, TAXCGRYDS

Apéndice F. Avisos

Esta información se ha desarrollado para productos y servicios ofrecidos en EE.UU. Es posible que IBM no proporcione los productos, servicios o características a los que hace referencia este documento en otros países. Póngase en contacto con su representante de IBM local para obtener información acerca de los productos y servicios disponibles actualmente en su área. 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 solicitudes de patente pendientes 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
500 Columbus Avenue
Thornwood, NY 10594
EE.UU.

Para realizar consultas sobre la licencia en relación a 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, Japón

El párrafo siguiente no es aplicable al Reino Unido ni a cualquier otro país en el que tales disposiciones contradigan la normativa 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.

Todas las referencias hechas en este documento a sitios Web que no son de IBM se proporcionan únicamente para su información y no representan en modo alguno

una recomendación de dichos sitios Web. El contenido de esos sitios Web no forma parte del contenido de este producto de IBM, por lo que la utilización de dichos sitios es responsabilidad del usuario.

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 IBM, o de cualquier acuerdo equivalente entre IBM y el cliente.

Todos los datos de rendimiento incluidos en este documento han sido determinados en un entorno controlado. Por consiguiente, los resultados obtenidos en otros entornos operativos pueden variar de forma significativa. Algunas mediciones pueden haberse realizado en sistemas de nivel de desarrollo y no hay ninguna garantía de que estas mediciones sean las mismas en sistemas de uso general. Asimismo, algunas mediciones se pueden haber estimado mediante extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deben verificar qué datos son aplicables a su entorno específico.

La información sobre productos que no son de IBM se ha obtenido de los distribuidores de dichos productos, de los anuncios publicados o de otras fuentes disponibles públicamente. IBM no ha probado esos productos y no puede confirmar la precisión del rendimiento, la compatibilidad ni ninguna otra afirmación relacionada con productos que no son de IBM. Las preguntas sobre las prestaciones de productos no de IBM deben dirigirse a los distribuidores de dichos productos.

Todas las declaraciones sobre futuras directrices o intenciones de IBM están sujetas a modificación o retirada sin previo aviso y representan únicamente metas y objetivos.

Todos los precios IBM mostrados son precios al por menor sugeridos por IBM, son actuales y están sujetos a cambios sin previo aviso. Los precios del comerciante pueden variar.

Esta información se proporciona únicamente con fines de planificación. Está sujeta a posibles cambios antes de que los productos que en ella se describen estén disponibles.

Esta información contiene ejemplos de datos e informes que se utilizan en operaciones comerciales cotidianas. Para ilustrar los ejemplos de la forma más completa posible, éstos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier similitud con nombres y direcciones utilizados por una empresa real es puramente casual.

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 que se ajusten 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, operatividad 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 que se ajusten a las interfaces de programas de aplicación de IBM.

Cada copia, parcial o completa, de estos programas de ejemplo, o cualquier trabajo obtenido a partir de los mismos, debe incluir el siguiente aviso de copyright:

©Copyright International Business Machines Corporation 2001. Parte de este código se ha obtenido a partir de Programas de ejemplo de IBM Corp. ©Copyright IBM Corp. 2000, 2001. Reservados todos los derechos.

Si examina esta información en formato de copia software, es posible que algunas fotografías o ilustraciones a color no aparezcan.

Las imágenes de tarjetas de crédito, las marcas registradas y los nombres de comercios de este producto sólo deben ser utilizados por comerciantes autorizados por el propietario de la marca de tarjeta de crédito para aceptar el pago a través de dicha tarjeta de crédito.

Marcas registradas

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

AIX	IBM	WebSphere
AS/400	IBM Payment Manager	
DB2	iSeries	
DB2 Universal Database	OS/400	
eServer	VisualAge	

Microsoft, Windows y Windows NT, Active Directory y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos y/o en otros países.

Oracle es una marca registrada y Oracle8 es una marca comercial de Oracle Corporation.

SET Secure Electronic Transaction, SET y el logotipo de SET son marcas registradas propiedad de SET Secure Electronic Transaction LLC. La utilización de las marcas registradas sin una licencia escrita de SET Secure Electronic Transaction LLC está estrictamente prohibida.

Solaris, Solaris Operating Environment, Java, JavaServer Pages, JavaBeans y todas las marcas registradas y los logotipos basados en Java son marcas registradas de Sun Microsystems, Inc.

UNIX es una marca registrada de The Open Group en los Estados Unidos y en otros países.

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

IBM