



Integration Guide for WebSphere Commerce with SAP R/3 using the IBM WebSphere InterChange Server

Version 5.6



Integration Guide for WebSphere Commerce with SAP R/3 using the IBM WebSphere InterChange Server

Version 5.6

Note

Before using this information and the product that it supports, read the information in “Notices” on page 51.

First Edition (February 2005)

This edition applies to IBM WebSphere Commerce Version 5.6 and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure that you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office that serves your locality. Publications are not stocked at the address given below.

IBM welcomes your comments. You can send your comments by using the online IBM WebSphere Commerce documentation feedback form, available at the following URL:

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

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2003, 2005. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Before you begin v

Conventions used in this guide vi

Default paths vi

Chapter 1. Introduction 1

Terminology 1

Overview 1

Business scenario-integration with the seller's
backend system 2

Extensibility 2

Business models enabled 2

Benefits 3

Functionality 3

References 4

Chapter 2. Prerequisites 5

Software prerequisites 5

Hardware prerequisites 6

Chapter 3. Integrating WebSphere Commerce business processes with SAP R/3. 7

WebSphere Commerce outbound process 7

WebSphere Commerce inbound process 8

WebSphere Commerce request/reply process 9

Sending a request. 9

Receiving a response 10

Chapter 4. Installing and configuring 11

Sample topology. 11

Installing and configuring WebSphere Commerce. . 11

Installing and configuring WebSphere MQ 12

Configuring the WebSphere Commerce messaging
system 13

Publishing the WICSSAPAdvancedB2BDirect store 13

Before publishing the store 13

Populating the units of measure 13

Publishing the store 14

Creating a new store 14

Disabling SAP specific store customization 15

Configuring the SAP R/3 system 15

Synchronizing product data between WebSphere
Commerce and SAP 15

Loading sample store data onto SAP 16

Loading material data onto WebSphere
Commerce. 18

Installing and configuring the WebSphere

InterChange server 21

Importing the InterChange server components. . 21

Setting up the System Manager. 21

Setting up the WebSphere InterChange server
environment 21

Generating SAP application specific business
objects 23

Loading and deploying the assets 24

Configuring MO_WCSConfig 28

Associating the collaboration objects 28

Port connector 28

Configuring the WebSphere Business Integration
Adapter for WebSphere Commerce 28

Configuring the WebSphere Business Integration
Adapter for mySAP.com 29

SAP modules. 29

Configuring the WebSphere InterChange server
queue manager 30

Enabling asynchronous communication 30

Configuring the WebSphere Commerce MQ 30

Enabling the message flow 30

Chapter 5. Verification procedure . . . 33

Verifying the messages 33

Customer create or update message (DEBMAS05) 33

CheckInventoryAvailabilityBE request or response
message 33

Order create message (ORDERS05) 34

Order confirmation status message (SISCSO01) 34

Order delivery status message (SISDEL01) . . . 34

Product inventory update message (INVCON01) 35

Order invoice status message (SISINV01) . . . 35

Product price update message (COND_A02) . . . 35

Appendix A. Messages integrated for this reference application. 37

Asynchronous messages 37

Synchronous messages 37

Appendix B. Generic business objects 39

Appendix C. Application specific business objects 41

Appendix D. Collaboration objects and ports 43

Appendix E. Maps with specific values for WebSphere Commerce and SAP R/3 45

Appendix F. Binding maps to business objects. 47

WebSphere Commerce connector 47

WebSphere Business Integration Adapter for
mySAP.com 47

Appendix G. Store customization . . . 49

Notices 51

Trademarks	52
----------------------	----

Before you begin

The *Integration Guide for WebSphere® Commerce with SAP R/3 using the IBM® WebSphere InterChange Server* is intended for those who want to integrate WebSphere Commerce to a backend system using the IBM WebSphere InterChange Server. This guide describes how IBM WebSphere Commerce Version 5.6 can be integrated in particular with the SAP R/3 4.6 Enterprise Resource Planning (ERP) system using the WebSphere InterChange Server 4.2.2. The approach described for integrating with SAP R/3 can be used for other backend systems that the WebSphere InterChange Server supports. This guide will assist developers and engagement teams developing an integrated solution for any backend system like SAP R/3. Additionally, demonstrators or marketing personnel who want to demonstrate the integration functionality of WebSphere Commerce and the WebSphere InterChange Server can use this book.

Note: Knowledge of WebSphere Commerce Business Edition, SAP R/3 Enterprise system, WebSphere InterChange Server, and WebSphere Business Integration (WBI) Adapters is assumed.

This document provides information on how asynchronous and synchronous messages can be exchanged between WebSphere Commerce and SAP R/3 through the WebSphere InterChange Server. It gives an overview of a typical end-to-end flow between the two systems, installation and configuration of the various components, and pointers to the WebSphere InterChange Server entities such as maps and collaboration objects.

The samples provided with this reference application are for WebSphere Commerce 5.6, Business Edition. However, the same principal and pattern can apply to WebSphere Commerce 5.6, Professional Edition.

This guide is divided into the following sections:

Chapter 1. Introduction

A brief overview of the integration of WebSphere Commerce Version 5.6, Business Edition with SAP R/3 using the WebSphere InterChange Server as well as the definition of the terms used in this book, and references to related documents.

Chapter 2. Prerequisites

Lists the software and hardware prerequisites for this reference application.

Chapter 3. Integrating WebSphere Commerce business processes with SAP R/3

Describes the asynchronous and synchronous message flow between WebSphere Commerce and SAP R/3 through the WebSphere InterChange server.

Chapter 4. Installing and configuring

Describes the installation and configuration instructions for the components of this reference application.

Chapter 5. Verification procedure

A list of instructions to verify the synchronous and asynchronous message flows.

Appendix A. Messages integrated for this reference application

A list of the asynchronous and synchronous messages that this reference application provides.

Appendix B. Generic business objects

Contains a list of the generic business objects enabled by this reference application.

Appendix C. Application specific business objects

Contains a list of the application specific business objects (ASBOs) enabled by this reference application.

Appendix D. Collaboration objects and ports

A list of the collaboration templates used by this reference application.

Appendix E. Maps with specific values for WebSphere Commerce and SAP R/3

A list of the hard coded WebSphere Commerce and SAP R/3 fields used in the maps provided with this reference application.

Appendix F. Binding maps to business objects

A list of the maps that you must explicitly bind to the business objects for the WebSphere Business Integration Adapter for WebSphere Commerce and WebSphere Business Integration Adapter for mySAP.com.

Conventions used in this guide

This guide uses the following conventions:

Boldface type	indicates commands or graphical user interface (GUI) controls such as names of fields, buttons, or menu choices.
monospaced type	indicates examples of text that you enter exactly as shown.
<i>Italic type</i>	is used for emphasis and for variables for which you substitute your own values.

Default paths

This guide uses the following default installation paths:

WC_installdir

This indicates the installation path for WebSphere Commerce. When you see this variable, substitute the installation path for your installation of WebSphere Commerce. For Windows®, substitute C:\IBM\WebSphere\CommerceServer56

WAS_installdir

This indicates the installation path for WebSphere Application Server. When you see this variable, substitute the installation path for your installation of WebSphere Application Server. For Windows, substitute C:\IBM\WebSphere\AppServer

WICS_installdir

This indicates the installation path for the WebSphere InterChange Server. When you see this variable, substitute the installation path for your installation of WebSphere InterChange Server. For Windows, substitute C:\IBM\WICS

Chapter 1. Introduction

This chapter gives an overview of the integration of WebSphere Commerce 5.6, Business Edition with SAP R/3 using the WebSphere InterChange Server. It defines the terms used in this book and provides references to other related sources of information.

Terminology

The following terms are used in this guide:

Intermediate Documents (IDocs)

Intermediate Documents (IDocs) are used to exchange data between SAP R/3, and non-SAP systems. This is the document format that the SAP R/3 system understands.

BAPI BAPIs (Business Application Programming Interfaces) are open business-oriented programming interfaces that external systems can use to access business processes and data in the SAP R/3 system.

Reference data

Reference data is a sample set of products and items included in the reference application that can be used for demonstrations. It is included as part of the sample store, and also as a delimited file that can be loaded onto the SAP R/3 system.

Note: In this guide, the following are used interchangeably:

- WebSphere Commerce and WebSphere Commerce Version 5.6, Business Edition
- WebSphere Business Integration Adapter for WebSphere Commerce and adapter for WebSphere Commerce
- WebSphere Business Integration Adapter for mySAP.com and adapter for mySAP.com
- SAP and SAP R/3
- Adapter and connector

Overview

As companies use the Internet to open their enterprises to customers, partners, and suppliers, for greater efficiency and productivity such as linking a purchasing department to outside vendors or adjusting inventory levels to match the latest customer buying patterns, they encounter layers of different enterprise systems that have been purchased over time, many of which are unable to operate together.

This reference application is designed to address one such integration process. It integrates SAP R/3 core business processes with the WebSphere Commerce sell-side solution, using the WebSphere InterChange server capabilities. This application provides the e-commerce functionality of WebSphere Commerce, along with the ERP functionality of the SAP R/3 enterprise system as the supplier and fulfillment center. This solution allows you to build, deploy and integrate high-performance Web sites with advanced e-business features using open standards.

WebSphere Commerce also extends the scope of enterprise applications such as SAP R/3 by providing a reliable, scalable, and open-standards based commerce front-end. For example, an organization using SAP R/3 for enterprise functions, when integrated with WebSphere Commerce, can easily add the Internet as a new sales channel for its products and services. In the scenario described below, the WebSphere Commerce server sell-side is an e-commerce Web front-end to the SAP R/3 enterprise system.

The WebSphere InterChange server system's distributed hub-and-spoke architecture offers process collaborations that automate and streamline business processes, sophisticated business object management, inter-applications connectivity, and data integration. WebSphere InterChange server with its dynamic configuration capability automatically propagates to the respective adapters or components. WBI adapters provide swift integration to packaged, legacy, and mainframe applications as well as e-business enabling technologies.

Business scenario-integration with the seller's backend system

WebSphere Commerce contains information about the products and services of the seller, and the profile and registration information of the customers. WebSphere Commerce processes the buyer's requests, such as placing an order, modifying profile information, checking availability and querying for the status of orders. SAP contains information about the seller's products and fulfillment status. As a result, information will be exchanged between WebSphere Commerce and SAP R/3. The new customers created and material data in the SAP system will be uploaded onto WebSphere Commerce on a regular basis. The nature of transactions in this scenario allows WebSphere Commerce to be updated with material and customer data in the SAP system, and alternatively, SAP to be notified of the buyer's requests in WebSphere Commerce.

Extensibility

This reference application consists of two parts: Integrating WebSphere Commerce with the WebSphere InterChange server and integrating the WebSphere InterChange server with the SAP backend system.

This reference application can be easily extended to integrate WebSphere Commerce with other backend systems. This requires you to modify the WebSphere InterChange server maps and connector configuration, specific to the system that is used to connect to your backend system.

Business models enabled

In this integration, WebSphere Commerce provides an e-commerce front-end Internet sales channel to the SAP R/3 enterprise system. Any customer registered with the SAP system can use WebSphere Commerce as the front end to browse and shop for products that are loaded from SAP onto the WebSphere Commerce site.

Using this capability, buyers in any part of the world can shop online for products using the online stores and catalog display functionality provided by WebSphere Commerce. From the WebSphere Commerce site, buyers can place orders, check the price and availability, query for the status of their orders and other relevant information that is present in the SAP system. This synchronization is possible by the initial upload of material data from SAP to WebSphere Commerce. Connectivity in the current implementation enables customer data, material data, and order status from SAP to be updated in WebSphere Commerce through a set

of messages. Integrating WebSphere Commerce to an SAP backend system allows sellers to use the WebSphere Commerce rich functionality in personalization, marketing, merchandising, product management, and user management to provide robust Internet selling sites for B2B direct and consumer direct clients.

Benefits

The following are the benefits of integrating WebSphere Commerce with SAP R/3 using the WebSphere InterChange server:

- Create channel specific business processes in WebSphere Commerce and change them rapidly without having to change the backend system.
- Easily and quickly add an Internet sales channel to the enterprise backend system.
- Develop Web channel specific processes that augment the backend process in a flexible manner, adding and collecting data specific for the Web channel without changes in the backend system. This allows more dynamic changes in the Web channel process without any impacts on the backend system. Additional product information such as long or short descriptions, which is not included in SAP, can be added in WebSphere Commerce. This information may be required for specific uses in a Web channel.
- Use WebSphere Commerce as a Web channel to enable customers to integrate SAP and other backend systems providing a single Web channel integrated business process.
- Provide customers with access to Web site functions such as browsing catalogs, placing orders, and making online payments.
- Create online catalogs in WebSphere Commerce from SAP materials.
- Provide multicultural support and personalized content based on profile, history, demographics, and other factors. Synchronize product and customer information between the front end and backend systems.
- Create orders in WebSphere Commerce and send the orders to SAP for order processing and fulfillment. Check the status of the order with SAP from the WebSphere Commerce site.
- Leverage the complex business processes available in the SAP enterprise system.
- Create commerce sites leveraging WebSphere Commerce product management, customer management, personalization, and merchandising capabilities.
- Manage complex relationships between buyers and sellers, organizations and sub organizations, which helps strengthen relationships with customers, partners, and suppliers.
- Provide highly scalable commerce storefront capabilities utilizing information from a SAP backend system while limiting the load on the SAP system.
- Provide operational and business analytical information based on site statistics, usage scenarios, campaign effectiveness, demographics, and other factors.

Functionality

This reference application provides the following functionality:

- **Order creation:** Buyers can create orders in WebSphere Commerce and the details of the order are sent to SAP in the IDoc format for further processing.
- **Order status:** Whenever there is a change in the status of an order, a message conveying the same can be triggered from SAP and sent to WebSphere Commerce. The three order status messages supported are:
 - Order Confirmation
 - Order Delivery

- Order Invoice
- **Customer creation:** When new customers are created in SAP, the details can be registered in WebSphere Commerce by sending the CustomerCreate message from SAP to WebSphere Commerce.
- **Customer update:** When existing customer information is updated in SAP the changes can be updated in WebSphere Commerce using the CustomerUpdate message.
- **Product price update:** When the product prices in SAP change, WebSphere Commerce can be notified of the changes using the ProductPriceUpdate message.
- **Product inventory update:** Changes in product inventory in SAP can be sent to WebSphere Commerce using the ProductInventoryUpdate message.
- **Checking the inventory availability:** This message is used to check the inventory availability of a particular product in SAP from WebSphere Commerce using the CheckInventoryAvailabilityBE message.

References

Apart from this guide, the following reference documents are available with their respective products:

- Messaging system information related to the WebSphere Commerce Business Edition can be found in the product documentation at http://www.ibm.com/software/webservers/commerce/wc_be/
- For SAP R/3 documentation refer to <http://help.sap.com>.
- For WebSphere InterChange Server refer to <http://www.ibm.com/software/integration/wbiserver/ics/library/infocenter/>
- For information on the WebSphere Business Integration Adapters refer to <http://www.ibm.com/software/integration/wbiadapters/>
- For WebSphere MQ documentation refer to <http://www.ibm.com/software/ts/mqseries/messaging>

Note: The preceding Web addresses can change at any time without notice. IBM is not responsible for the authenticity or correctness of information from non-IBM Web sites.

Chapter 2. Prerequisites

This section covers the software and hardware prerequisites for this reference application. This reference application assumes a Windows operating environment.

Software prerequisites

The software prerequisites include:

WebSphere Commerce 5.6 Business Edition

WebSphere Commerce 5.6, Business Edition is an e-commerce software that has various subsystems. The messaging system gives WebSphere Commerce the ability to communicate with an external environment. This communication includes sending and receiving messages to and from backend systems. This is achieved through the following components:

- A listener for WebSphere MQ to process inbound requests.
- An adapter for WebSphere MQ for outbound requests to allow you to integrate with backend and external systems.
- The adapter for WebSphere InterChange Server offers a mechanism to extend WebSphere Commerce business integration with WebSphere InterChange Server using the WebSphere InterChange Server, Server Access Interface (SAI). With this adapter, WebSphere Commerce can integrate with external systems by sending synchronous messages.

WebSphere MQ 5.3

WebSphere MQ (formerly MQSeries®) is used as the transport middleware to communicate with various external systems, including the WebSphere InterChange server. Refer to the *WebSphere Commerce Additional Software Guide* for information on setting up WebSphere MQ for WebSphere Commerce.

WebSphere InterChange server 4.2.2

The WebSphere InterChange Server (formerly IBM CrossWorlds®) is a suite of software integration products that include prebuilt modules for common business integration requirements, and development and management tools. These products provide connectivity for leading e-business technologies and enterprise applications. This reference application uses the complete WebSphere InterChange Server setup, which includes the WBI Adapters.

WebSphere Business Integration Adapters 2.3.

The following are the WebSphere Business Integration Adapters required:

WebSphere Business Integration Adapter for WebSphere Commerce

The adapter for WebSphere Commerce enables the WebSphere InterChange Server to exchange messages with WebSphere Commerce. For more information refer to the WBI adapter documentation.

WebSphere Business Integration Adapter for mySAP.com

The adapter for mySAP.com enables the WebSphere Interchange server to exchange messages with the SAP system.

Port connector

The port connector is used for binding to the collaboration ports, if no specific processing is expected on those ports.

SAP R/3 4.6

This is an ERP system that contains the master data such as catalog and customer information. WebSphere Commerce Business Edition provides the e-commerce functionality. The SAP system interacts with external applications by exchanging information in the form of messages. It generates IDocs that external applications use and accepts IDocs or BAPI requests from other applications to be processed by the SAP R/3 system.

Hardware prerequisites

For information on hardware prerequisites, refer to the documentation that comes with the appropriate software.

Chapter 3. Integrating WebSphere Commerce business processes with SAP R/3

This section describes the flow for the outbound messages from WebSphere Commerce and inbound messages to WebSphere Commerce that allow integration of WebSphere Commerce business processes with SAP R/3. It explains how messages are used to transport information through various components of this reference application.

WebSphere Commerce outbound process

Transactions in WebSphere Commerce can trigger messages to be sent to other applications through the WebSphere Commerce messaging system. In this reference application, the OrderCreate XML message can be generated from WebSphere Commerce to create an order in the SAP system.

In this reference application, when an order is created in WebSphere Commerce, it can be configured such that the order related information is sent to another system for further processing like fulfillment. Orders are sent through XML messages from WebSphere Commerce to WebSphere InterChange server. The WebSphere InterChange server processes this XML message and sends it to other systems like SAP through the application connectors.

When you place an order in WebSphere Commerce:

1. The OrderCreate message in XML format is generated and placed in the WebSphere Commerce remote queue (WebSphere MQ queue) as shown in Figure 1 on page 8.
2. The WBI WebSphere Commerce connector agent constantly polls for new messages in its input queue, which it passes to the WebSphere Commerce connector controller.
3. The controller receives the WebSphere Commerce specific business objects from the XML data handlers and invokes the maps that are bound to generate the generic business objects (GBO).
4. The GBOs are passed to the corresponding collaboration object, which processes them.
5. The processed generic business objects are sent to the adapter for mySAP.com controller, which uses the maps to create SAP specific business objects and passes them to the adapter for mySAP.com agent in SAP specific format.
6. The adapter for mySAP.com agent sends the message to the SAP R/3 system, to create the order. Similar flows apply to other ERP systems.

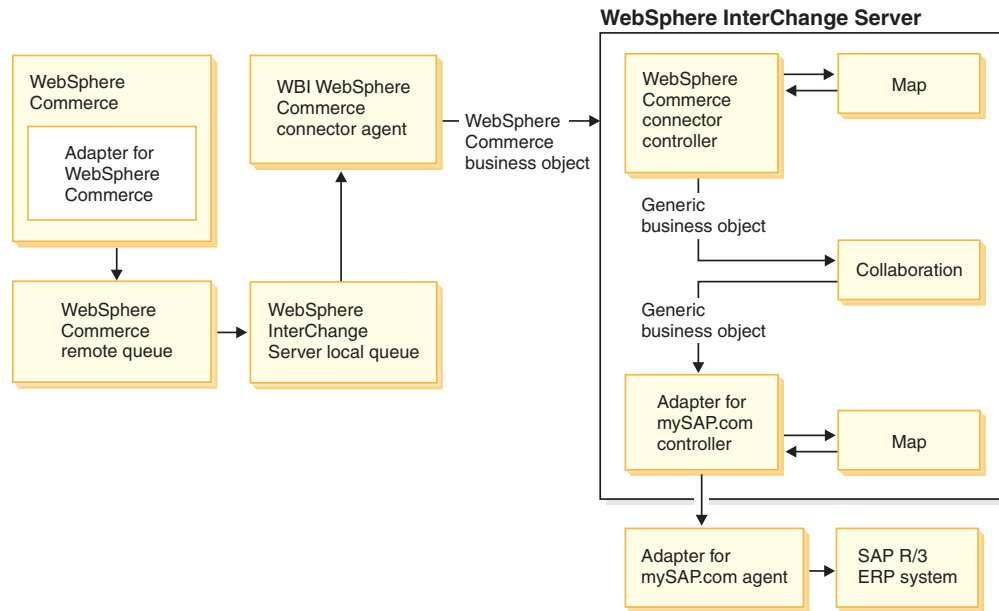


Figure 1. Asynchronous message flow from WebSphere Commerce

WebSphere Commerce inbound process

WebSphere Commerce supports processing inbound message requests to invoke certain business logic by calling WebSphere Commerce commands. In this reference application, the existing WebSphere Commerce messages are used to invoke business logic that SAP R/3 triggers.

During business processing in SAP, you can configure SAP to generate messages that will update WebSphere Commerce.

The WebSphere Commerce inbound process involves the following:

1. The WBI Adapter for mySAP.com agent polls for messages, receives and forwards them to the adapter for mySAP.com controller as shown in Figure 2 on page 9.
2. The controller receives the messages and invokes the maps to convert the SAP business objects to generic business objects that the collaboration object processes.
3. The collaboration sends the GBOs to the WebSphere Commerce connector controller, which invokes the map to convert the GBOs to application specific business objects.
4. The WebSphere Commerce connector agent converts the application specific business objects to an XML message, which is sent to WebSphere Commerce.

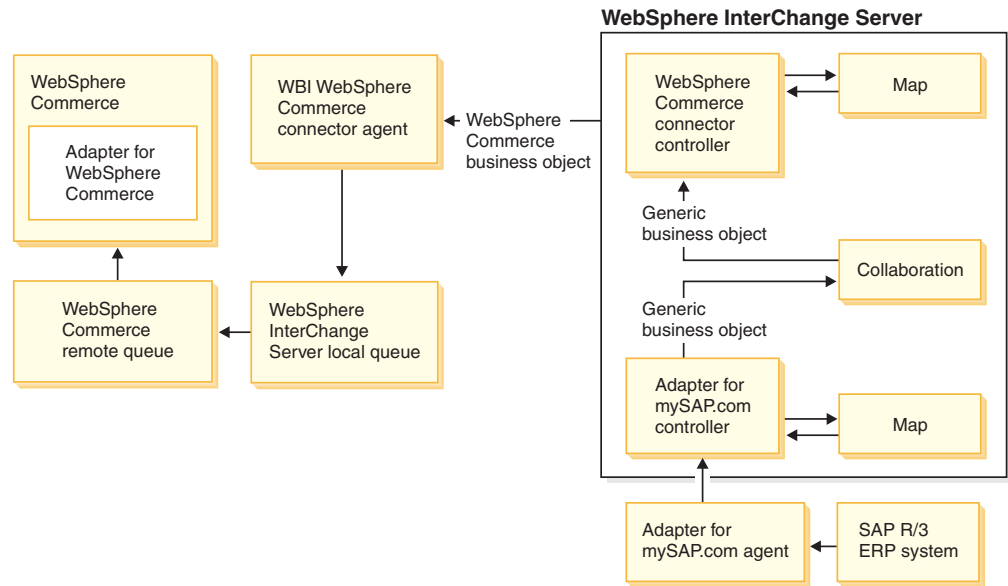


Figure 2. Asynchronous message flow into WebSphere Commerce

WebSphere Commerce request/reply process

The WebSphere Commerce messaging system allows business logic to initiate request/reply interactions with other systems when information is required in real time from external systems. In this example, the check product availability interface for external systems is used in combination with the WebSphere Commerce adapter for the WebSphere InterChange server to receive information about the availability of products from SAP R/3. This model can be used to initiate other interactions with external systems linked to the WebSphere InterChange server. In this reference application, the CheckInventoryAvailabilityBE synchronous message flow is described, see Figure 3 on page 10.

For more information about the adapter for WebSphere InterChange Server, refer to the *WebSphere Commerce Additional Software Guide*.

Sending a request

During normal order processing, WebSphere Commerce checks for the availability of products internally with corresponding inventory records. When you use the interface for checking product availability in an external system like SAP from WebSphere Commerce, the following takes place:

1. An XML message is generated and passed to the adapter for WebSphere InterChange Server in WebSphere Commerce.
2. The adapter for WebSphere InterChange Server sends the message to the Server Access Interface (SAI) in the WebSphere InterChange server.
3. The SAI invokes the data handler to generate the WebSphere Commerce specific business object.
4. This business object is passed to the subscribed map that generates the GBOs.
5. The GBOs are passed to the collaboration object specified in the adapter for WebSphere InterChange Server configuration properties.
6. The collaboration object processes the business object.

7. The processed generic business objects are then sent to the Adapter for mySAP.com controller, which invokes the related maps to create the SAP specific business objects and passes them to the adapter for mySAP.com agent.
8. The adapter for mySAP.com agent makes the Material Availability BAPI call using the application specific business objects to the SAP R/3 system.

Receiving a response

The SAP system responds with the status of the inventory availability that reaches the adapter for mySAP.com agent. Receiving a response involves the following:

1. The adapter for mySAP.com agent receives the message and generates the application specific business objects.
2. The connector agent sends the business objects to the adapter for mySAP.com controller.
3. The controller invokes the maps to convert the SAP business objects to generic business objects.
4. The controller then passes the generic business objects to the collaboration object, which processes the business objects and sends the processed business objects to the WebSphere InterChange Server SAI.
5. The SAI then generates the XML message, using data handlers, and responds to the transport adapter in WebSphere Commerce.

This model can be used for other request/reply interfaces initiated from WebSphere Commerce.

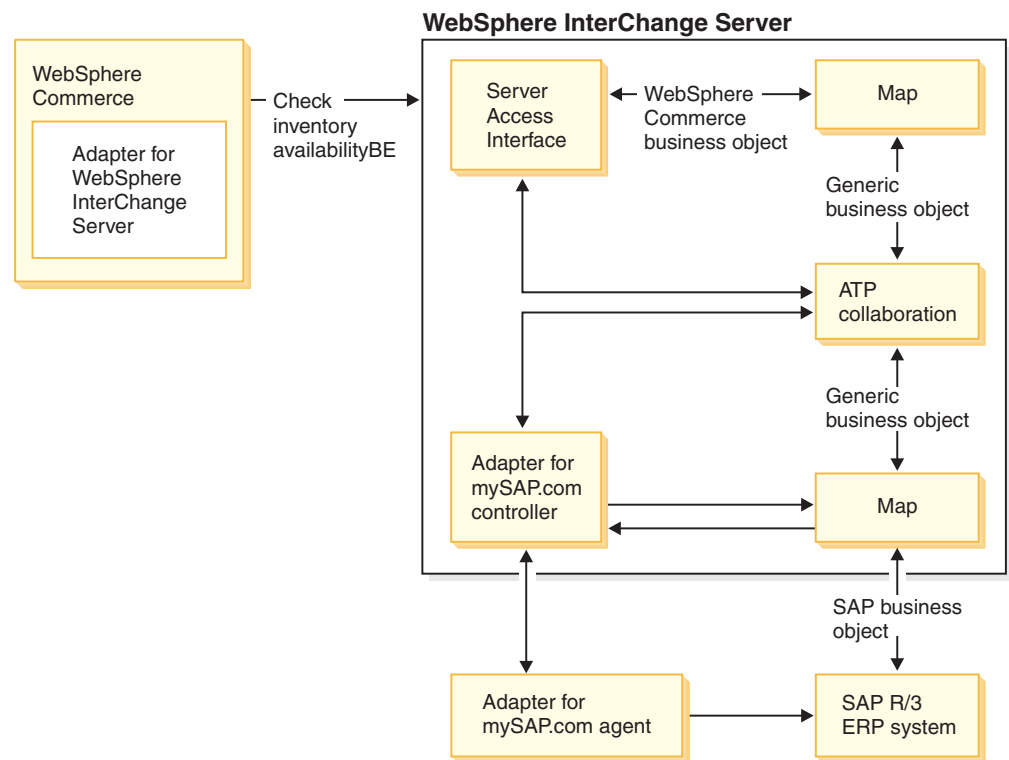


Figure 3. Request/reply message flow from WebSphere Commerce to SAP

Chapter 4. Installing and configuring

To enable the WebSphere Commerce—SAP integration using the WebSphere InterChange server, you must perform the following activities:

- Install and configure WebSphere Commerce.
- Install and configure WebSphere MQ.
- Configure the WebSphere Commerce messaging system.
- Publish the customized WICSSAPAdvancedB2BDirect business direct store.
- Synchronize product data between WebSphere Commerce and SAP.
- Install and configure WebSphere InterChange server.
- Generate and load the SAP application specific business object from your SAP system.
- Configure the SAP R/3 system.

Sample topology

Before you begin your installation you must decide on the hardware topology for this reference application. The following is the sample topology used for this integration:

Table 1. Sample topology

Machine 1	Machine 2
WebSphere Commerce Business Edition	WebSphere InterChange server
WebSphere MQ	WebSphere MQ
	WebSphere Business Integration adapter for mySAP.com
	WebSphere Business Integration adapter for WebSphere Commerce

Installing and configuring WebSphere Commerce

Install WebSphere Commerce 5.6, Business Edition. Refer to the *WebSphere Commerce Installation Guide* for the installation instructions and the post-install configuration.

You must update the CMDREG table, which is the command registry table in your WebSphere Commerce database to use the XML message format.

To enable the Report_NC_PurchaseOrder message, which is the OrderCreate XML message, update the CMDREG table using the following SQL statement:

```
update cmdreg set classname =  
'com.ibm.commerce.messaging.commands.SendXMLOrderCmdImpl'  
where interfacename = 'com.ibm.commerce.order.commands.OrderMessagingCmd'
```

Note: The above commands will take effect only after restarting the server or refreshing the registry using the WebSphere Commerce Administration Console.

Installing and configuring WebSphere MQ

Install WebSphere MQ 5.3 using the documentation provided with the WebSphere MQ product.

To configure WebSphere MQ to work with WebSphere Commerce, WebSphere Commerce requires a queue manager and a minimum of five queues for integration. The five queues include:

Table 2. Queues

Queue	Queue description
Error	Default error queue. Collects erratic inbound messages.
Inbound	Used by SendReceiveImmediate mode of the adapter for WebSphere MQ.
Parallel inbound	Any message arrive at this queue will be processed in parallel manner.
Serial inbound	Any message arrive at this queue will be processed in serial manner based on first-in-first-out.
Outbound	Used for WebSphere Commerce initiated outbound messages and reply messages from WebSphere Commerce.

This reference application uses the following:

- A queue manager
- Serial inbound queue
- Outbound queue

For instructions on creating a queue manager and queues, refer to the WebSphere MQ documentation.

For the sample topology provided ensure that:

1. your outbound queue must be created as a remote queue to enable communication with the remote WebSphere InterChange server.
2. The serial inbound queue must be created as a local queue to receive messages from the WebSphere InterChange server.

Similarly, you must create a queue manager and queues for the WebSphere InterChange server. The queues required to communicate with the WebSphere Commerce messaging system components are:

- A local queue definition, for processing the messages coming from the WebSphere Commerce system.
- A remote queue definition that will be mapped to the WebSphere Commerce serial inbound queue. This queue will send messages to the WebSphere Commerce system.

The following table shows how the queues must be mapped between WebSphere Commerce and WebSphere InterChange server:

Table 3. Mapping queues between WebSphere Commerce and WebSphere InterChange server

WebSphere Commerce	WebSphere InterChange server
Outbound queue (remote queue)	ICS_Inbound queue (local queue)

Table 3. Mapping queues between WebSphere Commerce and WebSphere InterChange server (continued)

WebSphere Commerce	WebSphere InterChange server
Serial inbound queue (local queue)	ICS_Outbound queue (remote queue)

Additionally, you need to create MQ channels for communication between the two WebSphere MQ servers. See, “Installing and configuring WebSphere MQ” on page 12. For more information refer to WebSphere MQ documentation on inter-communication.

After you create the queues, follow the instructions given in the WebSphere MQ section from the *WebSphere Commerce Additional Software Guide*. The instructions include information on how to use WebSphere Commerce and WebSphere Application Server with WebSphere MQ.

Configuring the WebSphere Commerce messaging system

The WebSphere Commerce messaging system is equipped to handle messages to interact with backend systems. Ensure that you have completed the steps in “Installing and configuring WebSphere MQ” on page 12. Configuring the WebSphere Commerce messaging system involves:

1. Start the WebSphere Commerce Administration Console. Login as a Site Administrator, go to the Configuration section and choose the Transport option. Select WebSphere MQ as your transport and change the status to active.
2. Log out from the Administration Console.

Note: When you complete publishing the store as described in “Publishing the store” on page 14, log into the WebSphere Commerce Administration Console as a Store Administrator and select the WICSSAPAdvB2BDirect store. From the **Configuration** section add **WebSphere MQ** to the store.

Publishing the WICSSAPAdvancedB2BDirect store

This section covers the following:

1. Instructions required before you publish the WICSSAPAdvancedB2BDirect store.
2. Publishing the WICSSAPAdvancedB2BDirect store

Before publishing the store

Before publishing the store unzip WICSSAPAdvancedB2BDirect.sar available in the WC56SAPWBIRRefApp.zip into any directory.

Note: For information on the directory structure, refer to the readme.txt that comes with the integration package that you have downloaded.

Populating the units of measure

This reference application provides a massloadable XML file containing the units of measures defined in SAP. Use this XML file to upload the SAP units of measures into WebSphere Commerce. Complete the following on the machine where you have installed WebSphere Commerce:

1. The UnitOfMeasure.xml file provides a sample value. Edit the UnitOfMeasure.xml file in \store\uom directory in a text editor to include the UOM values from your SAP system.
2. Edit the ImportUOM.bat file from the \store\uom directory in a text editor and change the class paths and database information in accordance with your WebSphere Commerce installation. Save this file.
3. From a DB2® command window, go to the \store\uom directory and run the following command:

```
ImportUOM -infile UnitOfMeasure.xml -method sqlimport
```

This populates the unit of measures in the WebSphere Commerce tables according to the standard SAP R/3 installation. The UnitOfMeasure.xml file is present in the \store\uom directory.

Publishing the store

The WICSSAPAdvancedB2BDirect.sar provided with this solution uses WebSphere Commerce Payments. For details about installing and configuring WebSphere Commerce Payments refer to the *WebSphere Commerce Installation Guide*.

Note: It is recommended that you take a backup copy of your database before publishing your store.

Creating a new store

The WICSSAPAdvancedB2BDirect.sar file is present in the \store directory. This store archive file is built on top of the business direct store model that comes as part of the WebSphere Commerce Business Edition standard installation.

To enable the sample store model and the catalog provided with this reference application do the following:

1. Copy the WICSSAPAdvancedB2BDirect.sar file from the \store directory, to WC_installdir\starterstores\AdvancedB2BDirect folder.
2. Copy the WICSSAPHardwareCatalogData.sar file from the \store directory, to WC_installdir\starterstores\AdvancedB2BDirect\samplecatalogs folder.
3. Edit WC_installdir\xml\tools\devtools\SARRegistry.xml to add the following lines in the section for the business direct sample store models:

```
<SampleSAR fileName="WICSSAPAdvancedB2BDirect.sar"
relativePath="AdvancedB2BDirect">
<view name="BusinessDirect"/>
</SampleSAR>
```
4. Save and close the file.
5. Launch the Administration Console to publish this store.
6. In the Store Archives page select the **B2B direct** view.
 - a. Select **WICSSAPAdvancedB2BDirect.sar** and navigate to the next page.
 - b. Select the catalog identified by WICSSAP from the available list of options and publish the SAR file. It may take a few minutes to complete publishing the store archive file. When complete, the status changes from **Publishing** to **Publishing completed successfully**.

Disabling SAP specific store customization

In the sample store provided, an SAP specific shopping flow is introduced. If you want to disable these store customization to revert to the default behavior, then you can disable the SAP feature. Refer to the WebSphere Commerce information center for more information on Enabling and Disabling Store Features. By default, the SAP feature is enabled, which is a prerequisite to use this reference application.

For more information on store customization for this reference application, see “Appendix G. Store customization” on page 49.

Configuring the SAP R/3 system

This section covers the configurations required in the SAP R/3 system for this reference application. Typically, a SAP consultant configures the various settings. The information provided here is a quick reference to the different configurations required in the SAP R/3 system.

For more information on how to configure each of the following, refer to the SAP documentation or the SAP online help:

Defining an organization structure

For this integration, you can use your existing organization structure or create a new one. Some of the required entities are Company code, Controlling Area, Plant, Storage location, sales area, and shipping point.

Customizing master data

Create a new account group and make the REGION and TERMS OF PAYMENT fields mandatory. In addition, assign the sold-to party, ship-to party, bill-to party, and payer partner functions to the new account group.

Configuring price, freight, and tax

Create the necessary condition tables, condition records, and access sequences for price, freight, and tax calculation.

Customizing order numbers

You must customize the corresponding field in SAP to store and send the WebSphere Commerce order number in the order status messages, for example, SISCO01, SISDEL01, and SISINV01 IDocs.

Configuring partner profile communication

Configure the partner profile communication to enable the SAP ALE layer to communicate with the external system; send and receive requests from the WebSphere Business Integration Adapter for mySAP.com. The configurations include defining logical systems, RFC destination, maintaining partner profiles, creating and assigning message types, and enabling change pointers.

Synchronizing product data between WebSphere Commerce and SAP

To synchronize product data between WebSphere Commerce and SAP after you complete publishing the store and configuring SAP, you can do one of the following:

- Load the reference data provided with the sample WICSSAPAdvancedB2BDirect store onto the SAP system.
- Load the material data from the SAP system onto WebSphere Commerce.

Loading sample store data onto SAP

To demonstrate this integration you can also use the reference data provided with this reference application. A portion of the sample store model items is packaged as reference data. You must import this data into the SAP R/3 system. This allows the synchronization of WebSphere Commerce and SAP R/3 material data.

BDC_MaterialInput.txt in the store\dataload\referencedata directory contains the material reference data that you must load onto SAP using a Batch Data Conversion (BDC) program in the following format:

Note: The prerequisite to load material data is to define the material groups in the SAP R/3 system. Use "SAP Customizing" to do this.

Table 4. Material reference data

Serial number	Description	Length	Depends on existing SAP configurations and data
1	Material Number	18	No
2	Industry Sector	1	Yes
3	Material Type	4	Yes
4	Plant	4	Yes
5	Sales Organization	4	Yes
6	Distribution Channel	2	Yes
7	Material Description	40	No
8	Unit of Measure	3	Yes
9	Material Group	9	Yes, you need to define this using SAP customizing
10	General item Category Group	4	Yes
11	Gross Weight	17	No
12	Weight Unit	3	Yes
13	Net Weight	17	No
14	Size/Dimensions	32	No
15	Description Language	16	Yes
16	Document Number	22	No
17	Class Type	3	Yes
18	Class Number	18	Yes
19	Cash Discount Indicator	1	No
20	First Entry Displayed	3	No
21	Delivering Plant	4	Yes
22	Item Category Group from Mat master	4	Yes
23	Checking Group for Availability Check	2	Yes
24	Transport Group	4	Yes
25	Loading Group	4	Yes

Table 4. Material reference data (continued)

Serial number	Description	Length	Depends on existing SAP configurations and data
26	MRP Type	2	Yes
27	MRP Controller	3	Yes
28	Lot Size	2	No
29	Procurement Type	1	Yes
30	In-house Production Line	3	No
31	Scheduling Margin Key for Floats	3	Yes
32	Period Indicator	1	No
33	Planning Strategy group	2	No
34	Total Replenishment Lead time	3	No
35	Valuation Class	4	Yes
36	Price Control Indicator	1	No
37	Price Unit	6	No
38	Standard Price	18	No

BDC_SellingPriceInput.txt in the store\dataload\referencedata directory contains the standard price for material reference data that you must load onto SAP using a BDC program in the following format:

Note: The prerequisite to load this data is to define the price condition in the SAP system.

Table 5. Standard price for material reference

Serial number	Description	Length	Depends on existing SAP configurations and data
1	Pricing Condition Type	4	Yes
2	Material Number	18	No
3	Material Selling Price	16	No
4	Currency	5	No

To import reference data into SAP as a batch process from the input file, you need a BDC program. This program requires you to record the transaction to load the data. To do this, use transaction SHDB and do the following:

1. Using the SAP client, from the SAP Easy Access screen go to SHDB transaction and enter the record name. Follow the on screen instructions to record the MM01 transaction and then record the VK12 transaction.
2. Based on the values in your SAP installation, make the necessary changes to the data in the reference data files before importing.
3. Use transaction SE38 to execute the recorded programs.
4. After loading the reference data, you have to create the inventory records before placing any orders for these materials.

Loading material data onto WebSphere Commerce

To load the material data onto WebSphere Commerce using the massload scripts provided, ensure that you extract the material data from the SAP R/3 system in the following sequence with a delimiter. The massload script expects the "~" delimiter. If you are using a different delimiter, you need to change this in the import schema files provided with this reference application.

Note: The sample store provided with this reference application includes the reference data. The instructions mentioned here apply only if you want to import any other materials that exist in your SAP system.

Table 6. Sequence to extract material data from SAP

Sequence number	Data at position	Data description
1	MaterialPartNumber	Part number of the material
2	MaterialGroupPartNumber	Part number of the material group
3	Language	Language specification in SAP. An example of language specification in WCS is en_US, for English.
4	MaterialName	Name of the material
5	MaterialShortDescription	Short description of the material
6	MaterialLongDescription	Long description of the material
7	MaterialImageName	Filename of the image/picture of the material.
8	MaterialLastUpdatedOn	Indicates the last time the material was updated.
9	MaterialPrice	Amount of the material price.
10	Currency	Currency of the material price.
11	MaterialWeightMeasure	The unit of measurement for weight.
12	MaterialSizeMeasure	The unit of measurement for length, width and height.
13	MaterialQuantityMeasure	The unit of measure for nominal quantity.
14	MaterialWeight	The nominal weight associated with the material
15	MaterialLength	The nominal length associated with the material.
16	MaterialWidth	The nominal width associated with the material.
17	MaterialHeight	The nominal height associated with the material.
18	MaterialNominalQuantity	Nominal quantity for a material, used for pricing. For example, if a material is priced as "3 for a dollar", then the nominal quantity of the material is 3, and the price of the material is one dollar.

Table 6. Sequence to extract material data from SAP (continued)

Sequence number	Data at position	Data description
19	MaterialDataIndicator	An indicator that specifies whether the data for that material is for CREATE or UPDATE
20	ManufacturerName	The name of the manufacturer of this material
21	ManufacturerPartNumber	The part number used by the manufacturer to identify this material

The attributes information for items in WebSphere Commerce is optional. You can load the materials without attributes. If you are loading the attributes for materials, then extract the characteristics information for these materials in the following format with the delimiter "~". If you are using a different delimiter, you need to change this in the import schema files provided with this reference application.

Table 7. Format to extract material attribute information from SAP

Sequence number	Data at position	Data description
1	MaterialPartNumber	Part number of the material
2	MaterialGroupPartNumber	Part number of the AUSP-KLART (Mandatory) material group
3	MaterialCharacteristicName	Name given to the material characteristic.
4	MaterialCharacteristicValue	Value of the respective material characteristic name Language specification in SAP.
5	Language	An example of language specification in WCS is en_US, for English.

To upload material data in the specified format do the following:

1. Write an SAP program to extract the material data following the sequence and delimiter described in Table 5 on page 17 and Table 6 on page 18. Move the extracted files into the \store\data\load\material directory.
2. Open ManifestFile.txt from the \store\data\load\material directory, and replace itabmara.txt with the output file name derived from the material information extract program. Replace itab.txt with the output file name derived from the material characteristics extract program. If the program for material characteristics is not executed then delete the following lines:

```
itab.txt,Import_cif_Schema10.xml,Output.xml,Append
itab.txt,Import_cif_Schema11.xml,Output.xml,Append
```

Note: For information on the directory structure, refer to the readme.txt in WC56SAPWBIRRefApp.zip.

3. Save and close this file.
4. Open the batch file MaterialUpload.bat from the \store\data\load\material directory, in an editor and change the following parameters in accordance with your WebSphere Commerce installation:

- *DB_NAME*—The type of database, which is DB2 in this case.
 - *WCS_DBNAME* —WebSphere Commerce instance database name, for example, mall.
 - *WCS_DBUSER*—Database user ID.
 - *WCS_DBPWD*—Database user password.
5. Edit the following line to include your commerce installation path:
`WC_installdir\bin\setenv.bat`
 6. Change the following literal in accordance with your installation. For example,
`set DB2_HOME=D:\WebSphere\sql\lib`
 7. Change the parameter values passed to the XMLTransformer according to your installation. You can find the values for these parameters in the WebSphere Commerce database tables.
 - *MemberIdValue*—Identifier of the storeowner (ORGENCY.ORGENTY_ID).
 - *TradingPositionName*—Trading position name associated with the store (TRADEPOSCN.NAME).
 - *CatalogName*—Catalog identifier of the store (CATALOG.IDENTIFIER).
 - *ImportLocation* `WC_installdir\schema\xml\wcs.dtd`—This is the location of the wcs.dtd file.
 - *StoreIdentifier*—Identifier of the store published to showcase this reference application (STORE.STORE_ID).
 - *FulfillmentCenterName*—The fulfillment center name that is associated with the store (FFMCENTER.NAME).
 8. Save the changes made and run the MaterialUpload.bat batch file from a DB2 command window, on the machine where you have installed WebSphere Commerce.
 9. Launch the store and check for the products and items under **SAP Products⇒ SAP Category 1 hierarchy**.

Note: Any item that is not specifically grouped under a product in SAP can be found under the product SAP10001 in the same hierarchy.

When manufacturer details are not present for a product, by default the product takes "SAP Tools" as the manufacturer name and product part number as the manufacturer part number.

When the short description details are not available for a product, by default, the material name is used as the short description.

When massloading the material data, only names of the product images are loaded. To view the product images on the corresponding page of the store you must manually copy the image files into the following directory:

```
WAS_installdir\installedApps\  
WC_Enterprise_App_instance_name.ear\wcstores.war\store_name\images.
```

Where, *instance_name.ear* is the name of the commerce instance in your installation and *store_name* is the name of the store to which the materials were uploaded.

Installing and configuring the WebSphere InterChange server

To install and configure the WebSphere InterChange server, refer to the *WebSphere InterChange Server System Installation Guide for Windows* provided with the product. Additionally, install the following:

- WebSphere Adapter Framework 2.3
- WebSphere Commerce connector agent
- mySAP.com connector agent

Importing the InterChange server components

This reference application comes with a set of components such as business object definitions, maps, relationships, and collaboration objects for each message. The instructions to import and deploy the components are available in “Setting up the System Manager.” For more details about the WebSphere InterChange server components refer to the corresponding appendix:

- See “Appendix B. Generic business objects” on page 39.
- See “Appendix C. Application specific business objects” on page 41.
- See “Appendix D. Collaboration objects and ports” on page 43.
- See “Appendix E. Maps with specific values for WebSphere Commerce and SAP R/3” on page 45.

Note: Some of the WebSphere Commerce and SAP system specific fields in the maps are hard coded. Change these fields to suit your WebSphere Commerce and SAP configuration.

Setting up the System Manager

Setting up the System Manager involves the following:

1. Setting up the WebSphere InterChange server environment.
2. Deploying the assets provided for this reference application.

Setting up the WebSphere InterChange server environment

When you install the WebSphere InterChange server, ensure that you choose to install all the prebuilt collaborations. The core collaborations are available in *WICS_installdir\WBICoreCollabs*. Using the core collaborations do the following to setup the WebSphere InterChange server environment:

1. Open the System Manager perspective in WebSphere Studio workbench.
2. Create an Integration Component Library called ICL_SAPREPS.
 - a. Right click **ICL_SAPREPS** and select **Import from Repository File**.
 - b. Browse to the *WICS_installdir\WBICoreCollabs* directory.
 - c. Open *BIA_BO_BaseCollabBOs.jar* and click **Finish**.
 - d. Repeat steps 2a, 2b, and 2c for the following files available in *WICS_installdir\WBICoreCollabs*:
 - *BIA_CT_ATP.jar*
 - *BIA_CT_CustomerSync.jar*
 - *BIA_CT_InventoryLevelManager.jar*
 - *BIA_CT_OrderBillingStatus.jar*
 - *BIA_CT_OrderDeliveryStatus.jar*
 - *BIA_CT_SalesOrderProcessing.jar*
 - *BIA_CT_OrderStatus.jar*

- BIA_CT_PriceManager.jar
- e. Expand **ICL_SAPREPS**⇒**Collaboration Template** and do the following:
 - 1) Double click the **ATP** collaboration template. The collaboration template opens in the Process Designer tool.
 - 2) Double click the **Definition** node in the left frame. The Template Definition window opens.
 - 3) Move to the **Properties** tab.
 - a) Expand the General Properties tree view.
 - b) Select the **ATP_OR_ATPLINE** property.
 - c) In the Value frame, select the row that contains ATP as the Value. Right click and select **Edit Value**.
 - d) Change the Value to ATPLine.
 - e) Click **Update** and then **Apply**.
 - 4) Click **Save** and exit the window.
 - f. Expand **ICL_SAPREPS**⇒**Collaboration Template** and do the following:
 - 1) Double click the **OrderDeliveryStatus** collaboration template. The collaboration template opens in the Process Designer tool.
 - 2) Double click the **Definition** node in the left frame. The Template Definition window opens.
 - 3) Move to the **Properties** tab.
 - a) Expand the General Properties tree view.
 - b) Select the **USE_RETRIEVE** property.
 - c) In the Value frame, select the row that contains true as the Value. Right click and select **Edit Value**.
 - d) Change the Value to **false**.
 - e) Click **Update** and then **Apply**.
 - 4) Click **Save** and exit the window.
 3. Setup the MO_DataHandler_Default object.
 - a. Right click **ICL_SAPREPS** and select **Import from Repository File**.
 - b. Browse to the *WICS_install\repository* directory.
 - c. Open MetaObjects.jar and click **Finish**.
 4. Setup the prebuilt port connector:
 - a. Right click **ICL_SAPREPS** and select **Import from Repository File**.
 - b. Browse to the *WICS_install\repository\port* directory.
 - c. Open CN_Port.jar and click **Finish**.
 5. Setup the prebuilt SAP connector:
 - a. From the System Manager menu select **Tools** ⇒ **Connector Configurator**.
 - b. The Connector Configurator opens in the create mode. Click **Cancel**.
 - c. From the Connector Configurator menu select **File** ⇒ **Open** ⇒ **From File**.
 - d. Browse to the *WICS_install\repository\SAP* directory.
 - e. Select CN_SAP.txt and click **Open**.
 - f. From the Connector Configurator menu select **File** ⇒ **Save** ⇒ **To project**, where project name is ICL_SAPREPS.
 6. Setup the prebuilt WebSphere Commerce connector:
 - a. From the System Manager menu select **Tools** ⇒ **Connector Configurator**.
 - b. The Connector Configurator opens in the create mode. Click **Cancel**.

- c. From the Connector Configurator menu select **File ⇒ Open ⇒ From File**.
 - d. Browse to the `WICS_install\repository\WebSphereCommerce` directory.
 - e. Select `CN_WebSphereCommerce.txt` and click **Open**.
 - f. From the Connector Configurator menu select **File ⇒ Save ⇒ To project**, where project name is `ICL_SAPREPS`.
7. Open the System Manager:
 - a. Expand **User Project** and right click **InterChange Server Project**.
 - b. Select **New ICS Project**.
 - c. In the window that opens specify a project name called `ICL_SAPREPS_UserProj`.
 - d. In the Available Integration Component Libraries frame select the check box corresponding to `ICL_SAPREPS`.
 - e. Click **Finish**.
 8. Deploy the prebuilt components present in `ICL_SAPREPS_UserProj`:
 - a. Ensure that the server is running.
 - b. In the InterChange Server Component Management window expand the InterChange Server Instances and select your server.
 - c. Right click and select **Connect**.
 - d. Select **UserProject ⇒ InterChange Server Project ⇒ ICL_SAPREPS ⇒ UserProj**. Right click **UserProj** and select **Deploy User Project**.
 - e. In the Deploy wizard select the **Destination Server** from the drop down list.
 - f. Select the check box corresponding to `ICL_SAPREPS_UserProj`.
 - g. Click **Finish**.
 9. In the InterChange Server Components window do the following:
 - a. From the InterChange Server Instances tree select the server on which you deployed the assets.
 - b. Expand the server and right click the **Collaboration Template**.
 - c. Invoke the **Compile All**. This displays the "Templates compiled successfully" message in the Console.
 10. Restart the server.

Generating SAP application specific business objects

The SAP application specific business objects must be generated from the user's SAP system. Use the SAP Object Discovery Agent (ODA) to generate the SAP application specific business objects. For more information, refer to the SAP Connector documentation. Use 'sap' as the prefix when specifying the property for the SAP ODA.

Generate the SAP application specific business objects from the IDocs listed in the following table:

Table 8. SAP application specific business objects and corresponding IDocs

SAP application specific business object	SAP IDoc
<code>sap_debmas05</code>	<code>DEBMAS_05</code>
<code>sap_cond_a02</code>	<code>COND_A02</code>
<code>sap_invcon01</code>	<code>INVCON01</code>
<code>sap_orders05</code>	<code>ORDERS05</code>

Table 8. SAP application specific business objects and corresponding IDocs (continued)

SAP application specific business object	SAP IDoc
sap_ciscso01	CISCO01
sap_sisdel01	SISDEL01
sap_sisinv01	SISINV01

See “Appendix C. Application specific business objects” on page 41 for the names of the generated SAP application specific business objects.

When the business objects for MATERIAL_AVAILABILITY BAPI is generated as described in point 4 of the “SAP modules” on page 29 section, the ODA will generate a data handler. Ensure that the class and Java™ (source) files are generated in the *WICS_installdir\connectors\SAP\bapi\client* directory.

In the SAP ODA wizard, there is option to save the generated application specific business object as an XSD file. Follow the on screen instruction to save the SAP application specific business objects in your file system. The instructions to load the saved SAP application specific business objects in the WebSphere InterChange Server system manager are described in “Loading SAP application specific business objects” on page 25.

Loading and deploying the assets

The WebSphere InterChange Server assets are provided in two packages to allow the separation of SAP specific assets from the WebSphere Commerce specific assets and generic assets. The package contains the following:

1. The WBI assets specific to WebSphere Commerce and the generic assets. This includes
 - a. The WebSphere Commerce specific business objects.
 - b. The maps to convert WebSphere Commerce application specific business objects to generic business objects and vice-versa.
 - c. The collaboration objects.
2. The WBI assets specific to SAP, which includes the maps to convert SAP application specific business objects to generic business objects and vice-versa.

Loading the WBI assets specific to WebSphere Commerce: From the WC56SAPWBIRefApp.zip file extract the contents of WC_Related_Assets.zip. into *WICS_installdir\my_WC_extract*. This creates two folders called Users and System that contain the WebSphere Commerce specific components. Load the components on to the WebSphere Studio Application Developer Workbench using the System Manager perspective.

1. Open the System Manager perspective in the WebSphere Studio workbench.
2. Expand **User Projects**, select **InterChange Server Project** and right-click **Import Solution**.
3. From the Import Solution window, browse and select *WICS_installdir\my_WC_extract* directory. This creates the following assets:
 - a. In the InterChange Server Project folder:
 - SAPRefApp_WC56_UserProj
 - b. In the Integration Component Library:
 - SAPRefApp_WC56

4. Import the generic business object components from the WebSphere InterChange Server instance to the SAPRefApp_WC56 folder.
 - a. Right-click **SAPRefApp_WC56** and select **Import component from server** ⇒ **BusinessObjects**.
 - b. The Import Component window opens. Expand the tree corresponding to your server.
 - c. Expand the Business Object tree. The list of generic business objects deployed in “Setting up the WebSphere InterChange server environment” on page 21 displays.
 - d. Select the following business objects:
ATP, ATPLine, Customer, CustomerPartner, InventoryLevel, InventoryLocation, Item, OrderBillingStatus, OrderDeliveryStatus, OrderStatus, PriceRecord, Order, Contact, MO_Server_DataHandler, and MO_DataHandler_Default

Note: Ensure that you select the **Deep** option check box from the Import Component window when importing the generic business objects.
5. Import the collaboration template components from the WebSphere InterChange Server instance to the SAPRefApp_WC56 folder.
 - a. Right-click **SAPRefApp_WC56** and select **Import component from server** ⇒ **Collaboration Templates**.
 - b. The Import Component window opens. Expand the tree corresponding to your server.
 - c. Expand the Collaboration Templates tree. The list of collaboration templates deployed in “Setting up the WebSphere InterChange server environment” on page 21 displays.
 - d. Select the following collaboration templates:
ATP, CustomerSync, InventoryLevelManager, OrderBillingStatus, OrderDeliveryStatus, OrderStatus, PriceManager, and SalesOrderProcessing
6. Import the connector components from the WebSphere InterChange Server instance to the SAPRefApp_WC56 folder.
 - a. Right-click SAPRefApp_WC56 and select **Import component from server** ⇒ **Connector**.
 - b. The Import Component window opens. Expand the tree corresponding to your server.
 - c. Expand the Connector tree. The list of connectors deployed in “Setting up the WebSphere InterChange server environment” on page 21 displays.
 - d. Select the WebSphere Commerce, SAP, and Port connectors.

Loading SAP application specific business objects: Loading SAP application specific business objects involves the following:

1. Open the System Manager perspective in the WebSphere Studio workbench.
2. From the System Manager menu, select **Tools**⇒**BusinessObject Designer**.
3. The BusinessObject Designer opens and displays the New Business Object dialog. Click **Cancel**.
4. From Business Object Designer menu select **File**⇒ **Open From File**. The Import Business Object(s) from File(s) dialog displays.
5. Select **SAPRefApp_WC56** from the To Project drop down list.
6. Open the **sap_debmas05.xsd** that was generated and stored in the file system. See, “Generating SAP application specific business objects” on page 23. Verify if the sap_debmas05 business object is loaded in the System Manager successfully.

7. Repeat steps 1 to 6 to load the following SAP application specific business objects:
 - sap_cond_a02.xsd
 - sap_invcon01.xsd
 - sap_orders05.xsd
 - sap_cisco01.xsd
 - sap_sisdel01.xsd
 - sap_sisinv01.xsd

Loading the WBI assets specific to SAP: From the WC56SAPWBIRefApp.zip file extract the contents of SAP_Related_Assets.zip. into *WICS_install\my_SAP_extract*. This creates two folders called Users and System that contain the SAP specific components. Load the components on to the WebSphere Studio Application Developer Workbench using the System Manager perspective.

1. Open the System Manager perspective in the WebSphere Studio workbench.
2. Expand **User Projects**, select **InterChange Server Project** and right-click **Import Solution**.
3. From the Import Solution window, browse and select *WICS_install\my_SAP_extract* directory. This creates the following assets:
 - a. In the InterChange Server Project folder:
 - SAPRefApp_WC56_UserProj
 - b. In the Integration Component Library:
 - SAPRefApp_WC56

This completes loading the WBI assets specific to WebSphere Commerce and SAP.

Compiling and deploying the assets: Do the following to compile and deploy the WBI assets specific to WebSphere Commerce and SAP:

1. Compile the maps
 - a. Expand the Integration Component Libraries.
 - b. Select **SAPRefApp_WC56⇒Maps**.
 - c. Right-click the **Maps** folder and select **Compile All**. This compiles all the maps required for this integration.
2. Specify the Supported Business Objects and Associated maps properties for the adapter for WebSphere Commerce. See “Configuring the WebSphere Business Integration Adapter for WebSphere Commerce” on page 28.
3. Specify the details for the Standard and Connector specific properties. Refer to the *Guide to the IBM WebSphere Business Integration Adapter for WebSphere Commerce*.
4. Specify the Supported Business Objects and Associated maps properties for the adapter for mySAP.com. See “Configuring the WebSphere Business Integration Adapter for mySAP.com” on page 29.
5. Specify the details for the Standard and Connector specific properties. Refer to the *Guide to the WebSphere Business Integration Adapters for mySAP.com (SAP R/3 V.4.x)*.
6. For the port connector, you need to specify only the Supported Business Objects. See “Appendix B. Generic business objects” on page 39 for the list of generic business objects to be specified for the port connector.

7. To check if all the ports in the collaboration object are bound correctly to the appropriate connectors, see “Associating the collaboration objects” on page 28.
8. Configure the meta-object (MO_WCSConfig). See “Configuring MO_WCSConfig” on page 28.
9. Update the SAPRefApp_WC56_UserProj:
 - a. Select the **SAPRefApp_WC56_UserProj** under **User Projects⇒Interchange Server Projects**.
 - b. Right click and select **Update Project**.
 - c. Select the **SAPRefApp_WC56** check box.
 - d. Click **Finish**.
 - e. A message displays stating that the yyyy Object already exists and would you like to replace it? Click **Yes to ALL**.
10. Deploy the delivered assets:
 - a. Ensure that the server is running. In the InterChange Server Component Management window expand the **InterChange Server Instances** and select your server.
 - b. Right click and select **Connect**.
 - c. Select **UserProject⇒InterChange Server Project⇒SAPRefApp_WC56_UserProj**.
 - d. Right click **SAPRefApp_WC56_UserProj** and select **Deploy User Project**.
 - e. In the Deploy wizard select the Destination Server from the drop down list.
 - f. Deploy the business objects:
 - 1) Expand **SAPRefApp_WC56_UserProj** and select the check box corresponding to Business Object.
 - 2) Click **Finish**.
 - g. Deploy the maps:
 - 1) Complete steps d and e.
 - 2) Expand **SAPRefApp_WC56_UserProj** and select the check box corresponding to Maps.
 - 3) Click **Finish**.
 - h. Deploy the connectors:
 - 1) Complete steps d and e.
 - 2) Expand **SAPRefApp_WC56_UserProj** and select the check box corresponding to Connectors.
 - 3) Click **Finish**.
 - i. Deploy the collaboration objects:
 - 1) Complete steps d and e.
 - 2) Expand **SAPRefApp_WC56_UserProj** and select the check box corresponding to Collaboration objects.
 - 3) Click **Finish**.
11. In the InterChange Server Components window, do the following:
 - a. From the InterChange Server Instances tree select the server on which you deployed the assets.
 - b. Expand the server and right click the **Maps**.
 - c. Invoke the **Compile All** option. This displays the Maps compiled successfully message in the System Manager Console.

Configuring MO_WCSConfig

The WebSphere Commerce connector static meta object consists of a list of conversion properties defined for all the specific business objects that WebSphere Commerce supports. For this reference application, inbound messages to WebSphere Commerce from SAP require the *ConfigurationMetaObject* meta object to be configured to suit your environment. The value of *ConfigurationMetaObject* is MO_WCSConfig. This involves editing the application specific information of the inbound messages to WebSphere Commerce.

Open the meta object in the business object designer tool and update the application specific information with the queue manager and queue name to suit your setup. Deploy the meta object to the WebSphere InterChange Server instance for the changes to take effect. For more information refer to the *Guide to the IBM WebSphere Business Integration Adapter for WebSphere Commerce*.

Associating the collaboration objects

The collaboration objects associated with the sample messages are provided with this reference application. For the list of collaboration objects and the corresponding port binding information, see “Appendix D. Collaboration objects and ports” on page 43.

Port connector

The collaboration ports that are not used in a collaboration object to support a message are bound to the port connector. For details, see “Appendix D. Collaboration objects and ports” on page 43.

Configuring the WebSphere Business Integration Adapter for WebSphere Commerce

The WBI adapter for WebSphere Commerce allows the WebSphere InterChange Server collaborations to exchange business objects with the WebSphere Commerce Server, which sends and receives messages over its messaging system using WebSphere MQ.

Configuring the adapter for WebSphere Commerce involves setting the values for the generic and connector specific configuration properties. Do the following:

1. Open the WebSphere Commerce connector using the connector configurator.
 - a. Specify the WebSphere Commerce specific business objects in the Supported Business Objects tab.
 - b. Select the **Agent supported** check box for the WebSphere Commerce application specific business objects.
 - c. Specify the related generic business objects. Clear the **Agent supported** check box for generic business objects.

For more details about the supported objects, see “Appendix B. Generic business objects” on page 39 and “Appendix C. Application specific business objects” on page 41 respectively.

2. In the connector specific properties set the value of the *ConfigurationMetaObject* to MO_WCSConfig.
3. Specify the associated maps with Explicit Binding for the supported business objects. See, “Appendix F. Binding maps to business objects” on page 47 to identify the business objects that require maps to be specified.
4. Set the BO prefix:

The application specific business objects for WebSphere Commerce are generated with the 'WCS' prefix. Ensure that 'WCS' is set for the BOPrefix property of the MO_DataHandler_Default object. Use the Business Object Designer tool to navigate to the following object hierarchy:

- MO_DataHandler_Default—The parent business object
- Text.xml—The child object of MO_DataHandler_Default
- BOPrefix—Property name (Set WCS as the default value for this property)

For more information on configuring the standard and connector properties, refer to the *The IBM WebSphere Business Integration Adapter Guide for WebSphere Commerce*.

5. Deploy the adapter for WebSphere Commerce and the business objects for the changes to take effect. For more information on editing the adapter properties and deploying the adapter, refer to the *Implementation Guide for WebSphere InterChange Server*.

Configuring the WebSphere Business Integration Adapter for mySAP.com

The WBI connector for SAP R/3 enables the WebSphere InterChange server to exchange business objects with SAP applications. The WebSphere InterChange server SAP business objects are defined in SAP as IDocs. IDocs are part of the Application Link Enabling (ALE) module. The IDoc definitions are stored in the Business Object Repository in SAP. WebSphere InterChange server provides an IDoc Handler that supports business objects developed using IDocs.

To install the adapter for mySAP.com agent, refer to the *WebSphere InterChange server Installation Guide for Windows*. The WebSphere Business Integration Adapter for mySAP.com components can be found in the \connectors\SAP directory.

SAP modules

The three SAP modules used in this reference application include:

- ALE module
- BAPI module
- RFCServer module

To set values to the configuration properties of the WebSphere Business Integration adapter for mySAP.com refer to the *Guide to the WebSphere Business Integration Adapters for mySAP.com (SAP R/3 V.4.x)* in the WebSphere InterChange server documentation.

Configuring the adapter for mySAP.com involves setting the values for the generic and agent specific configuration properties. You must provide the following details:

1. Specify the list of generic and application specific business objects that the adapter for mySAP.com must support. For more information see, "Appendix B. Generic business objects" on page 39 and "Appendix B. Generic business objects" on page 39 respectively.
2. Specify the associated map with the 'Explicit Binding' option selected for the supported business object. See, "Appendix F. Binding maps to business objects" on page 47 to identify the business objects that require maps for conversion.
3. To set up the Transaction ID (TID) management, refer to the *Guide to the WebSphere Business Integration Adapters for mySAP.com (SAP R/3 V.4.x)*. The corresponding objects are provided as part of the WebSphere Business Integration Adapter for mySAP.com installation.

4. To support the BAPI module, the adapter for mySAP.com requires the specific runtime library. This runtime library processes the BAPI for information exchange between SAP and the external system. Do the following to generate the BAPI specific business objects and its corresponding runtime library:
 - a. Configure SAP ODA to connect to the desired SAP system. Refer to the *Guide to the WebSphere Business Integration Adapters for mySAP.com (SAP R/3 V.4.x)* for detailed instructions.
 - b. When using the ODA, expand RFC and select the required BAPI, in this case it is the Material Availability BAPI. Click Next.
 - c. Confirm that you have selected the correct BAPI and click Next.
 - d. Follow the subsequent instructions to generate the business object and the runtime library. The generated runtime library (.java and .class) are copied to the *WICS_installdir\connectors\SAP\bapi\client*.
 - e. Manually load the generated business object in the System Manager for deployment in the WebSphere InterChange Server.

Configuring the WebSphere InterChange server queue manager

This reference application requires you to configure a WebSphere MQ Queue Manager and a WebSphere MQ Listener at the WebSphere InterChange server side. Refer to the WebSphere InterChange server documentation for more information.

Enabling asynchronous communication

This section details the steps required to enable WebSphere MQ to communicate between WebSphere Commerce and WebSphere InterChange server.

Configuring the WebSphere Commerce MQ

Channels are used to communicate between two WebSphere MQ systems. Refer to the WebSphere MQ documentation to create the sender and receiver channels.

Create the sender and receiver channels in the WebSphere InterChange server and the WebSphere Commerce systems using WebSphere MQ Explorer.

The name of the sender channel in WebSphere InterChange server must be identical to the name of the receiver channel in WebSphere Commerce. The name of the receiver channel in WebSphere InterChange server must be identical to the name of the sender channel in WebSphere Commerce.

After you finish configuring the WebSphere Commerce and WebSphere InterChange server WebSphere MQ start the receiver channel and then the sender channel. The status must be "running".

Enabling the message flow

To enable a synchronous or an asynchronous message flow, do the following:

1. To enable and use the CheckInventoryAvailabilityBE synchronous message using the adapter for WebSphere InterChange Server, refer to the *WebSphere Commerce Additional Software Guide*.
2. Ensure that the corresponding application specific business objects (for WebSphere Commerce and SAP) and maps are loaded into the WebSphere InterChange server repository. For details, see "Appendix D. Collaboration objects and ports" on page 43.

3. Open the Properties window for the WebSphere Commerce connector. Specify the WebSphere Commerce specific business objects in the Supported Business Objects tab. Select the Agent supported check box for the objects. Specify the related generic business objects. Clear the Agent supported check box. For more details about the supported objects, see “Appendix B. Generic business objects” on page 39. Repeat this step for the adapter for mySAP.com with the SAP specific business objects.
4. Ensure that the collaboration objects are configured and bound with the appropriate connectors. See, “Appendix D. Collaboration objects and ports” on page 43 for details.
5. Ensure that the business objects and solution specific maps are explicitly bound for the connectors. For details see, “Appendix F. Binding maps to business objects” on page 47.
6. Ensure that the collaborations, connectors, and maps are running.
7. Start the WebSphere Commerce and WebSphere Business Integration adapter for mySAP.com agents. Initiate the message flow.

You have now completed the following:

- Installing and configuring WebSphere MQ
- Configuring the WebSphere Commerce messaging system
- Publishing the WICSSAPAdvancedB2BDirect store
- Installing and configuring WebSphere InterChange server
- Configuring the WebSphere InterChange server queue manager
- Configuring the WebSphere Business Integration Adapter for WebSphere Commerce
- Configuring the WebSphere Business Integration Adapter for mySAP.com
- Enabling the message flow

WebSphere Commerce is now enabled to communicate with WebSphere InterChange server. The WebSphere InterChange server is configured to connect to your SAP R/3 backend system. You have loaded the business objects, maps and collaboration objects, published the WICSSAPAdvancedB2BDirect store, and loaded the master data. To test the synchronous and asynchronous message flows see, Chapter 5, “Verification procedure,” on page 33.

Chapter 5. Verification procedure

This chapter provides the procedural details for ensuring the end-to-end flow for each message. Before you begin verification, confirm that the WebSphere MQ queue manager and the channels are running and are in the active stage. Ensure that the adapters for WebSphere Commerce and mySAP.com agents are running.

Any errors can be detected in the InterChange system log, adapter for mySAP.com and WebSphere Commerce connector agent logs or, WebSphere Commerce Business Edition logs respectively. In WebSphere Commerce, the logging for the messaging component must be enabled.

Verifying the messages

To verify the messages do the following:

Customer create or update message (DEBMAS05)

To create or update customer information in SAP and sending the inbound message to WebSphere Commerce, do the following:

1. Generate a DEBMAS05 IDOC message using one of the following options:
 - Create a new customer using the SAP transaction xd01.
 - Change the details of an existing customer using the SAP transaction xd02.
2. Enter the details for the customer and save the details.
3. Execute SAP transaction se38 to generate the IDoc. Select the program RBDMIDOC and run the program.
4. On the next screen type DEBMAS as the message and run the program. This creates the IDoc but does not dispatch it.
5. To send the IDoc, go to SAP transaction se38 and run the RBDOUTPU program.
6. Select **dispatch** on the next page and run the program.
7. Type DEBMAS05 as the basic type on the next page and run the program. This sends the DEBMAS05 IDoc.
8. To verify whether the IDoc message is sent out of the SAP system, run transaction we02 and check whether the DEBMAS message generated is listed in the outbound list.
9. Log into the WICSSAPAdvancedB2BDirect store. Select **Account** and click **Change Personal Information**. Note the change in the address. You can also verify the corresponding values in the USERREG, ADDRESS and USERS tables.

CheckInventoryAvailabilityBE request or response message

To check the available inventory for the products in SAP do the following:

1. Log into the WebSphere Commerce store as a customer, using a valid user ID. You can use the logon ID of the customer created in SAP.
2. Add products to the shopping cart.
3. Select the check box displayed against the products in the shopping cart.
4. Click the **Check Inventory availability** button.

5. The page refreshes with the details of available quantity and available dates for the products selected, by sending a request to SAP and updating the results in WebSphere Commerce.

Order create message (ORDERS05)

To create an order in WebSphere Commerce and check whether the order is created and processed correctly in the SAP system, do the following:

1. If the customer belongs to a buyer organization, then the role of Registered Customer must be assigned before logging into the store.
2. Ensure that the customer, product, pricing, and inventory data in WebSphere Commerce is consistent with that in the SAP system.
3. Log into the WebSphere Commerce store as a customer, using a valid user ID. You must use the logon ID of the customer created in SAP.
4. Add products to the shopping cart and submit an order. This generates the XML order create message. It may take some time to deliver the message, as determined by the scheduler configuration in WebSphere Commerce. By default, the scheduler process for sending messages into WebSphere MQ runs at an interval of five minutes. Note down the order number created in the order Confirmation page.
5. To check whether the order create XML was successfully parsed, formatted, and sent to the SAP system, run the SAP transaction we02. The ORDERS05 IDoc message must be listed in the inbound section.
6. Note the status under which ORDERS05 displays. It must be under status 53, which means order created successfully.

Order confirmation status message (SISCSO01)

To verify if the order confirmation status message is generated and processed correctly do the following:

1. In response to an order created successfully in SAP, SAP automatically sends the order confirmation status message. The IDoc message for this is SISCSO01. The generation of this IDoc message can be verified using SAP transaction we02.
2. Log into the WICSSAPAdvancedB2BDirect store. Select **Order Status** and look for the order in the Orders Confirmed section. You can also verify that the corresponding order in WebSphere Commerce is set to G in the ORDERS table. The respective entries are made in the ORDSTAT and ORDISTAT tables.

Order delivery status message (SISDEL01)

To generate the order delivery status message for the confirmed order do the following in the SAP system:

1. To create the SISDEL01 IDoc, run SAP transaction va02.
2. Type the order number that SAP generates. This can be obtained from the SISCSO01 IDoc that was generated previously; alternatively you can enter the purchase order number or customer number to search for the orders created.
3. When the Order Details page displays, select **Sales Document - Deliver** from the menu.
4. This process creates the delivery message and sends the IDoc message, unless an error occurs. The SAP client in the error log shows the errors.

5. Log into the WICSSAPAdvancedB2BDirect store. Select **Order Status** and look for the order in the Orders Shipped section. You can also verify the ORDSTAT and ORDISTAT tables in WebSphere Commerce. The status of the corresponding order items must be S.

Note: WebSphere Commerce allows you to version the order status messages. Depending on the option selected, either the existing status record will be updated or a new record will be added to these tables. By default, the order status header and the order status item are not version controlled.

Product inventory update message (INVCON01)

To verify if the product inventory update message is generated and processed correctly for a single product, do the following:

1. To generate INVCON01 IDoc, run transaction mb1c.
2. Type the movement type, plant, and storage location details and press F8.
3. Enter the product SKU number and the quantity. This message supports the inventory update of a single product only.
4. To verify in WebSphere Commerce, check the inventory of the ordered material in the INVENTORY table.

Order invoice status message (SISINV01)

To verify if the order invoice status message is generated and processed correctly, do the following:

1. Run the transaction vf01. A page displays asking for the billing type.
2. Select Invoice (F1) from the menu. This automatically retrieves the document number for the delivery created previously.
3. If the document number does not display, then select the document number from the menu. This creates the SISINV01 IDoc.
4. Login to the WICSSAPAdvancedB2BDirect store. Select **Order Status** and look for the order in the Orders Invoiced section. You can also verify the ORDSTAT and ORDISTAT tables in WebSphere Commerce. The status of the corresponding order should be set to 'I'.

Note: WebSphere Commerce allows you to version the order status messages. Depending on the option selected, either the existing status record will be updated or a new record will be added to these tables. By default, the order status header and the order status item are not version controlled.

Product price update message (COND_A02)

To verify if the product price update message is generated and processed correctly do the following:

1. To generate the product price update message, run SAP transaction vk12. When prompted for the condition type, enter the appropriate pricing condition you have created.
2. When the transaction runs, you are asked for the key combination. Based on your preference, select the option. Enter the required fields on the next screen, and execute (F8).
3. In the next screen, change the rate for a WCBE-known material and save the document.
4. Run SAP transaction se38 to generate the IDocs.
5. Type RBDMIDOC as the program name and execute (F8).

6. On the next page, type COND_A as the message type and run the program. This creates the IDoc but the IDoc will not be sent at this stage.
7. To send the IDoc, go to SAP transaction se38 and run the RBDOUTPU program.
8. Select **dispatch** on the next page and run the program.
9. On the next screen type COND_A02 as the basic type and run the program. This should dispatch the COND_A02 IDOC.
10. Log into the WICSSAPAdvancedB2BDirect store. Check the price of the order item after adding it to the shopping cart. You can also verify the change in price from the OFFERPRICE and OFFER tables in WebSphere Commerce.

Appendix A. Messages integrated for this reference application

This section describes the asynchronous and synchronous messages that this reference application enables. These messages are used to exchange information between the WebSphere Commerce and SAP systems. For example, order status, inventory update, and others.

Asynchronous messages

Asynchronous messages do not elicit an immediate response to a request. For example, when WebSphere Commerce executes the OrderCreate message it does not expect an immediate reply from the SAP system. As a result, the buyer is not updated immediately regarding the status of the order. This reference application supports the following asynchronous messages:

Order create message (outbound from WebSphere Commerce)

WebSphere Commerce Business Edition generates this message when an order is submitted in the commerce server.

Order status message (inbound to WebSphere Commerce)

The SAP system generates this message. There are three order status messages:

- Order confirm status: Generated when SAP confirms the order.
- Order delivery status: Generated when delivery for the order is complete at the SAP end.
- Order invoice status: Generated when the order is invoiced in SAP.

Customer new message (inbound to WebSphere Commerce)

SAP generates this message when a new customer is registered in SAP.

Customer update message (inbound to WebSphere Commerce)

SAP generates this message when an existing customer's information is updated in SAP.

Product price update message (inbound to WebSphere Commerce)

SAP generates this message when the product price is updated in SAP.

Product quantity update message (inbound to WebSphere Commerce)

SAP generates this message when the product quantity is updated in SAP. This could occur when the inventory is:

- Reduced at the time of goods issue for an order or
- Updated manually

Synchronous messages

A synchronous message elicits an immediate response to a request. For example, when WebSphere Commerce requests the SAP system to check the inventory availability, the SAP system replies immediately with details of the inventory check. As a result, the buyer gets a real-time response and can view the availability status before placing the order. This reference application supports the following synchronous message:

CheckInventoryAvailability message (outbound from WebSphere Commerce)

This message is used to check the available inventory of a product in SAP from WebSphere Commerce.

To enable and use the CheckInventoryAvailabilityBE message using the adapter for WebSphere InterChange Server, refer to the *WebSphere Commerce Additional Software Guide*.

Appendix B. Generic business objects

Ensure that the WebSphere Commerce connector, mySAP.com connector, and the Port connector are bound to the following generic business objects.

Note: Disable the agent support for all generic business objects

Table 9. Generic business objects

WebSphere Commerce connector	mySAP.com connector	Port connector
Customer	ATP	ATP
CustomerPartner	ATPLine	ATPLine
InventoryLevel	Customer	Contact
InventoryLocation	CustomerPartner	Customer
Item	InventoryLevel	Item
Order	InventoryLocation	Order
OrderBillingStatus	Item	
OrderDeliveryStatus	Order	
OrderStatus	OrderBillingStatus	
PriceRecord	OrderDeliveryStatus	

Note: The port connector is used to bind those ports of a collaboration that do not require a process flow through the application specific connectors.

Appendix C. Application specific business objects

Ensure that the WebSphere Commerce connector and adapter for mySAP.com are bound to the application specific business objects described in Table 10.

Note: Enable agent support for all application specific business objects.

Ensure that the list of SAP application specific business objects should have been generated and loaded in accordance with the instructions specified in “Generating SAP application specific business objects” on page 23 and “Loading SAP application specific business objects” on page 25 respectively.

Table 10. Application specific business objects

WebSphere Commerce connector	mySAP.com connector
MO_DataHandler_Default	MO_Server_DataHandler 1
MO_WCSConfig	sap_bapi_material_availability
WCS_Create_WCS_Customer	sap_cond_a02
WCS_Update_WCS_Customer	sap_debmas05
WCS_Report_NC_PurchaseOrder	sap_invcon01
WCS_Update_WCS_OrderStatus	sap_orders05
WCS_Update_WCS_ProductInventory	sap_siscso01
WCS_Update_WCS_ProductPrice	sap_sisdcl01
WCS_Request_WCS_BE_ProductInventory (Synchronous)	sap_sisinv01
	SAP_TransId

Appendix D. Collaboration objects and ports

The following table lists the messages, their collaboration templates, and ports:

Table 11. Collaboration objects and ports

Message	Collaboration object	Collaboration template name	From port	To port	Destination App Retrieve	Other ports
Inventory Check from SAP	Product Quantity Update_From_Sap	InventoryLevel Manager	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	ToItem Wrapper - SAP Connector
Product Price Update from SAP	ProductPrice Update_From_Sap	PriceManager	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	ToItem Wrapper - Port Connector
OrderBilling Status from SAP	OrderBilling Status_From_Sap	OrderBilling Status	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	
Order Delivery Status from SAP	Order Delivery Status_From_Sap	OrderDelivery Status	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	
Order Confirmation Status	Order Confirmation Status_From_Sap	OrderStatus	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	
Customer Create/ Update from SAP	Customer Create Update_From_Sap	CustomerSync	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	To Customer Partner Wrapper - Port Connector
Check Inventory Availability BE	SAP_Check Inventory Availability	ATP	External Connector Incoming map: GBO_ATP_From_WCS Outgoing map: GBO_ATP_TO_WCS	SAP Connector	SAP Connector	ToLine - SAP Connector Destination App Retrieve Line - SAP Connector
Order Create from WebSphere Commerce	OrderCreate_From_WCBE	SalesOrder Processing	WebSphere Commerce Connector	SAP Connector	SAP Connector	To Customer Wrapper - Port Connector ToContact Wrapper - Port Connector ToItem Wrapper - Port Connector
Product Quantity Update	Product Quantity Update_From_Sap.out	InventoryLevel Manager	SAP Connector	WebSphere Commerce Connector	WebSphere Commerce Connector	ToItem Wrapper - SAP Connector

Note: When using external connectors you must explicitly specify the incoming and outgoing maps.

Appendix E. Maps with specific values for WebSphere Commerce and SAP R/3

The following are WebSphere Commerce and SAP specific values in the maps provided with this reference application. You can change them according to your configuration settings.

Table 12. Maps with WebSphere Commerce and SAP specific values

Map name	Hard coded attributes
GBO_ATPLine_To_SAP	Unit_of_measure_for_display, Plant, Checking_rule
GBO_CustomerCreate_To_WCS	Value, AddressType, Type
GBO_CustomerUpdate_To_WCS	PreferredLanguage, CustomerStatus, value, AddressType, type
Sub_GBO_DeliveryStatusLine_To_WCS	Type
GBO_OrderCreate_To_SAP	Name_of_table_structure Partner_type_of_recipient Partner_number_of_recipient Partner_type_of_sender Sender_port Partner_number_of_sender SAP_Release_for_Idoc Name_of_basic_typeIdoc_type Logical_message_type EDI_message_type Partner_function_e_g_sold_to_partyQualifier_for_IDOC_date_segment IDOC_qualifier_reference_document Partner_number_of_sender
Sub_GBO_OrderCreate_To_SAP	Plant IDOC_object_identification_such_as_material_no_customer IDOC_qualifier_reference_document
GBO_ProductPriceUpdate_From_SAP	PriceListId (Edit the custom code and replace the hardcoded values within quotes to WebSphere Commerce specific values) For example: if your store owner (WebSphere Commerce) member_id is 7000000000000000002 then set (sapSalesOrg.equals("WSO1")) sapSalesOrg ="7000000000000000002";)
GBO_ProductPriceUpdate_To_WCS	Precedence Published
GBO_ProductQuantityUpdate_To_WCS	MerchantID FulfillmentCenterID
GBO_ProductQuantityUpdate_from_SAP	Plant InvLocationId

Note: Ensure that the values entered in the fields described previously are in accordance with your configuration settings.

Appendix F. Binding maps to business objects

The following tables list the maps that you must explicitly bind to the business objects for WebSphere Commerce connector and WebSphere Business Integration Adapter for mySAP.com:

WebSphere Commerce connector

Table 13. WebSphere Commerce connector business objects and maps

Business object	Map
OrderBillingStatus	GBO_BillingStatus_To_WCS
OrderStatus	GBO_ConfirmationStatus_To_WCS
WCS_Report_NC_PurchaseOrder	GBO_OrderCreate_From_WCS
PriceRecord	GBO_ProductPriceUpdate_To_WCS
Customer	GBO_CustomerCreateUpdate_polymap
OrderDeliveryStatus	GBO_DeliveryStatus_To_WCS
InventoryLevel	GBO_ProductQuantityUpdate_To_WCS

WebSphere Business Integration Adapter for mySAP.com

Table 14. WebSphere Business Integration Adapter for mySAP.com business objects and maps

Business object	Map
ATPLine	GBO_ATPLine_To_SAP
sap_sisco01	GBO_ConfirmationStatus_From_SAP
sap_sisdel01	GBO_DeliveryStatus_From_SAP
sap_sisinv01	GBO_BillingStatus_From_SAP
Order	GBO_OrderCreate_To_SAP
sap_debmas05	GBO_CustomerCreateUpdate_From_SAP
sap_invcon01	GBO_ProductQuantityUpdate_From_SAP
SAP_TransId	SAP_TransId_Mgmt
sap_bapi_material_availability	GBO_ATPLine_From_SAP
sap_cond_a02	GBO_ProdcutPriceUpdate_From_SAP

Appendix G. Store customization

The following changes are made to the business direct store to achieve the SAP integration functionality:

Reference data

The catalog related XMLs are modified to populate the reference data. The list of XMLs includes catalog.xml, en_US/catalog.xml, offering.xml, and others.

Inventory

Storefulfill.xml is modified to populate the inventory details for the reference catalog items.

Store Language

The store.xml is updated to support the English language only. Only en_US locale specific properties are provided for this reference application.

Address page

Modifications to the AddressForm.jsp file are made provide a selection box to select the country and state codes. This information is mandatory in SAP for customer registration.

Shoppingcart page

Modifications to the CurrentOrderDisplay.jsp file are made to provide selection of order items and a button to check inventory. Another column was added to show the estimated available dates. This column will be blank before the user clicks **Check Availability**. When the user clicks **Check Availability** this column displays the date that SAP provides.

Track Order Status

Modifications to the OrderStatusDisplay.jsp file allows you to retrieve the list of orders confirmed, shipped, or invoiced, based on details available in the ORDSTAT and ORDISTAT tables. The possible status values of the orders are: C (Confirmed), S (Shipped), and I (Invoiced). A link is provided for each of the orders in the list to view the detailed order status.

Order Status Details

Modification to OrderDetailDisplay.jsp file displays the detailed order status available in the order status tables.

Shipping Modes

The Shipping.xml file is modified to replace the shipping codes A1, A2, A3 with BYTRUCK, BYRAIL, BYAIR respectively, and shipping carrier 'XYZ Carrier' with CFR. en_US/Shipping.xml is modified to set the description for the preceding shipping modes to 'CFR-TRUCK', 'CFR-RAIL', and 'CFR-AIR' respectively. The changes to the Shipping.xml file are made to match the shipping modes that SAP provides. These shipping modes are used when placing the order using the default contract. If any other contracts are created, then you can use these shipping modes while defining terms and conditions, otherwise create new shipping modes before they are used in the terms and conditions of the contract. For more details, refer to the WebSphere Commerce information center.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the responsibility of the user to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information that has been exchanged, should contact:

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

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

This document may contain information about other companies' products, including references to such companies' Internet sites. IBM has no responsibility for the accuracy, completeness, or use of such information.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The IBM logo and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries or both:

IBM
CrossWorlds
DB2

WebSphere
DB2 Universal Database™
MQSeries

Microsoft®, Windows, and Windows NT® are trademarks or registered trademarks of Microsoft Corporation.

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

SAP and R/3 are trademarks or registered trademarks of SAP AG.

Other company, product, or service names may be trademarks or service marks of others.



Printed in USA