

IBM WebSphere Commerce Business Edition
IBM WebSphere Commerce Professional Edition



Migration Guide for OS/400

from WebSphere Commerce Suite 5.1

Version 5.5

IBM WebSphere Commerce Business Edition
IBM WebSphere Commerce Professional Edition



Migration Guide for OS/400

from WebSphere Commerce Suite 5.1

Version 5.5

Note:

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 167.

First Edition, Fourth Revision (December, 2004).

This edition applies to IBM WebSphere Commerce Business Edition Version 5.5 for @server iSeries, and to IBM WebSphere Commerce Professional Edition Version 5.5 for IBM @server iSeries, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality.

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 2001, 2003. All rights reserved.

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

Contents

About this book.	vii
Summary of changes	vii
What's new in WebSphere Commerce 5.5	viii
Who should perform the migration?	xii
Supported migration paths	xiii
Migrating from earlier versions	xiv
Migration paths by operating systems	xiv
Other considerations	xiv
Conventions used in this book	xv
Path variables	xv
Running database scripts	xvi
References to Oracle	xvi

Part 1. Required migration steps. . . . 1

Chapter 1. Before you migrate from Commerce Suite 5.1 3

Premigration actions.	3
Ensuring no new messages are stored in MSGSTORE table.	4
Closing your existing auctions	5
Preparing the staging server for migration	5
Checking the unique index in the CONTRACT table	5
Premigration considerations	6
Migrating to ATP inventory	6
Orders and order items.	7
Master catalog.	8
Default contract	10
Access control	10
Member subsystem.	11
Distinguished names	15
Transitioning to WebSphere Application Server 5.0	16

Chapter 2. Backing up Commerce Suite 5.1 17

Backing up your Commerce Suite 5.1 system	17
Backing up directories and files.	17
Backing up your databases	19

Chapter 3. Upgrading your software 21

Mapping the Commerce Suite 5.1 to WebSphere Commerce 5.5 IBM software.	21
Approach to upgrading the software	21
Installing WebSphere Commerce 5.5 on a separate machine (remote migration)	22
Additional machine requirements for transition	23
Creating a WebSphere Commerce test instance	23
Migrating your previous WebSphere Application Server configuration	24
Next Step	25
Upgrading your existing system to WebSphere Commerce 5.5 (in-place migration)	25
Upgrading your hardware	25

Upgrading your operating system	25
Upgrading additional software components	25
Upgrading to WebSphere Application Server 5.0	26
Upgrading to WebSphere Commerce 5.5.	27
Next Step	27

Chapter 4. Migrating your Commerce instance configuration 29

Copying the Commerce Suite 5.1 product.xml file	29
Updating system values in the WebSphere Commerce 5.5 product.xml file	29
Checking the WebSphere Commerce 5.5 product.xml file (in-place migration only)	30
Steps before migrating your instance configuration	30
Migrating your instance configuration using WCIM	31
Prerequisites for running WCIM	32
Migrating a Commerce Suite 5.1 instance	33
After running the WCIM script.	43
Removing obsolete class paths in WebSphere Application Server	43
Adjusting JVM properties in WebSphere Application Server	44
Running the wcmWasConfig.jacl script	44
Checking the JDBC Provider.	46
Deploying the migrated .ear file	47
Selecting the WebSphere Application Server pass by reference setting.	48
Creating JNDI names for JCA connectors	49
Updating file and directory permissions.	50
Transitioning and deploying your custom code	50
Updating deployed instance XML files	50
Customizations in httpd.conf	50
Reconfiguring Web servers	50
Enabling WebSphere Application Server EJB Security	51
Copying files with static content to the documentation root directory	51

Chapter 5. Before migrating your database 53

Ensuring database journal receivers are large enough.	53
Updating database system views to the V5R2 level	54
Considering the order of columns	54
Checking the unique indexes in the MSGTYPES table.	55
Running the database preparation script.	55
DB2 databases	56
Dropping custom constraints	57

Chapter 6. Migrating your Commerce Suite 5.1 database. 59

Migrating your database	59
DB2 databases	60

Restoring custom constraints	61
DB2 Databases	61
Updating distinguished names	62
DB2 databases	62
Removing case sensitive search.	63
Assigning a master catalog	63
DB2 Databases	64
Changing the status of orders and order items.	64
DB2 Databases	64
Running the database consistency checker	65
DB2 databases	65
Remigrating your database	65

Chapter 7. Post instance and database migration. 67

iSeries Apache HTTP Server	67
Migrating your security configuration	67
Java virtual machine heap size consideration	68
Confirm encryption setting	68
Running the MigrateEncryptedInfo utility	68
Migrating store assets	70
Updating the Commerce Suite 5.1 JSP files	70
+ Updating the Commerce Suite 5.1 JSP files	73
Tools XML Files Migration	74
Updating custom code for discount, shipping, or taxation	74
Migrating cache policies	77
Enabling WebSphere Application Server dynamic cache	77
Enabling the dynamic cache service and servlet caching.	78
Web server plug-in consideration	78
Restarting the Web server and WebSphere Application Server	78
Changing your host name(remote migration only)	79
Verifying successful instance and database migration	79

Chapter 8. Migrating to Commerce Payments 81

Payments migration overview	81
Components migrated	81
Transition methodology	81
Third-party cassette considerations (custom code)	82
Premigration considerations	82
Migrating your Payments instance and database	83
Migrating your Payments instance using WCIM	84
Updating the Web server ports	88
Migrating your Payments database	89
Post instance and database migration steps.	90
Changing to WCSRealm (recommended)	90
Continuing to use PSOS400Realm (optional)	91
Starting the Payments instance	91
Payments migration usage scenarios	91
Migrating the previous Payment Manager in-place.	91
Migrating the previous Payment Manager on a remote machine	92
Multiple WebSphere Commerce instances pointing to one instance of Payments.	93
Payments subsystem migration considerations.	94

What the data migration does	95
Additional Payments subsystem migration considerations	96
Migrating from PAYMTHD table to payment policies.	96
Business policies and business policy commands	97
Migrating to WebSphere Commerce 5.5 business policy commands for payment	99
Changing CMDREG entry for DoCancelCmd	100
Sample JavaServer Pages file - PayStatusPM.jsp	100

Part 2. Additional migration steps 101

Chapter 9. Member subsystem migration 103

Overview of migration procedures	103
Existing Commerce Suite 5.1 user with existing directory server	105
Continue using the Commerce Suite 5.1 directory server in WebSphere Commerce 5.5	105

Chapter 10. Access control subsystem considerations. 107

Migrating the ACCCUSTEXC table	107
--	-----

Chapter 11. Migrating other WebSphere Commerce components . 109

Configuring user roles	109
Migrating the staging server	110
Reconfiguring the database cleanup utility.	110
Migrating the rule server configuration.	110
Auctions	112
Business accounts and contracts	113
Shipping Calculation Codes	113
Format for logonID in the edit_registration page (when LDAP is used).	113
Product Advisor migration	114
ATP inventory rounding.	115
Default currency behavior	115
Access control policy subscriptions	116

Part 3. Appendixes 117

Appendix A. Where to find more information 119

WebSphere Commerce information	119
WebSphere Commerce online help	119
WebSphere Commerce technical library.	119
WebSphere Commerce Payments information.	119
IBM HTTP Server information.	121
WebSphere Application Server information	121
DB2 Universal Database information	121
Other IBM publications	121

Appendix B. Renaming your Payments instance 123

Appendix C. Restoring a Payments library under a different name when the Payments instance has been moved to a new system	125
--	------------

Appendix D. Extending the data migration script	127
--	------------

Appendix E. Overview of the WCIM tool and the data migration script . . .	129
--	------------

Instance migration using WCIM	129
Data migration script.	130
Member migration	130
Catalog migration	131
ATP inventory migration	132
Order items migration	132
Shipping calculation codes	132
Discount data migration.	132
Contract migration	133
Campaign migration	134
Access Control migration	134
Supplemental information for data migration.	135

Database preparation script return codes	135
Ensuring column lengths are not exceeded	140
Considerations for ATP migration	140
Checking the database migration log files	141
Database consistency checker output	144

Appendix F. Converting to ATP inventory at a later time	147
--	------------

For DB2 databases.	147
----------------------------	-----

Appendix G. Updates to sample JSPs	149
---	------------

register.jsp	150
account.jsp	155
infashiontext_en_US.properties	159

Appendix H. Troubleshooting	161
--	------------

Enabling logging and tracing	162
Logging	163
Tracing	164

Notices	167
--------------------------	------------

Trademarks	169
----------------------	-----

About this book

This document describes the steps to migrate IBM® WebSphere® Commerce Suite 5.1 to IBM WebSphere Commerce 5.5 on IBM @server® iSeries™. If you wish to migrate your previous version of WebSphere Commerce Studio to WebSphere Commerce Studio 5.5, see the *WebSphere Commerce Studio Migration Guide*.

This document collectively refers to the previous release plus any associated fix packs that have been applied to that previous release, as “Commerce Suite 5.1”. This document refers to the current release, WebSphere Commerce 5.5, as “WebSphere Commerce 5.5” or “WebSphere Commerce”.

Summary of changes

This Migration Guide, and any updated versions of this Migration Guide, will be available in the WebSphere Commerce Technical Library Web page (<http://www.ibm.com/software/genservers/commerce/library/>). For additional information for your WebSphere Commerce edition, see the overview pages:

- Business Edition
(<http://www.ibm.com/software/genservers/commerce/wcbe/>)
- Professional Edition
(<http://www.ibm.com/software/genservers/commerce/wcpe/>)

For additional support information, see the WebSphere Commerce Support site (<http://www.ibm.com/software/genservers/commerce/support/>).

To learn about last-minute changes to the product, see the updated product README file, also available from the above Web site. For instructions on how to install WebSphere Commerce 5.5 and its supported products, see the *WebSphere Commerce Installation Guide*.

Any updates to this book will be summarized in this section.

Updates from the last version of this book are identified by revision characters contained in the margin. This book uses the following conventions for revision characters:

- The “+” (plus) character identifies updates that have been made in the current version of this book.
- The “|” (split vertical bar) character identifies updates that have been made in the previous versions of this book.

The following table shows the main changes that have been made to this book.

Change	Chapter, section, or page affected
Added in information on selecting the WebSphere Application Server pass by reference setting after instance migration.	“Selecting the WebSphere Application Server pass by reference setting” on page 48
Added in information on creating JNDI names for JCA connectors.	“Creating JNDI names for JCA connectors” on page 49
Added setenv.sh file requirement for WCIM script when migrating Payments.	“Migrating your Payments instance using WCIM” on page 84

Change	Chapter, section, or page affected
Corrected steps to check the JDBC provider in the WebSphere Application Server Administration Console.	"Checking the JDBC Provider" on page 46
Added several steps to the procedure on changing your host name to the final host name after migration.	"Changing your host name(remote migration only)" on page 79
Added steps to enable the "Enable servlet caching" flag.	"Enabling the dynamic cache service and servlet caching" on page 78
Clarify steps for deploying ear file.	"Deploying the migrated .ear file" on page 47
Corrected directory path.	"Configuring user roles" on page 109
Added troubleshooting item due to WebSphere Application Server behavior change.	Appendix H, "Troubleshooting," on page 161
Added item on safely ignoring an error message for MigrateEncryptedInfo script.	"Migrating your security configuration" on page 67

What's new in WebSphere Commerce 5.5

Important enhancements and new features relative to Commerce Suite 5.1 have been introduced in WebSphere Commerce 5.5. For more detailed information on the new features and enhancements provided with WebSphere Commerce 5.5, see the *What's New in WebSphere Commerce for Version 5.5* document. The following list is a summary of the enhancements:

- Access control policies — Updates to access control policies, including organization-based authorization policy and policy group subscription.
- Advanced discounts and promotions — New capabilities include a more flexible and easily customizable framework for both simple and complex discount and promotion models.
- Analytics and Business Intelligence — WebSphere Commerce Analyzer, Advanced Edition is now packaged with WebSphere Commerce, to provide a more sophisticated data extraction.
- Attachment support — Support introduced for a variety of file attachment formats.
- Business models — Includes Consumer direct (business-to-consumer), Business direct (business-to-business), and Business indirect which enables business-to-business transactions involving multiple enterprises.
- Catalog manager — No longer includes the Web Editor. Use the Product Management tooling in WebSphere Commerce 5.5.
- Channel management — The new Channel Management feature adds several business direct and business indirect models to support the relationships between various trading parties, such as Resellers, Distributors, Manufacturers, and Buyers, and the interactions with one another.
- Collaboration — Various enhancements including support for multiple queues and the ability for Customer Service Representatives to route or prioritize the customers waiting for assistance.
- Configuration Manager — Now contains tools to configure remote Web servers. Also new in this release is the Password Manager tool.

- Coupon Promotions — Various enhancements including support for guest customers to capture and redeem coupons and the enhanced capabilities for the Marketing Manager to manage coupons.
- Documentation — Two new guides have been introduced for WebSphere Commerce 5.5: the *Administration Guide* and the *Sample Store Guide*. The online help applicable to developers such as API and command reference information, is fully integrated with WebSphere Commerce Studio. The WebSphere Commerce help system has been tailored for Site Administrators and business users. API and command reference information has been moved to the WebSphere Commerce Studio help.
- E-mail activities — Enhancements include: opting out from e-mail at a store level rather than site, viewing statistics on e-mail activities, user interface for registering e-mail activity content.
- Outbound messages notification — Allows you to monitor whether outbound messages (including e-mail messages) have been sent or not sent.
- Installation — WebSphere Commerce 5.5 and all its associated software can now be installed through the WebSphere Commerce Install Shield. There are now three install methods: Typical, Custom, and Quick. The Quick install method allows you to quickly install WebSphere Commerce and create a WebSphere Commerce instance with minimal user interaction.
- Logging — Now uses the WebSphere Application Server logging facility, the JRas Logging Toolkit. JRas consists of multiple Java™ packages that provide message logging and diagnostic trace primitives and includes loggers, handlers, formatters, and so on.
- Messaging — Has migrated to the Java 2 Enterprise Edition Connector Architecture (J2EE/CA) standard (though the existing message system functionality is the same).
- WebSphere MQ — The existing listener for WebSphere MQ (previously, MQ transport adapter) now uses the new Java 2 Enterprise Edition Connector Architecture (J2EE/CA) infrastructure to integrate with JMS/MQ. WebSphere Commerce 5.5 supports WebSphere MQ Version 5.3
- Organization administration console — New functionality includes: search for organizations, the organization list appears with the list of organizations that meet search criteria, and members can be included or excluded from a group to which they implicitly belong, and so on.
- PRM messages — PRM (Partner Relationship Management) messages that represent the business flows between the Reseller marketplace and the distributors for the business indirect model have been introduced.
- Product advisor — Numerous enhancements to Product Advisor tasks.
- Product management — New wizards and notebooks have been created for bundles and kits, and merchandising associations. You can now use a wizard for creation and a notebook for changes.
- QShell Commands — iSeries for OS/400® command line utilities were provided by CL commands, or native commands. Now all command line utilities are run from the QShell. To run the commands, you need to enter the QShell.
- Sample stores — Existing sample stores such as FashionFlow and Tooltech (shipped with Business Edition only) have been enhanced, and a new CommercePlaza store is introduced (Business Edition only).
- Store development — The Publish utility has moved from Store Services to the Administration Console. The Store Profile, Tax and Shipping notebooks have now moved to the WebSphere Commerce Accelerator, allowing you to edit live

data in the database, rather than the store archive. As a result, Store Services no longer exists. The packaging and publishing functionality is more flexible.

- Subsystems — See the *What's New in WebSphere Commerce for Version 5.5* document for details on subsystem changes.
- Systems management — Enhancements have been made to administration, problem determination, logging, messaging, and performance monitoring.
- Web Services — Web services are self-contained, self-describing, modular applications that can be published, located, and invoked across the Web. Web services functions that can be invoked range from a simple request to complicated business processes. Once a Web service is deployed and registered, other applications can discover and invoke the deployed service. The foundation for Web services are standards such as simple object access protocol (SOAP), the Web services description language (WSDL), and the Universal Description, Discovery, and Integration (UDDI) registry. You can allow WebSphere Commerce to be the service provider by enabling its business functions as Web services that can be accessed by external systems. You can also allow WebSphere Commerce to be the service requestor by enabling it to invoke Web services hosted by external systems.
- WebSphere Commerce Payments — Formerly known as Payment Manager, WebSphere Commerce Payments is now integrated with WebSphere Commerce. As such, Payments is part of the WebSphere Commerce installation and configuration. New features include support for the new Cassette for Paymentech. The Cassette for VisaNet supports the use of SSL gateways to process credit-card transactions through the Internet. It also enables transactions to be processed through the First Horizon Merchant Services financial network in addition to the Vital Processing Services financial network. In WebSphere Commerce Payments, the SET™ and CyberCash payment methods are no longer supported.
- WebSphere Commerce Roles — Two new roles introduced in this release of WebSphere Commerce: Channel Manager and Registered Customer.

After you have migrated your Commerce Suite 5.1 system to WebSphere Commerce 5.5 as described in this Migration Guide, check the WebSphere Commerce 5.5 online help and other product documentation for information on how to use these features in your migrated system.

In addition to the above, important enhancements and features relative to WebSphere Commerce Suite 5.1 were introduced in WebSphere Commerce 5.4. These features and enhancements, in the following areas, have been carried forward in WebSphere Commerce 5.5. For more details on the enhancements, refer to the *What's New In IBM WebSphere Commerce, 5.4* document available at the WebSphere Commerce Technical Library Web page (<http://www.ibm.com/software/commerce/library/>):

- Collaboration

WebSphere Commerce supports two types of collaboration functionality: collaborative workspaces and customer care.

Collaborative workspaces are discussion forums for multiple parties to share business information such as negotiating contract terms and conditions between a Buyer and a Seller (or Account Representative for the Seller) and amongst the business users within the Seller's organization. Collaborative workspaces support asynchronous communication by way of Lotus® QuickPlace®.

Customer care provides real-time customer service support through a synchronous text interface using the Lotus Sametime® server. A customer may

enter the site and click a link on the store page to connect to a Customer Service Representative (CSR). The two parties can then communicate or chat over the Internet.

- Contract based commerce

In contrast to the account feature of WebSphere Commerce, contracts define the details of transactions between the seller and a unit of the buyer organization. Contracts contain certain information about orders from all of the customers governed by the contract. Using the WebSphere Commerce Accelerator, your Account Representatives and Sales Managers have control of the accounts and contracts that are defined for your store.

- Inventory management

The inventory subsystem is a component of WebSphere Commerce that provides real-time inventory management. The inventory subsystem provides functionality to record information about inventory received from vendors and returned by customers; adjust inventory quantity; determine the disposition of returned inventory; and ship and receive inventory. WebSphere Commerce supports inventory functionality in the following areas:

- Ad hoc inventory receipts
- Expected inventory
- Inventory adjustment
- Releases and shipments
- Fulfillment
- Operational reporting
- Inventory returns and dispositions

- Organization Administration Console

This buyer-side interface enables the buyer administrator to approve the shoppers within their organization, and the orders made by those shoppers. The buyer administrator also has the ability to assign access levels to their members.

- Product Management

The Product Management tools in the WebSphere Commerce Accelerator allow you to manage the products in your store's master catalog using various wizards and notebooks. The Product Management tools replace the Catalog Editor in previous versions. Using the WebSphere Commerce Accelerator, you can work with:

- Products
- SKUs
- Categories
- Attributes
- Pricing

- Request for Quotation (RFQ)

WebSphere Commerce Business Edition features RFQ support. These requests are sent to the seller organizations in order to get price quotes for unique items, or for large volume purchases. Once received, the seller organization responds by creating an RFQ response. Orders can be generated directly from RFQ responses. You have complete control of how RFQ support is processed at your site using XML files which define and govern the flow of an RFQ from receipt to resolution.

- Returns and refunds including RMA functionality

In order to appease a customer not satisfied with their purchase, the merchant or seller can offer a refund. That refund may then be used towards a new order for

replacement merchandise (this is not the same as an exchange, which is not currently supported). System settings, which can all be overridden with the appropriate level of security, control whether certain items are refundable, whether the original merchandise must be returned, and the refund amount. If the merchant or seller chooses, a Return Merchandise Authorization (RMA) may be issued through a self-service interface, using the configurable auto-approval mechanism or with the assistance of a Customer Service Representative (CSR). Once the returned merchandise is received, it may be marked unsalable and then discarded, or it may be returned to inventory to satisfy another order.

- Search Enhancements

WebSphere Commerce includes a number of search methods to facilitate providing search functionality to your users and to your customers. The following search methods are provided:

- Catalog search
- Product Advisor
- WebSphere Commerce Accelerator tools

After you have migrated your Commerce Suite 5.1 system to WebSphere Commerce 5.5 as described in this Migration Guide, check the WebSphere Commerce 5.5 online help and other product documentation for information on how to use these features in your migrated system.

Who should perform the migration?

Since the migration tasks are quite technical, the bulk of these tasks should be performed by a System Administrator. The following is a summary of the various users and their expected role in the migration process:

System Administrator

Knowledge and experience:

- Familiar with programming (for example, Java, JavaServer Pages, and so on)
- Familiar with database administration
- Web master
- Knowledgeable on System Architecture

Tasks:

- Installs, configures, and maintains WebSphere Commerce.
- Manages the database.
- Manages the Web server.
- Controls access.
- Manages data updates through mass import or other mechanisms.

Expectations of the migration process:

*The migration programs and procedures should allow the current **system assets** to be migrated with minimum downtime.*

Store Developer

Knowledge and experience:

- Familiar with programming.
- Familiar with multimedia tools

Tasks:

- Creates and customizes the store.
- Sets up and customizes payment, shipping, and tax support.

Expectations of the migration process:

*The migration programs and procedures should allow the current **store assets** to be migrated with minimum downtime.*

Store Administrator

Knowledge and experience:

- Familiar with business-procedures.
- Web-literate.

Tasks:

- Manages orders
- Processes payments
- Assists shoppers
- Maintains online store
- Modifies online store

Expectations of the migration process:

Information acquired online such as orders and shoppers will be available and usable after migration.

Catalog Administrator

Knowledge and experience:

- Is a product expert.
- Is Web-literate and computer-literate.
- Is familiar with multimedia tools.

Tasks:

- Creates store catalogs
- Creates and manages products and categories
- Creates and manages pricing schemes
- Creates and manages reports

Expectations of the migration process:

Information using Commerce Suite 5.1 will not have to be recreated. Tools will be adaptable to extensions to the catalog.

Supported migration paths

Note:

This Guide describes the process to migrate to WebSphere Commerce 5.5 on the IBM @server iSeries.

The following migration paths for all national language versions are supported:

- Commerce Suite 5.1 Pro Edition to WebSphere Commerce 5.5 Professional Edition or Business Edition

Attention: This Migration Guide has been tested only for the above migration paths. This Guide does not address migration to WebSphere Commerce 5.5 from any version other than Commerce Suite 5.1.

Note that direct migration from WebSphere Commerce 5.4 to WebSphere Commerce 5.5 is also supported. To migrate from WebSphere Commerce 5.4, see the *WebSphere Commerce Migration Guide* for that path.

For any information on migration issues arising from the application of future fix packs, interim fixes, or other enhancements, see the WebSphere Commerce Support page (<http://www.ibm.com/software/commerce/wscom/support/index.html>)

Migrating from earlier versions

This guide describes the migration process for the above supported migration paths.

Direct migration from earlier versions of Net.Commerce™ or Commerce Suite to WebSphere Commerce 5.5 is not supported.

To migrate from earlier versions of Net.Commerce or Commerce Suite, you must first migrate your existing system to one of the above supported levels and then use this document to migrate to WebSphere Commerce 5.5.

- To transition and migrate to Commerce Suite 5.1 from earlier versions of Net.Commerce, see the *WebSphere Commerce Suite 5.1 Migration and Transition Guide*.

Migration paths by operating systems

WebSphere Commerce 5.5 supports the migration of Commerce Suite 5.1 on the same operating system. It specifically supports migration from the following paths:

- AIX® to AIX
- iSeries to iSeries
- Solaris Operating Environment to Solaris Operating Environment
- Windows® 2000 to Windows 2000
- Windows NT® to Windows 2000

For information on the steps to migrate to WebSphere Commerce 5.5 for any of the the above platforms, refer to the appropriate *WebSphere Commerce Migration Guide* for Version 5.5, for that platform.

WebSphere Commerce 5.5 does not support migration between dissimilar operating systems, for example, Commerce Suite 5.1 on Windows NT to WebSphere Commerce 5.5 on iSeries.

Other considerations

- If you install WebSphere Commerce 5.5, you must first upgrade the operating system level to V5R2. WebSphere Application Server 3.5, which is required for Commerce Suite 5.1, is not supported on the OS/400 V5R2 level. Therefore, you cannot continue to use your Commerce Suite 5.1 instances, once you have completed the system upgrade. To minimize downtime, you may wish to perform the migration on a second machine, or on a second partition of the machine.
- If you have customized code or commands written in Java or Enterprise JavaBeans™ objects in Commerce Suite 5.1, you need to redeploy them to the

level required for WebSphere Commerce 5.5. See the "Transitioning your customized code" section of the *WebSphere Commerce Studio Migration Guide*. This transition is required due to the move to WebSphere Application Server 5.0 from IBM WebSphere Application Server 3.5.x.

Conventions used in this book

This book uses the following highlighting conventions:

- **Boldface type** indicates commands or graphical user interface (GUI) controls such as names of fields, icons, or menu choices.
- Monospace type indicates examples of text you enter exactly as shown, file names, and directory paths and names.
- *Italic type* is used to emphasize words. Italics also indicate names for which you must substitute the appropriate values for your system. When you see any of the following names, substitute your system value as described:

host_name

The fully qualified host name of your WebSphere Commerce server (for example, `mymachine.mydomain.ibm.com` is fully qualified).



This icon marks a Tip - additional information that can help you complete a task.

Note: References to "iSeries" in the book should be understood to mean the OS/400 operating system running on the iSeries hardware platform, and not the Linux for iSeries operating system.

Path variables

This Guide uses the following variables to represent directory paths:

WAS35_installdir

This variable represents the actual installation directory for WebSphere Application Server Version 3.5.x provided with Commerce Suite 5.1:
`/QIBM/ProdData/WebASAdv`

WAS50_installdir

This variable represents the actual installation directory for WebSphere Application Server Version 5.0 provided with WebSphere Commerce 5.5:
`/QIBM/ProdData/WebAS5/Base`

WAS50_userdir

This variable represents the directory for all the data that is used by WebSphere Application Server 5.0 which can be modified or needs to be configured by the user: `/QIBM/UserData/WebAS5/Base/WAS_instance_name`

WCS51_installdir

This variable represents the actual installation directory for Commerce Suite 5.1: `/QIBM/ProdData/CommerceSuite5`

WCS51_userdir

This variable represents the directory for all the data that is used by WebSphere Commerce which can be modified or needs to be configured by the user: `/QIBM/UserData/CommerceSuite5`

WC55_installdir

This variable represents the actual installation directory for WebSphere Commerce 5.5. The following is the default installation directory:
/QIBM/ProdData/CommerceServer55

WC55_userdir

This variable represents the directory for all the data that is used by WebSphere Commerce which can be modified or needs to be configured by the user: /QIBM/UserData/CommerceServer55

WPM221_path

This variable represents the actual installation directory for the previous version of WebSphere Commerce Payments (WebSphere Payment Manager 2.2) provided with Commerce Suite 5.1. The following is the default installation directory: /QIBM/ProdData/PymSvr

Payments_installdir

This variable represents the actual installation directory of WebSphere Commerce Payments Version 3.1.3 provided with WebSphere Commerce 5.5. The following is the default installation directory:
/QIBM/ProdData/CommercePayments/V55

Note: The above default installation paths cannot be altered or modified on iSeries. The WebSphere Commerce for iSeries product expects the above directories to exist, and will not function properly if they do not exist.

Running database scripts

In many sections throughout this book, you will be required to run scripts against your database. Other occurrences throughout this book refer to this description.

To run scripts against the database using IBM iSeries Access for Windows, V5R2 , do the following:

1. Open the iSeries Navigator.
2. Click on the system corresponding to the iSeries server where the database is located.
3. Expand the **Databases** icon, right-click your database name, and select **Run SQL Scripts**.
4. The **Run SQL Scripts** window appears.
5. In this window, enter the SQL statements or scripts as detailed in this book. Optionally, you can use this window to open and edit a script.

Note: You can set your default schema by clicking the **Connection** menu, and then selecting the **JDBC Setup** submenu.

References to Oracle

You may find some references to “Oracle” in some pieces of code during the migration process. These are artifacts of the port of WebSphere Commerce to the iSeries platform from the Windows platform. You can ignore these references as Oracle is not supported on the iSeries platform.

Part 1. Required migration steps

The chapters in this part of the Migration Guide describe the mandatory tasks that are required to migrate to WebSphere Commerce 5.5 from Commerce Suite 5.1.

They include:

- Chapter 1, “Before you migrate from Commerce Suite 5.1,” on page 3
- Chapter 2, “Backing up Commerce Suite 5.1,” on page 17
- Chapter 3, “Upgrading your software,” on page 21
- Chapter 4, “Migrating your Commerce instance configuration,” on page 29
- Chapter 5, “Before migrating your database,” on page 53
- Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59
- Chapter 7, “Post instance and database migration,” on page 67
- Chapter 8, “Migrating to Commerce Payments,” on page 81

In addition, Chapter 11, “Migrating other WebSphere Commerce components,” on page 109 describes optional post migration actions that you may wish to take depending on your requirements.

Chapter 1. Before you migrate from Commerce Suite 5.1

The following sections describe specific steps you must complete while you still have an operational Commerce Suite 5.1 system. The sections also describe some considerations you must make prior to beginning your migration to WebSphere Commerce 5.5.

Important

- This Guide describes two methods of implementing the migration process:
 - *In-place* migration - which requires a single machine
 - *Remote* migration - which requires two machines.

While the in-place method is convenient since it requires only one machine, most users would likely find the downtime for their system using this method to be excessive, particularly for a production site. The complete migration process time duration consists of two major items:

- Database migration - which could take many hours, depending on the size of the database (number of users, products, orders, user traffic, and so on).
- Code migration - where code customization and testing may take a number of days.

Since the live Commerce Suite 5.1 site has to be shut down as soon as the migration effort starts, the *remote method is recommended over the in-place method for live sites where downtime is of concern*. The in-place method is suitable however for customization and testing of the migration process on a site that is not live.

- If you have customized code or commands written in Java or Enterprise JavaBeans in Commerce Suite 5.1, you need to redeploy them to the level required for WebSphere Commerce 5.5. See the "Transitioning your customized code" section of the *WebSphere Commerce Studio Migration Guide*. This transition is required due to the upgrade to WebSphere Application Server 5.0 from WebSphere Application Server 3.5.x.
- After you install WebSphere Commerce 5.5, you need to install the IBM WebSphere Commerce 5.5.0.2 fix pack. This fix pack includes fixes and enhancements for the migration process. This fix pack is available from the WebSphere Commerce Support site (<http://www.ibm.com/software/genservers/commerce/support/>). Click on the link for the edition you are using and see the section **Download**. Download this fix pack, and follow the instructions that are provided to install it. You should not proceed with the migration steps in this Guide (such as migrating your instance or database) without installing this fix pack.

Premigration actions

Before proceeding with the migration steps contained in the rest of this guide, ensure that you take the following actions while your system is operational at the Commerce Suite 5.1 level.

Ensuring no new messages are stored in MSGSTORE table

In Commerce Suite 5.1, when you send a SendTransacted message, a message is stored in the MSGSTORE table in the database. Once the message is sent, the entry is removed from the database. In fact, the data in this table is considered to be transient because there should not be any remaining messages in this table (unless some settings are incorrect). In your Commerce Suite 5.1 system, you should have been cleaning up this table on a regular basis.

When you migrate to WebSphere Commerce 5.5, the data in this table cannot be reused. Before shutting down your original Commerce Suite 5.1 instances, you should perform the steps below. A period of time is required where the original system does not take any further orders or requests, in order to bring the system to a halt gracefully. During this period, ensure that there are no new messages stored in the MSGSTORE table as follows:

1. Log on to the Administration Console as the Site Administrator.
2. At the Site or Store selection panel, select **Site**
3. Open the Configuration - Transports menu
4. Change the status of each transport to **Inactive**
5. Repeat the steps above to disable the transports for each store

Disabling the transport does not prevent messages in the MSGSTORE table from being sent. It only prevents new messages from being saved in the MSGSTORE table.

The scheduler will attempt to deliver all messages in MSGSTORE. (By default, the scheduler runs SendTransactedMsg job at 5 minute intervals, and the number of retries is 3.)

After 15 minutes, there should not be any message in MSGSTORE with retries that are greater than zero.

6. Check whether there are any remaining entries in MSGSTORE with retries greater than zero using the following SQL statement:

```
select count(distinct msgid) from msgstore where retries > 0
```

The select statement shows how many messages Commerce Suite 5.1 still needs to deliver. The result is one row, one column with a numeric result that is 0 (zero) or greater.

If the result is 0 (zero), it means that there are no pending messages to be delivered, and that the table is ready to be migrated.

If the result is greater than 0 (zero), it means that Commerce Suite 5.1 is still attempting to deliver remaining messages. Note that after this count reaches 0 (zero), it should not increment if you subsequently run this SQL. If you notice that the result keeps on incrementing, it means that not all transports have been shut down. Refer to the steps above to disable the transport (you need to do that for the site as well as all stores).

Once the SQL select statement returns 0 (zero), you can run the following:

```
select count(distinct msgid) from msgstore where retries = 0 or retries = -1
```

This select statement determines whether there are any messages that have not been delivered.

If the result is zero, there are no left over messages.

If the result is anything larger than zero, there are messages left in the MSGSTORE table. It is recommended that you delete the left over messages. (It is always a good practice to clean up this table periodically by deleting left over messages.)

Closing your existing auctions

If you are going to migrate your inventory representation to use the new ATP inventory representation, you must close all existing auctions in your Commerce Suite 5.1 system before migrating to WebSphere Commerce 5.5. For information on closing your auctions, see the topic "Close bidding for an auction" in the Commerce Suite 5.1 online help .

Preparing the staging server for migration

You must synchronize your staging server database with your production Commerce Suite 5.1 database prior to migration. You can either:

- Run the Stage Propagate utility command (PRPWCSTG) prior to the data migration. See the topic "Stage Propagate utility command " in the Commerce Suite 5.1 online help for more information on running the utility.
- If you do not wish to propagate the changes you have made to your database, you can use the Stage Copy utility command (CPYWCSTG) to synchronize your databases. (This option will normally take much longer.) See the topic "Stage Copy utility command " in the Commerce Suite 5.1 online help for more information on running the utility.

After synchronizing your staging server database:

- Clean up the staging database to speed up migration (optional).

In order to speed up the staging server database migration, run the Stage Copy utility (CPYWCSTG) using the CLEANUP(*ONLY) option. For details on how to run the Stage Copy utility, see the information in the section "Staging server commands" in the Commerce Suite 5.1 online help.

Note: You need to drop all your customized triggers. Otherwise those triggers may get executed during the staging database migration and create an unexpected problem. You need to remember those triggers and reapply them after the staging database migration if necessary.

- Migrate your staging server database using the database migrations script as documented in Chapter 6, "Migrating your Commerce Suite 5.1 database," on page 59. The process of migrating this database will be similar to your production database.

Checking the unique index in the CONTRACT table

The unique index in the CONTRACT table for WebSphere Commerce 5.5 is now constructed based on the following columns:

- NAME
- MEMBER_ID
- MAJORVERSION
- MINORVERSION
- ORIGIN

If the MEMBER_ID, MAJORVERSION, MINORVERSION, and ORIGIN values are the same across two or more rows, you need to make sure that the NAME column in the CONTRACT table is different across these rows to ensure uniqueness. Otherwise, you may encounter problems when migrating your data from Commerce Suite 5.1.

Premigration considerations

The following sections highlight some key items for WebSphere Commerce 5.5 that you need to consider prior to beginning the migration process.

Migrating to ATP inventory

Previous versions of WebSphere Commerce provided task command interfaces to check and update available inventory levels. The default task command implementations used the INVENTORY table to record available inventory levels. This previous level of capability is referred to as *Compatibility Mode Inventory*.

Table 1. *Compatibility Mode Inventory*

Compatibility Mode Inventory Task Command Interface	Description	Called By
ResolveFulfillmentCenterCmd	Determines the fulfillment center for an order item.	OrderItemAdd, OrderItemUpdate, OrderPrepare
CheckInventoryCmd	Checks to see if there is sufficient available inventory for an item.	ResolveFulfillmentCenterCmd
UpdateInventoryCmd	Decreases the available inventory for an item.	OrderProcessCmd, PaySynchronizePM
ReverseUpdateInventory	Increases the available inventory for an item.	Order Management user interface when canceling an Order.

WebSphere Commerce 5.5 enhances this capability (originally enhanced in the WebSphere Commerce 5.4 release) with new task command interfaces to check, allocate, or back order available or expected inventory items. The new default task command implementations use information in the RECEIPT, RADETAIL, and other related tables. When items are checked or back ordered, estimated availability times are obtained. This enhanced capability is referred to as *Available to Promise (ATP) Inventory*. Allocations and back orders can expire if payment is not initiated on time.

Table 2. *Available to Promise (ATP) Inventory*

ATP Inventory Task Command Interface	Description	Called By
AllocateInventoryCmd	Checks, allocates, or back orders available or expected inventory. Determines fulfillment centers and estimated availability times. Can also be used to reverse allocations or back orders.	OrderItemAdd, OrderItemUpdate, OrderPrepare, OrderProcess, ProcessBackOrders
GetEligibleFulfillmentCentersCmd	Determines a prioritized list of fulfillment centers.	AllocateInventoryCmd
CheckInventoryAvailabilityCmd	Obtains estimated availability times.	AllocateInventoryCmd
AllocateExistingInventoryCmd	Allocates available inventory.	AllocateInventoryCmd
DeallocateExistingInventoryCmd	Reverses an allocation.	AllocateInventoryCmd, ReleaseExpiredAllocations
AllocateExpectedInventoryCmd	Creates a back order.	AllocateInventoryCmd
DeallocateExpectedInventoryCmd	Reverses a back order.	AllocateInventoryCmd, ReleaseExpiredAllocations

The OrderItemAdd, OrderItemUpdate, and OrderPrepare commands have new parameters that allow the caller to check, allocate, or back order inventory, or reverse allocations or back orders. OrderProcess always allocates, or, if unable to allocate, back orders inventory for any OrderItems not yet allocated nor back ordered.

Table 3. ATP Inventory Commands

ATP Inventory Enabled Command	Enhanced ATP Capabilities	Default Action
OrderItemAdd, OrderItemUpdate	Check, allocate, back order, reverse.	Check.
OrderPrepare	Check, allocate, back order, reverse.	Allocate or back order.
OrderProcess	Allocate or back order.	Not applicable.

A new column in the STORE table, named ALLOCATIONGOODFOR, is used to enable the ATP Inventory capability for a store. When the value in this column is zero, Compatibility Mode Inventory is enabled. When the value is greater than zero, ATP Inventory is enabled. The value specifies the number of seconds after which allocations and back orders expire, if payment is not yet initiated. When you migrate your database from Commerce Suite 5.1 to WebSphere Commerce 5.5, you can choose to migrate to ATP Inventory. If you do so, then information in the INVENTORY table will be moved to the RECEIPT table, and STORE.ALLOCATIONGOODFOR will be set to its default value (43, 200 seconds or 12 hours).

Catalog entries in the WebSphere Commerce Suite 5.1 INVENTORY table that are either products, DynamickitBean, or bundles, will not be moved to the WebSphere Commerce 5.5 RECEIPT table during the database migration as WebSphere Commerce 5.5 inventory function does not support them. Only catalog entries that are packages or items in the WebSphere Commerce Suite 5.1 INVENTORY table will be moved to the WebSphere Commerce 5.5 RECEIPT table during the database migration. The new ATP function supports only these types of catalog entries.

If you choose not to migrate to ATP Inventory at this time, you can do so at a later time using the migrateATP script. This script is described in Appendix F, “Converting to ATP inventory at a later time,” on page 147.

Orders and order items

There are two premigration considerations you need to make with respect to your Commerce Suite 5.1 orders or order items:

- Orders or order items are in the M state (that is, the shopper has initiated payment and the inventory update is successful, but the order or order item has not been authorized).
In this case, you should either complete, or delete, or cancel the orders or order items as appropriate prior to migration. Usually, orders or order items in this state are simply waiting authorization to proceed, and are only in the M state over a short time period. They will however, remain in this state if authorization either fails or gets rejected. If you migrate these orders and order items while they are in the M state, WebSphere Commerce 5.5 will run the PayResetPMCmd command as a scheduled job after the migrated instance is restarted, to clean up these orders and order items.
- Orders or order items are in the C state (that is, payment has been authorized).
If the items are truly complete and have been shipped, you should move the order or order item to a final S state (that is, the order item has been shipped). This will prevent the order items from being allocated again in WebSphere Commerce 5.5.

During the database migration process, a `ctos.sql` script gets generated by the database migration script in the `WC55_userdir/instances/instance_name/migration/DB2` directory whenever there are any orders that are in C state. The `ctos.sql` script changes the status of any orders or order items with a status of C, from C to S. You should run the `ctos.sql` script in the following cases:

- If you run the database migration script using the ATP option:
You need to run the `ctos.sql` script prior to restarting the Web server and the WebSphere Commerce Server – `instance_name` application server after migration.
- If you run the database migration script without using the ATP option and complete your migration, then later decide to migrate to ATP inventory after running your migrated system:
Running the `migrateATP` script after migrating to WebSphere Commerce 5.5 generates the `ctos.sql` script. In this case, you need to run it before you restart the Web server and the WebSphere Commerce Server – `instance_name` application server.

Although you will be able to view and edit the migrated orders and order items that are in the C state with the WebSphere Commerce Accelerator tools, *it is not recommended that you edit them using these tools*. If you try to edit them using these tools (which you should not do as they are already completed), there may be nonrecoverable errors. In this case, the order status changes to either E state (CSR edit - a Customer Service Representative is working with the order) or T state (temporary - used by the Order Management user interface to temporarily back up an order). The CSR can view the summary of the E state order to find out the order number for the T state order. The T state order is the backup copy of the original order (before editing with the tools). The CSR can print out this T state order for reference and using the Commerce Accelerator, manually recreate the order for the customer.

Note that the migration script does not associate the order items in the C state with a specified item (ITEMSPC) as this may degrade performance, as there may be large numbers (millions) of completed order items.

Master catalog

In WebSphere Commerce Suite 5.1, the catalog system did not require structured catalog data. With the introduction of the master catalog in WebSphere Commerce 5.4, WebSphere Commerce 5.5 requires catalog data to be structured in a certain way.

In WebSphere Commerce 5.5, the master catalog is the central location to manage your store's merchandise. Everything you need in your store is contained within the master catalog. It is the single catalog containing all products, items, relationships and standard prices for everything that is for sale in your store.

Every store in the WebSphere Commerce system must have a master catalog. You can share the master catalog across stores and define as many stores as needed. In addition to creating a master catalog for catalog management, you may also choose to create one or more navigational catalogs for display purposes. A navigational catalog may contain the same entries as the master catalog, but with a much more flexible structure for customer display purposes. You can have as many navigational catalogs as you want. However, since it is the master catalog that is used to manage your online merchandise, it is recommended that you also use the master catalog as your navigational catalog to minimize maintenance overhead.

You can use the Product Management tool to view and manage your master catalog

For additional information on creating and managing a master catalog, see the WebSphere Commerce 5.5 online help after you have completed your migration.

Important structural considerations

In order to use the WebSphere Commerce 5.5 catalog tools such as the Product Management tool, your master catalog must meet the following conditions:

- The master catalog must be a proper tree, which means that there are no cycles. This implies that you must avoid the following type of scenario: The parent category **A** has a subcategory **B**. It is important that **B** and any of **B**'s subcategories are not the parent category of **A**.
- You cannot categorize a product under more than one category. To place a product in more than one category, use a navigational catalog.
- All items that belong to a product must be categorized under the same category as the product is categorized.
- The Product Management tool only works with a master catalog.

Migration of catalog information

If you are currently using one catalog in your Commerce Suite 5.1 store, the database migration script will assign that catalog as the master catalog for your store.

If you are currently using multiple catalogs in your Commerce Suite 5.1 store, you need to begin thinking about which one needs to be assigned as the master catalog for your migrated store. This assignment is done using the `choosemc.sql` script as described in "Assigning a master catalog" on page 63.

Implications when running the database migration scripts

The tool for catalog management (the Product Management tool available in the WebSphere Commerce 5.5 Administration Console) will use products as its starting point. All products have sets of attributes, and any items that have those common attributes can be added under a product.

In Commerce Suite 5.1, there is no enforcement that any items must belong to a product. During the premigration phase of database migration, the premigration script checks if there are any orphaned items in your Commerce Suite 5.1 database. That is, whether your Commerce Suite 5.1 database contains any items that *do not* have any parent products.

When migrating to WebSphere Commerce 5.5:

- If you do not want the database migration script to create one product per orphaned item:

You will need to group your items with common attributes, and create a product before you begin the migration to WebSphere Commerce 5.5. The following tables need to be populated:

- CATENTRY (create ProductBean)
- CATENTDESC (product description)
- CATENTREL (relationship between the product and its items)

The advantages of this method are:

- You will have organized your items under their product, and thus you do not need to reorganize your items after the migration to WebSphere Commerce 5.5.
- There will be less products to manage.
- If you select to use the default migration script action:

If you wish to allow the supplied data migration script to create a product for each of orphaned items as a placeholder, you can use the WebSphere Commerce 5.5 product management tool to manage the products after migration, and add items which have similar attributes to the product . You can organize the new items with similar attributes under the same product.

The advantages of this method is that you can exploit all the functionality of the WebSphere Commerce 5.5 product management tool. The implications of this method are:

 - You will need to manage both the placeholder products and the items.
 - Your database space disk space requirement will be larger than before. The migration script tool will not be able to determine which items should grouped under which products, which is why it creates one *placeholder* product per orphaned item. You may need to reorganize your *placeholder* products and item relationships. The SKU (Partnumber) in the *placeholder* product is based on the part number of orphaned item (P-Partnumber of item).

Default contract

Business

WebSphere Commerce 5.4 introduced terms and conditions to provide contract support that are carried over to WebSphere Commerce 5.5. The migration process will create a default contract for your system which will have the same behavior and characteristics as the WebSphere Commerce Suite 5.1 business flows (for example, for shipping charges).

Normally, you do not need to take any actions during the migration process, since the default contract is automatically created. If you need to create additional contracts for your business processes, use the WebSphere Commerce Accelerator to do so, after you have migrated to WebSphere Commerce 5.5. See the section "Creating a new contract" in the WebSphere Commerce 5.5 online help.

Access control

Access control is an integral part of the overall security and flow control for a commerce site. Site administrators and participating organizations need to control which participants can perform which actions on which objects in the system. In order to increase the customizability by externalizing the access control decisions from the code that manipulates business objects, WebSphere Commerce 5.5 resource level access control is policy-based instead of programmatic as in Commerce Suite 5.1.

In WebSphere Commerce 5.5, access control is managed using a GUI and XML files which are used to define access control policies. These policies are stored in the WebSphere Commerce 5.5 database. They are loaded into memory when you startup the WebSphere Commerce 5.5 system.

Following are the key improvements made to access control in WebSphere Commerce 5.5 that were originally introduced in WebSphere Commerce 5.4:

- To increase flexibility, access control policies are externalized from code that manipulates business objects.
- Hierarchical access control is built into the access control model.
- All access control policies are based on groups: ActionGroups, ResourceGroups, AccessGroups.

In addition, an organization can now explicitly subscribe to groups of policies, in which case, the organization has total control over the policies that will apply to it.

If you are satisfied with the command level access control as implemented by Commerce Suite 5.1, and do not intend to make any code changes:

- You will need to migrate the Commerce Suite 5.1 access control tables to the appropriate policies and policy groups. In order for this to work properly, you must run the database framework migration as described in Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59. If this is not done, the access control policies for some customized commands and views will not be migrated.
- If you have overridden the `getResourceOwners()` method, each of the resource owners returned will be used as the owner of the protectable resource, the command, for command level access control.
- If you have not overridden the `getResourceOwners()` method, the owner of the protectable resource, the command, will be the store owner, if `storeId` is specified in the command context. Root organization will be used if no `storeId` is specified in the command context.
- If you have overwritten the method `checkPermission()`, this method will be called after command level access control is performed.

If you want to take advantage of the resource level access control in WebSphere Commerce 5.5, in addition to the above steps, you have to implement the `getResources()` method. In addition, if you are migrating any existing commands, you may want to remove your own implementation of `getResourceOwners()` if you have implemented it. Using the store owner or site organization should be adequate for command level access control. The finer level of access control can be performed by the resource level access control.

For more information on access control, see Chapter 10, “Access control subsystem considerations,” on page 107.

Member subsystem

A key difference between WebSphere Commerce 5.5 and WebSphere Commerce Suite 5.1 is that in WebSphere Commerce 5.5, every user and organizational entity member must have a parent member which is another organizational entity. This allows users and organizational entities to form a membership hierarchy. Member groups are not part of the membership hierarchy and therefore they do not have parent members.

During the migration process, the database migration script determines the parent and ancestors for users and organizational entities based on the following:

- Whether users have records in the `BUSPROF` table and the values of the `ORG_ID` and `ORGUNIT_ID` columns.
- The value of the `MEMBER_ID` column in the `ORGENTITY` table for organizational entities.

As a result of determining the parent and ancestors of users and organizational entities, the MBRREL table is populated, which captures the membership hierarchy. WebSphere Commerce 5.5 business logic uses the membership hierarchy. Therefore, you should ensure certain columns in your database contain the proper values so that the membership hierarchy can be properly determined. The parent and ancestors of users and organizational entities are determined as follows by the database migration script:

- For users with a record in the BUSPROF table and with profile type set to B (B2B user):
 - The ORGUNIT_ID is used as the parent member ID if it is not null.
 - If the ORGUNIT_ID is null, the ORG_ID is used as the parent member ID if it is not null.
 - If both ORGUNIT_ID and ORG_ID are null, the Default Organization organizational entity (ORGENTITY) is used as the parent member.

To prevent your B2B users from being assigned the Default Organization as their parent in the migration process, you should scan the Commerce Suite 5.1 BUSPROF table and try to fill in the ORGUNIT_ID and ORG_ID columns prior to migration. It is not recommended that Default Organization be the parent organizational entity of B2B users. If you cannot fill in the ORGUNIT_ID and ORG_ID columns in the BUSPROF table for certain registered users, you should change the profile type of those registered users from B (B2B user) to C (B2C user).

There is a PROFILETYPE column in the USERS table which allows null, B, or C as valid values:

- B (registered B2B user)
- C (registered B2C user)
- null (no profile data)

If you have customized Commerce Suite 5.1 code and this code has not been setting the profile type for your users, you should *clean up* the PROFILETYPE column in the USERS table. For WebSphere Commerce 5.5, it is recommended that:

- B2C users be under Default Organization and have profile type C. Typically, B2C users do not have records in the BUSPROF table and have the Default Organization as their parent.
- B2B users have profile type B, have records in the BUSPROF table, and have the appropriate organizational entity as their parent. It is not recommended that B2B users be under Default Organization.

In addition, ensure that you set your administrators (that is, those users whose register type in the USERS table is an A or S and that have entries in the ACCMBRGRP table) to have a profile type of B.

- Users without a record in the BUSPROF table, will have their parent organizational entity set to the Default Organization by the database migration script.
- For organizational entities in the ORGENTITY table, the MEMBER_ID column is used as the parent member ID, if it is not null. If the MEMBER_ID column is null, the parent member is set to be the Root Organization.

With the new MBRREL table which contains records for registered users and organizational entities, the database migration script determines membership hierarchy solely using the content of the MBRREL table. Note that the MBRREL table does not contain records for the generic user, guest users, and member groups.

During migration, the following consistency check is performed by the database migration script: For a user with a record in BUSPROF, if the ORG_ID and

ORGUNIT_ID columns are not null, then starting with ORGUNIT_ID, the script uses the ORGENTITY table to traverse up the organizational hierarchy until MEMBER_ID eventually contains a null or contains the same value as ORGENTITY_ID. It checks that the value of MEMBER_ID column in the ORGENTITY table is the same as the ORG_ID column in the BUSPROF table. If it is not, the script notifies the user to correct the inconsistency.

Other member subsystem considerations

- During migration to WebSphere Commerce 5.5, the STATE column in the MEMBER table gets set as follows:
 - For guest users (where register type is G), the migration script sets STATE to null.
 - For registered users (where register type is R), the migration script sets STATE to approved.
 - For organizational entities, the migration script sets STATE to approved.
 - For member groups, the migration script sets STATE to null.

Organizational entities are migrated so that they do not own any approval groups, which means by default, the migrated organizational entities do not want approvals for business processes such as B2B users self-registration.

- The roles that are shipped by default in WebSphere Commerce 5.5 include all of the roles shipped in Commerce Suite 5.1 plus some new roles, with the following exceptions:

- The Order clerk role is not included.

Commerce Suite 5.1 supported an Order Clerk role which is no longer required or used in WebSphere Commerce 5.5. The tasks that used to be performed by the Order Clerk role have either been automated or can be performed by the Customer Service Supervisor in WebSphere Commerce 5.5. If a user has an Order Clerk role (-5) in Commerce Suite 5.1 and an entry in the ACCCMDGRP table, the user is migrated as part of the access control migration and the role is treated as a user-defined role.

Review if you still require the Order Role clerk. If you do not require it, remove it since it is no longer supported in WebSphere Commerce 5.5.

- The Store Developer role is not included.

Commerce Suite 5.1 supported a Store Developer role which is no longer required or used in WebSphere Commerce 5.5. The tasks that used to be performed by the Store Developer role have either been automated or can be performed by the Site Administrator in WebSphere Commerce 5.5. If a user has an Store Developer role (-7) in Commerce Suite 5.1 and an entry in the ACCCMDGRP table, the user is migrated as part of the access control migration and the role is treated as a user-defined role.

Review if you still require the Store Developer role. If you do not require it, remove it since it is no longer supported in WebSphere Commerce 5.5.

If any users belong to a Store Developer member group, change them to the Site Administrator member group before migration. If a user wants to maintain the Store Developer role in WebSphere Commerce 5.5, the migration script will migrate the Store Developer role as a user-defined role in WebSphere Commerce 5.5.

- The Customer role is not included.

Commerce Suite 5.1 included an access group called Customer. Every access group in Commerce Suite 5.1 has a name which is a role name. Normally, an access group in Commerce Suite 5.1 is used for two purposes:

- Assigning commands to the access group (association stored in the ACCCMDGRP table)
- Assigning users to the access group (association stored in the ACCMBRGRP table)

Users assigned to an access group play the role with a name same as the access group name. Thus, assigning a user to an access group is the same as assigning a role to that user, allowing the user to execute commands associated with that access group. The Customer access group in Commerce Suite 5.1 is associated with a set of commands that *all users* can execute. This means the Customer access group represents all users in the Commerce Suite 5.1 system, and it is not necessary to assign every user with the Customer role.

In WebSphere Commerce 5.5, an AllUsers member group is shipped to replace the Customer access Group. An access control policy has been created to associate the set of commands that can be executed by all users with the AllUsers member group. Since explicit assignment of every user with a Customer role is unnecessary, a Customer role is not shipped by default in WebSphere Commerce 5.5. For a particular organization entity, there is a set of users who are employees of the organizational entity, and the non-employees are considered *customers*. Users that have been explicitly assigned into the Customer access group in Commerce Suite 5.1, are explicitly assigned into the AllUsers member group during data migration to WebSphere Commerce 5.5. In such cases, the migration script issues a warning message, since such explicit assignment is not necessary.

- The Merchant role has been renamed to Seller (*Merchant* is a B2C term, and *Seller* is a B2B term).
- The Merchandising Manager role has been renamed to Product Manager for the same reason.
- In Commerce Suite 5.1, the USERS table contains a REGISTERTYPE column which accepts four valid values:
 - R — registered users
 - G — guest users
 - S — Site Administrator
 - A — Administrator

Register types S and A are role-related. In WebSphere Commerce 5.5, the REGISTERTYPE column in the USERS table continues to support the same set of values. However, given the set of roles that are available by default from WebSphere Commerce 5.5 the meaning of register type A needs to be refined. Type A is now given a more concrete meaning of: *employees of the Seller organization who play certain roles*. The set of roles which correspond to register type A is configurable by changing the definition of the Administrators access group using the Administration Console. In addition, the register type value of a user is set to an A or an S automatically during role assignment and unassignment, thus ensuring the value of the register type is consistent with the roles played by a user.

In summary:

- If a user from the Seller organization is assigned the Site Administrator role, the register type value for this user is S
- If a user from the Seller organization is assigned one of the roles defined in the Administrators access group other than Site Administrator, the register type value for this user is A.

- During migration, the database migration script explicitly assigns users from Commerce Suite 5.1 with register type of A and that do not belong to any access group, to the Administrators access group in WebSphere Commerce 5.5. Note that if there are no entries in the ACCMBRGRP table, then the script does not perform this step. Before running the migration script, you should do the following:
 - Examine the definition of the Administrators access group and modify the list of roles if necessary. For example, if you have assigned users in Commerce Suite 5.1 to an access group named XXX with REGISTERTYPE=A and XXX is not one of the roles listed in the Administrators access group in WebSphere Commerce 5.5, you should add role=XXX as an additional criteria to the Administrators access group.
 - If you have any customized logic which directly examines the REGISTERTYPE column of the USERS table for the S and A values, you should change the code as follows:
 - Replace your logic with an access control policy if your logic tries to examine the REGISTERTYPE column in order to determine authorization to be given to a user. In WebSphere Commerce 5.5, a new access control design is available and it is recommended that you use access control policies instead of hard coding authorization-related logic. See “Access control” on page 10.
 - If your logic examines the REGISTERTYPE column, but not for access control purposes, you should use one of the programming interfaces available from the Member Subsystem:
 - isAdministrator()
 - isSiteAdministrator()
 - isMemberInRole()

See the WebSphere Commerce 5.5 online help after you have migrated for more information on these interfaces. This change removes any dependency your code has on the actual values of register type. Note that in a future version of WebSphere Commerce, the set of values that are valid for register type may change.
 - During migration from Commerce Suite 5.1 to WebSphere Commerce 5.5, the EXCLUDE column in the MBRGRP table for migrated users within a member group is set to 0 (zero). A value of zero means such users are explicitly included in the member group.

Distinguished names

In WebSphere Commerce 5.5, the DN column in the ORGENTITY needs to be populated. It should contain all the parent hierarchy of the current organizational entity. During data migration, the migration script generates the fillorgDN.sql script in the *WC55_userdir/instances/instance_name/migration/DB2* directory. However, if you have a large number of organizational entities (for example, say in the millions) it may take a long time to generate as the migration script needed to traverse up the hierarchy tree. You may consider designing your own fillorgDN.sql script based on your knowledge of how your organizational entity hierarchy is organized.

Ensure that you place your custom fillorgDN.sql script in the *WC55_userdir/instances/instance_name/migration/DB2* directory. This script gets used after the data migration.

The content of the `fillorgDN.sql` is typically as shown below. For each organizational entity (ORGENCY) update the DN so that it contains all the grand parent and parent organizations. For example:

```
update orgentity set dn='o=Root Organization' where orgentity_id=-2001;
update orgentity set dn='o=Default Organization,o=Root Organization'
  where orgentity_id=-2000;
update orgentity set dn='o=MyGrandParentOrg,o=Default Organization,o=Root Organization'
  where orgentity_id=3455;
update orgentity
  set dn='o=MyParentOrg,o=MyGrandParentOrg,o=Default Organization,o=Root Organization'
  where orgentity_id=3456;
```

If you decide to write your own `fillorgDN.sql` script, you can skip the generation of `fillorgDN.sql` by the data migration script. To do this, you need to place the `fillorgDN.sql` file (which can be empty) in the `WC55_installdir/instances/instance_name` directory. You can then fill in the content of the `fillorgDN.sql` script using the above example for guidance.

Transitioning to WebSphere Application Server 5.0

IBM WebSphere Application Server, Version 5, offers a world-class infrastructure for the next chapter in open e-business platforms. As the foundation of the WebSphere software platform, WebSphere Application Server provides a rich, e-business application deployment environment with a complete set of application services including capabilities for transaction management, security, clustering, performance, availability, connectivity and scalability. Version 5 offer full J2EE specification support (Servlet 2.3, JSP 1.2, EJB 2.0, and others), as well as a variety of extensions.

For details on the new features introduced in WebSphere Application Server 5.0, check the WebSphere Application Server for iSeries Web site (<http://www.ibm.com/servers/eserver/iseries/software/websphere/wsappserver/>). Documentation for WebSphere Application Server 5.0 including migration information is available at this site.

Chapter 2. Backing up Commerce Suite 5.1

Before you proceed with the migration process documented in this Guide, you should perform a complete system backup of your working Commerce Suite 5.1 system. This will enable you to recover your previous system in the event that you encounter problems while migrating to WebSphere Commerce 5.5.

If you are migrating *in-place*, you cannot revert to your previous version of Commerce Suite 5.1 after you complete the migration to WebSphere Commerce 5.5.

Backing up your Commerce Suite 5.1 system

To back up your Commerce Suite 5.1 system:

- Perform a complete system backup of your Commerce Suite 5.1 system in accordance with the documentation provided with your operating system or any specialized backup and restore software you are using. Typically, you can backup your system to a tape drive, a ZIP drive, or some other file system.
- Back up any customized files and directories used for Commerce Suite 5.1 and its associated components such as the database, the Web server, WebSphere Commerce Payments (formerly WebSphere Payment Manager) and WebSphere Application Server.
- Back up your Commerce Suite 5.1 database as described in “Backing up your databases” on page 19.
- In particular, backup the main Commerce Suite 5.1 installation directory including all underlying subdirectories and files to a location that is easily accessible during the migration process as you may need to refer to these directories and files at some point in the migration process. In particular, back up the instances and *your_instance* subdirectories under *WCS51_userdir* (/QIBM/UserData/CommerceSuite5).

Backing up directories and files

To manually back up any of the following directories or files:

1. From a command prompt, create a temporary backup directory.
2. Switch to the Commerce Suite 5.1 user installation directory (*WCS51_userdir*).
3. Select the appropriate directories or files and copy them to the temporary backup directory. Alternatively, use the Save Object (SAV) and Restore Object (RST) commands to save and restore files and directories:

```
SAV DEV('save_file') OBJ(('IFS_folder'))
RST DEV('save_file') OBJ(('IFS_folder'))
```

For example, the *WCS51_userdir/instances/myinstance* folder can be backed up in a save file (*myinstsav*) in the MYLIB library with the following command (on one line):

```
SAV DEV('/QSYS.LIB/MYLIB.LIB/MYINSTSAV.FILE')
    OBJ(('WCS51_userdir/instances/your_instance'))
```

The folder and its contents can be restored back with the following command (on one line):

```
RST DEV('/QSYS.LIB/MYLIB.LIB/MYINSTSAV.FILE')
    OBJ(('WCS51_userdir/instances/your_instance'))
```

In particular, you should backup the following directories and files:

- The following important Commerce Suite 5.1 files and directories under the *WCS51_installdir*. Note that the WCIM instance migration tool backs up the following files and directories when you run it in “Migrating your instance configuration using WCIM” on page 31, however it is still recommended that you back these files up to a location that you can access:
 - web/*
 - stores/*
 - properties/*
 - xml/product.*
 - PaymentManager/profile.dtd
 - bin/cfg.passwd
- The following important Commerce Suite 5.1 files and directories under the *WCS51_userdir* directory:
 - instances/wcs_instances
 - instances/*instance_name*/* (Instance root directory and all files and subfolders in it.)
- The following HTTP files:
 - HTTP server instance file:
/QSYS.LIB/QUSRSYS.LIB/QATMHINSTC.FILE/*instance_name*.MBR
 - HTTP server configuration file:
/QSYS.LIB/QUSRSYS.LIB/QATMHTTPC.FILE/*instance_name*.MBR
- The rules projects for campaigns. These files are generated when you publish a campaign in Commerce Suite 5.1. They are named after the campaign, but have a number of different extensions as follows:
 - *campaign_name*.adv
 - *campaign_name*.cdd
 - *campaign_name*.dbcp
 - *campaign_name*.flow0
 - *campaign_name*.flow1
 - *campaign_name*.jcp
 - *campaign_name*.rb

Once you start to run the campaign code in WebSphere Commerce 5.5, WebSphere Commerce will look at these files and store them persistently into the new WebSphere Commerce schema tables. Once the data gets transferred from these files to the schema, the campaign code will start to look at the database instead of the files. The files will eventually become obsolete when the campaign is run to completion in WebSphere Commerce 5.5.

- Any customized code that you need to backup. Customized code is not backed up by the WCIM tool. For further details on custom code migration, refer to the *WebSphere Commerce Studio Migration Guide*.
- Other miscellaneous files:
 - Any file-based content such as static HTML pages and GIF files.
 - Any database .sql scripts.
 - Any JavaServer Pages (JSP files).
 - Any customized commands and files (for example, .java, .class, .jar, .zip, or .properties files).
 - Any customized document files (for example, .pdf or text files).

Backing up your databases

The following sections describe how to backup your database.

To backup your database, perform the following actions on the machine where your WebSphere Commerce database resides:

1. Log on using a user profile with *SECOFR access rights
2. Create a save file using the CRTSAVF command.
3. Stop all database operations.
4. Use the SAVLIB command to save your schema library to the savefile you just created.

For more information on backing up libraries, and the commands used to do so, see the manuals that come with your Operating system.

Chapter 3. Upgrading your software

This chapter describes how to upgrade your software to the levels required by WebSphere Commerce 5.5. Prior to upgrading your software, you should backup your Commerce Suite 5.1 system, including the database. See Chapter 2, “Backing up Commerce Suite 5.1,” on page 17 on how to perform a system backup.

If you need to stop any of your Commerce Suite 5.1 software components, see the appendix of the appropriate Installation Guide for your product and platform for the detailed steps.

Mapping the Commerce Suite 5.1 to WebSphere Commerce 5.5 IBM software

The following table maps the version levels and the installation paths between Commerce Suite 5.1 and WebSphere Commerce 5.5 for most of the software packaged with those versions.

Table 4. Supplied software levels and default installation paths for WebSphere Commerce for iSeries

Software	WebSphere Commerce Suite 5.1	WebSphere Commerce 5.5
WebSphere Commerce	Version 5.1 /QIBM/ProdData/CommerceSuite5	Version 5.5 /QIBM/ProdData/CommerceServer55
IBM WebSphere Application Server — Advanced Edition	Version 3.5.2 (Version 3.5 plus FixPak 2 and eFixes) /QIBM/ProdData/WebASAdv	Version 5.0 /QIBM/ProdData/WebAS5/base
IBM SDK for Java	Version 1.2.2 /QIBM/ProdData/Java400/jdk12	Version 1.3.1 /QIBM/ProdData/Java400/jdk13
WebSphere Commerce Payments (Formerly IBM WebSphere Payment Manager)	Version 2.2 /QIBM/ProdData/PymSvr	Version 5.5 /QIBM/ProdData/CommercePayments/V55
IBM WebSphere Commerce Analyzer	Version 5.1	Version 5.5
IBM SecureWay® Directory Server	Version 3.2.1	Version 3.2.2
Blaze Advisor™ Rule Server	Version 3.1.2 /QIBM/ProdData/CommerceSuite5/blaze	Version 4.5.5 /QIBM/ProdData/CommerceServer55/blaze
Macromedia LikeMinds Personalization Server	Version 5.1	Version 5.5

Approach to upgrading the software

It is generally recommended that you install WebSphere Commerce 5.5 including the prerequisite software on a *clean* machine (that is, a machine which does not have any of the WebSphere Commerce 5.5 software components, or previous versions of them installed). This *clean* machine should be remote from your Commerce Suite 5.1 machine, that is, it should be a separate machine. It would be helpful in the migration process if the two machines are connected over the

network so that assets can be easily transferred between them. Note that you can set up your WebSphere Commerce 5.5 system according to any of the supported topologies:

- 1-tier (All WebSphere Commerce components are on the same machine)
- 2-tier (The database is on a remote machine relative to WebSphere Commerce)
- 3-tier (The database and the Web server are on remote machines relative to WebSphere Commerce)
- In addition, WebSphere Commerce Payments can be installed on a separate machine from your WebSphere Commerce machine.

This approach minimizes the downtime to migrate compared to an *in-place* migration where you upgrade the software components on your Commerce Suite 5.1 machine. However, the approach does require additional machine resources. With this approach, you can keep your Commerce Suite 5.1 system running until your WebSphere Commerce system is fully operational, and all required assets are transferred, migrated, and fully tested. At this point, you can shut down your Commerce Suite 5.1 system and start up your WebSphere Commerce 5.5 system, thereby switching over the machines. You can redeploy the Commerce Suite 5.1 machine for other uses.

- If you wish to do a *remote* migration, that is, a migration where you install WebSphere Commerce 5.5 on a new machine, refer to “Installing WebSphere Commerce 5.5 on a separate machine (remote migration).”
- If you wish to do a migration *in-place*, that is, upgrade your existing Commerce Suite 5.1 system to WebSphere Commerce 5.5 without requiring additional machine resources, see “Upgrading your existing system to WebSphere Commerce 5.5 (in-place migration)” on page 25.



This Guide describes two methods of implementing the migration process:

- *In-place* migration - which requires a single machine
- *Remote* migration - which requires two machines.

While the in-place method is convenient since it requires only one machine, most users would likely find the downtime for their system using this method to be excessive, particularly for a production site. The complete migration process time duration consists of two major items:

- Database migration - which could take many hours, depending on the size of the database (number of users, products, orders, user traffic, and so on).
- Code migration - where code customization and testing may take a number of days.

Since the live Commerce Suite 5.1 site has to be shut down as soon as the migration effort starts, the *remote method is recommended over the in-place method for live sites where downtime is of concern*. The in-place method is suitable however for customization and testing of the migration process on a site that is not live.

Installing WebSphere Commerce 5.5 on a separate machine (remote migration)

Install and setup your WebSphere Commerce 5.5 system on a machine that is separate from Commerce Suite 5.1. The target machine must be one of the supported WebSphere Commerce 5.5 platforms with the prerequisite amount of memory, CPU-type, and disk space. For information on these pre-requirements, see

the *WebSphere Commerce Installation Guide* for your platform. For information on installing the WebSphere Commerce 5.5 system, see the chapters on installation in the *WebSphere Commerce Installation Guide* for your platform.

Important

After you install WebSphere Commerce 5.5, you need to install the IBM WebSphere Commerce 5.5.0.2 fix pack. This fix pack includes fixes and enhancements for the migration process. This fix pack is available from the WebSphere Commerce Support site (<http://www.ibm.com/software/genservers/commerce/support/>). Click on the link for the edition you are using and see the section **Download**. Download this fix pack, and follow the instructions that are provided to install it. You should not proceed with the migration steps in this Guide (such as migrating your instance or database) without installing this fix pack.

Note: In order to successfully apply all WebSphere Application Server interim fixes during the WebSphere Commerce 5.5 installation process, you should ensure that your Web server is shut down. In addition, after the WebSphere Commerce 5.5 installation is complete, check the `wcWASeFixer.log` file and ensure that all WebSphere Application Server interim fixes were successfully installed.

Additional machine requirements for transition

In addition to the pre-requirements documented in the *WebSphere Commerce Installation Guide*, you need to meet the following additional requirement for migration:

For a single-tier configuration, the target machine hard disk space requirement should be at least twice the current Commerce Suite 5.1 database size in addition to the WebSphere Commerce 5.5 installation hard disk requirement. The guideline is based on the following requirements:

- Space to hold a backup copy of the Commerce Suite 5.1 database.
- Space to hold a backup copy of the WebSphere Commerce 5.5 database at various stage of the migration. It is estimated that the final size of the WebSphere Commerce 5.5 database will not be larger than double the current size of the Commerce Suite 5.1 database.
- Space for logs.

Creating a WebSphere Commerce test instance

After you have successfully installed and verified the WebSphere Commerce 5.5 software as documented in the *WebSphere Commerce Installation Guide*, you should create a WebSphere Commerce test instance on your WebSphere Commerce 5.5 system.

To create a test instance, log on to the WebSphere Commerce Configuration Manager interface.

For information on the steps to create a WebSphere Commerce instance using Configuration Manager, see the configuration section of the *WebSphere Commerce Installation Guide* for your database and platform.

You can create a test store to verify the functionality of the WebSphere Commerce 5.5 installation. If you intend to retain the test instance, ensure that you choose an instance name that is different from the one that you are migrating from Commerce Suite 5.1, to avoid any possible conflicts. Once you are satisfied that the WebSphere Commerce 5.5 system is functional, you can delete your test instance and test database. See the *WebSphere Commerce Installation Guide* for information on deleting a WebSphere Commerce 5.5 instance.

Migrating your previous WebSphere Application Server configuration

In order to migrate your previous WebSphere Application Server administrative configurations to the new WebSphere Commerce 5.5 system, you need to use the WASPreUpgrade and WASPostUpgrade tools.

1. For a remote migration, you have to install WebSphere Application Server 5.0 on the previous Commerce Suite 5.1 machine in order to get access to the WASPreUpgrade and WASPostUpgrade scripts.
2. Run the WASPreUpgrade tool on your previous WebSphere Application Server machine to preserve the previous administrative configurations to a backup directory. The WASPreUpgrade tool is available in the *WAS50_installdir/bin* directory. The syntax to run this tool is described in the WebSphere Application Server 5.0 for iSeries migration site at:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>



Due to the time spent on importing and deploying EJBs, you may experience a performance problem with the instance migration to WebSphere Commerce 5.5. To improve instance migration performance, you need to modify the *websphere_backup.xml* file and remove the EJB-related section after you have run the WASPreUpgrade tool as follows:

- a. After running WASPreupgrade, locate the file *websphere_backup.xml* and make a backup copy of it, and then open the file in an editor.
- b. Locate the `<container name="WCS EJB Container" action="update">` entry and remove the entire node:

```
<container name="WCS EJB Container" action="update">
...
</container>
```

This will prevent the EJBs from being imported or deployed during instance migration.

3. Transfer the complete backup directory of your previous administrative configurations (including all files and subdirectories) from your previous Commerce Suite 5.1 machine to your WebSphere Commerce 5.5 machine.
4. After transferring the backup directory, edit the *websphere_backup.xml* file (that is located in the backup directory on the WebSphere Commerce 5.5 machine). Search for all occurrences of the Commerce Suite 5.1 machine's node name, IP address, and domain, and replace them with the WebSphere Commerce 5.5 machine's node name IP address, and domain respectively.
5. Before running WASPostUpgrade tool, ensure that the `<alias-list>` section in the *websphere_backup.xml* file correctly reflects the new machine environment.
6. Run the WASPostUpgrade tool on your WebSphere Application Server 5.0 machine to migrate the previous configuration (stored in the backup directory) to the new installation. The WASPostUpgrade tool is available in the *WAS50_installdir/bin* directory.

The syntax to run this tool is described in the WebSphere Application Server 5.0 for iSeries migration site at:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>

Check the `WASPostUpgrade.log` file located in the `WAS50_installdir/logs` directory. You can ignore any messages related to missing EJB files since they will not be available on the WebSphere Commerce 5.5 machine.

7. Logon to the WebSphere Application Server Administration Console and uninstall the WebSphere Commerce Server_ *instance_name* MigratedApp application that gets created by WASPostUpgrade from WebSphere Application Server 5.0, if it exists. It is not required in the migration process.

WebSphere Application Server JDBC Provider level consideration

The default behavior of WASPostUpgrade is to maintain the previous scope of the JDBC Provider level of WebSphere Application Server 3.5.x when migrating to WebSphere Application Server 5.0. This means that the JDBC provider scope will be kept as global as this the default scope in WebSphere Application Server 3.5.x. In WebSphere Application Server 5.0, the default for the JDBC provider scope is under the server level as opposed to global.

Next Step

Proceed to Chapter 4, “Migrating your Commerce instance configuration,” on page 29.

Upgrading your existing system to WebSphere Commerce 5.5 (in-place migration)

This section describes how to upgrade the software for your Commerce Suite 5.1 machine *in-place*, and describes:

- “Upgrading your hardware”
- “Upgrading your operating system”
- “Upgrading additional software components”
- “Upgrading to WebSphere Application Server 5.0” on page 26
- “Upgrading to WebSphere Commerce 5.5” on page 27

Upgrading your hardware

For information on upgrading your hardware, see the pre-installation section of the *WebSphere Commerce Installation Guide*.

Upgrading your operating system

Ensure that your system is running OS/400 Version 5 Release 2.

Upgrading additional software components

This section describes how to upgrade additional software components that are supported by Commerce Suite 5.1, such as the Web browser.

Internet Explorer 6.0 or later

You can only access the WebSphere Commerce tools and online help using Microsoft® Internet Explorer 6.0 from a machine running a Windows operating system on the same network as your WebSphere Commerce machine.

You can download Internet Explorer from the Microsoft Corporation Download Web site (<http://www.microsoft.com/downloads/>)

Shoppers can access Web sites by using any of the following Web browsers, all of which have been tested with WebSphere Commerce:

Upgrading to WebSphere Application Server 5.0

The step-by-step migration from your previous version of WebSphere Application Server is described at the WebSphere Application Server 5.0 for iSeries migration site at:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>

Manually migrating your previous WebSphere Application Server configuration

In order to migrate your previous WebSphere Application Server administrative configurations to the new WebSphere Commerce 5.5 system, you need to use the WASPreUpgrade and WASPostUpgrade tools as follows:

1. Install WebSphere Application Server 5.0.x in accordance with WebSphere Application Server installation instructions.
2. Run the WASPreUpgrade tool to preserve the previous administrative configurations to a backup directory. The WASPreUpgrade tool is available in the *WAS50_installdir/bin* directory. The syntax to run this tool is described in the WebSphere Application Server 5.0 for iSeries migration site at:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>



3. Due to the time spent on importing and deploying EJBs, you can experience a performance problem with the instance migration to WebSphere Commerce 5.5. To improve instance migration performance, you need to modify the *websphere_backup.xml* file and remove the EJB-related section after you have run the WASPreUpgrade tool as follows:

- a. After running WASPreupgrade, locate the file *websphere_backup.xml* and make a backup copy of it, and then open the file in an editor.
- b. Locate the `<container name="WCS EJB Container" action="update">` entry and remove the entire node:

```
<container name="WCS EJB Container" action="update">
...
</container>
```

- c. Run the WASPostupgrade utility, as described in the following steps.

This will prevent the EJBs from being imported or deployed during instance migration.

4. Before running WASPostUpgrade tool, ensure that the `<alias-list>` section in the *websphere_backup.xml* file correctly reflects the new machine environment.
5. Run the WASPostUpgrade tool on your WebSphere Application Server 5.0 machine to migrate the previous configuration (stored in the backup directory) to the new installation. The WASPostUpgrade tool is available in the *WAS50_installdir/bin* directory.

The syntax to run this tool is described in the WebSphere Application Server 5.0 for iSeries migration site at:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>

6. Uninstall the *instance_name_-_WebSphere_Commerce_Server_MigratedApp.ear* file that gets created by WASPostUpgrade from WebSphere Application Server

5.0, if it exists. You can normally find this file in the `WAS50_userdir/installedApps/cell_name` directory

WebSphere Application Server JDBC Provider level consideration: The default behavior of WASPostUpgrade is to maintain the previous scope of the JDBC Provider level of WebSphere Application Server 3.5.x when migrating to WebSphere Application Server 5.0. This means that the JDBC provider scope will be kept as global as this the default scope in WebSphere Application Server 3.5.x. In WebSphere Application Server 5.0, the default for the JDBC provider scope is under the server level as opposed to global.

Upgrading to WebSphere Commerce 5.5

Installing WebSphere Commerce 5.5 on the same system does not upgrade the Commerce Suite 5.1 installation. The two installations can coexist. However, you must upgrade your system to OS/400 V5R2 before installing WebSphere Commerce 5.5. Since IBM WebSphere Application Server 3.5 is not supported on OS/400 V5R2, the commerce instance can no longer be used at the Commerce Suite 5.1 level. Before beginning a migration, you must stop all operations against that instance. For information on these pre-requirements, see the *WebSphere Commerce Installation Guide*.

For information on installing the WebSphere Commerce 5.5 system, see the chapters on installation in the *WebSphere Commerce Installation Guide* for your platform.

Notes:

1. Before you upgrade to WebSphere Commerce 5.5, ensure that all message retry counters are set down to zero as documented in “Ensuring no new messages are stored in MSGSTORE table” on page 4.
2. Do not delete your previous Commerce Suite 5.1 installation tree until after you have completed and verified the migration.

Important

After you install WebSphere Commerce 5.5, you need to install the IBM WebSphere Commerce 5.5.0.2 fix pack. This fix pack includes fixes and enhancements for the migration process. This fix pack is available from the WebSphere Commerce Support site (<http://www.ibm.com/software/genservers/commerce/support/>). Click on the link for the edition you are using and see the section **Download**. Download this fix pack, and follow the instructions that are provided to install it. You should not proceed with the migration steps in this Guide (such as migrating your instance or database) without installing this fix pack.

Next Step

Proceed to Chapter 4, “Migrating your Commerce instance configuration,” on page 29.

Chapter 4. Migrating your Commerce instance configuration

This section describes how to migrate your Commerce Suite 5.1 instance configuration to a WebSphere Commerce 5.5 instance configuration. For information on how to create a new WebSphere Commerce 5.5 instance, refer to the section "Create or modify an instance with Configuration Manager" in the *WebSphere Commerce Installation Guide* for your database and platform.

The Commerce Suite 5.1 instance migration is performed by starting from a base WebSphere Commerce 5.5 instance (EAR) and adding in the required components from the Commerce Suite 5.1 instance. As there are some changes in the instance packaging between Commerce Suite 5.1 and WebSphere Commerce 5.5, the instance migration tool places the instance files into the appropriate module within the WebSphere Commerce 5.5 EAR, and updates the instance files, configuration files and deployment descriptors as necessary.

The WCIM (WebSphere Commerce Instance Migrator) tool is provided to perform the WebSphere Commerce instance migration step.

Copying the Commerce Suite 5.1 product.xml file

The WebSphere Commerce Suite 5.1 Pro Edition did not have a `product.xml` file. WebSphere Commerce 5.5 provides a prebuilt `product.xml` .51.pro file, in the `WC55_installdir/migration` directory.

- Copy the `product.xml` .51.pro file to the `WCS51_userdir/xml` directory. (For a remote migration, this needs to be copied to the remote Commerce Suite 5.1 machine.)
- Rename the file you copied over to `product.xml`. (The WCIM tool requires the `product.xml` file for migration, prior to performing a backup.)
- Update the following line in the `product.xml` file that you copied over:

```
<path>$COMMERCE_INSTALL_PATH</path>
```

with the `WCS51_installdir` installation path:

```
<path>WCS51_installdir</path>
```

- Finally, copy the `product.dtd` file from the `WC55_installdir/xml` directory to the `WCS51_userdir/xml` directory. (For a remote migration, this needs to be copied to the remote Commerce Suite 5.1 machine.)

Updating system values in the WebSphere Commerce 5.5 product.xml file

You need to fill in the correct host name and domain of your system into the WebSphere Commerce 5.5 `product.xml` file located in the `WC55_installdir/xml` directory. The host name and domain have to be in the same case as it is on your system. Locate and update the `host_name` and `domain` in the `<system>` section of the `product.xml` as follows:

```
<system>
  <OS>
    <type>OS400</type>
    <edition></edition>
    <version>V5R2M0</version>
  </OS>
```



```

        <RAM></RAM>
        <hostname>host_name</hostname>
        <domain>domain</domain>
</system>

```

In addition, you need to update the following line in the WebSphere Commerce 5.5 `product.xml` file:

```
<configInstanceList>/qibm/ProdData/CommerceServer55/instances/wcs_instances</configInstanceList>
```

This line should reference `UserData` instead of `ProdData` as follows:

```
<configInstanceList>/qibm/UserData/CommerceServer55/instances/wcs_instances</configInstanceList>
```

Checking the WebSphere Commerce 5.5 `product.xml` file (in-place migration only)

You should check the `<migrationFrom>` section of the `product.xml` file on the WebSphere Commerce 5.5 machine and ensure that all the values are properly set for your migration path prior to running WCIM. The `product.xml` file is located in the `WC55_installdir/xml` directory.

Verify that the values are correctly set in the `<migrationFrom>` section in the `product.xml` file as follows. (The `<migrationFrom>` section specifies information for your previous machine, from where you are migrating.)

The following is an example of what your `product.xml` file needs to contain if you are migrating from WebSphere Commerce Suite 5.1.0.1 Pro Edition:

```

<migrationFrom>
  <edition>
    <name>Pro</name>
  </edition>
  <version>5</version>
  <release>1</release>
  <modification>0.1</modification>
  <fixpak>0</fixpak>
  <path>/QIBM/ProdData/CommerceSuite5</path>
  <altpath>/QIBM/ProdData/WebCommerce</altpath>
</migrationFrom>

```

Steps before migrating your instance configuration

Before migrating your Commerce Suite 5.1 instance to WebSphere Commerce 5.5:

- The WCIM tool will back up the entire directory structure for your instance. If you have Commerce Suite 5.1 customized files and directories in other locations outside of the instance directory, you must now copy these files and directories from these other locations to a directory under the instance root directory structure. This way, WCIM will restore your customized files and directories for you. For details on custom code migration, refer to the *WebSphere Commerce Studio Migration Guide*.

To restore the files and directories from the other locations, do the following:

1. Open a command window and navigate to the directory that contains the customized files, or to the custom directory.
 2. Select the appropriate files or directories and copy them to the appropriate Commerce Suite 5.1 directory.
- If WebSphere Application Server security is turned on, you must disable it before migrating your instance as follows:
 1. Open the WebSphere Application Server Administration console.

2. Click **Console > Security Center** and clear the **Enable Security** check box on the General tab.
 3. Click **Finish**.
 4. Restart the WebSphere Application Server administrative server.
- Updating the WebSphere Commerce administration tools ports (optional)
In WebSphere Commerce 5.5, the administration tools run on separate ports than the WebSphere Commerce server. Instance migration, assigns the following default port numbers for these administration tools:

Port number

WebSphere Commerce tool

- 8000 WebSphere Commerce Accelerator
- 8002 WebSphere Commerce Administration Console
- 8004 WebSphere Commerce Organization Administration Console

If you wish to use non-default ports for the above tools, add the following three lines in the <Websphere> node under the <InstanceProperties> node in the *instance_name.xml* file. This file is located in the *WC55_userdir/instances/your_instance/xml* directory. The following lines override the default settings and use a non-default port for these administration tools:

```
ToolsPort = "desired_tools_port_number"
AdminPort="desired_admin_port_number"
OrgAdminPort="desired_org_admin_port_number"
```

For example:

```
<Websphere Name="WebSphere Commerce DB2 DataSource"
  JDBCDriverLocation=""
  StoresWebApp="true"
  ToolsWebApp="true"
  WCMWebApp="true"
  name="WebSphere"
  ToolsPort = "9000"
  AdminPort="9001"
  OrgAdminPort="9002"
  port="900"/>
```

Migrating your instance configuration using WCIM

This section describes how you run WCIM to migrate your previous instance of WebSphere Commerce. For details on what WCIM does, see “Instance migration using WCIM” on page 129

The WCIM package provides two main scripts in the *WC55_installdir/bin* directory:

- *wcimenv.sh* — Sets up the environmental settings for *wcim* to run properly on your system.
- *wcim.sh*— Performs the actual instance migration.

The basic process to run *wcim* is as follows:

1. Update the *wcimenv.sh* file for backup of your instance. For a remote migration, this is done on the Commerce Suite 5.1 system. For an in-place migration, this is done on your Commerce Suite 5.1 system that has been upgraded to WebSphere Commerce 5.5.
2. Run *wcim* for backup.

3. Update the `wcimenv.sh` file for migration of your instance. For a remote migration, this is done on the WebSphere Commerce 5.5 system. For an in-place migration, this is done on your Commerce Suite 5.1 system that has been upgraded to WebSphere Commerce 5.5.
4. Run `wcim` for migration.

The details on running the scripts are contained in the following sections.

Note: The default behavior of `wcim.sh` is to read the environment file (`wcimenv.sh`) from the same directory where you run `wcim.sh` (that is, the `WC55_installdir/bin` directory). However, since the default values of `wcimenv.sh` may not be correct for every users's system, it is recommended that you copy `wcimenv.sh` from the `WC55_installdir/bin` directory to a temporary directory (such as `/tmp`) in order to make changes to the file. When you run the `wcim.sh` script, you need to specify the full path to where the `wcimenv.sh` file is located. For example:

```
./wcim.sh path_to_updated_wcimenv.sh
```

Prerequisites for running WCIM

Before running the WCIM tool to migrate your instance, you must ensure the following:

- WebSphere Application Server 5.0 is installed on the machine where you are running WCIM.
For an *in-place* migration, it needs to be installed on the WebSphere Commerce 5.5 machine.
For a *remote* migration, it needs to be installed on both the WebSphere Commerce 5.5 and the Commerce Suite 5.1 machines. In a remote migration, WebSphere Application Server 5.0 needs to be uninstalled from the Commerce Suite 5.1 machine after you have successfully completed the instance migration.
- The WebSphere Commerce software components including the Web server have been upgraded to the WebSphere Commerce 5.5 level
- All installed WebSphere Commerce applications have been migrated successfully by WebSphere Application Server 5.0.
- Before running the WCIM utility to migrate your instance, ensure that the user ID that runs the utility has full permission to access files and subfolders in the following directories:
 - `WC55_userdir/instances`
This directory will not exist if a WebSphere Commerce 5.5 instance has not been previously created, that is, if you have not created a test instance after installing WebSphere Commerce 5.5, as recommended. If the directory does not exist, the WCIM tool will create it.
 - `WC51_userdir/instances`
 - A migration working directory (for example, `WC55_userdir/temp`).

Notes:

1. When you run the `wcim.sh` script with the `WCIM_BACKUP` variable in `wcimenv.sh` set to true, WCIM generates a backup ZIP file of your previous instance by passing the required parameters to the script. The ZIP file, `wcbakup51.zip`, is generated in the `WORK_DIR/zip` directory, where `WORK_DIR` is a user-defined working directory in the `wcimenv.sh` file. For example, you can define the working directory, to `WC55_userdir/temp/` (see Table 5 on page 33 where this variable is defined).

2. If your Commerce Suite 5.1 machine is remote from your WebSphere Commerce 5.5 machine, you have to manually copy the ZIP file, `wcbackup51.zip`, into the `WORK_DIR/zip` directory (`WC55_userdir/temp/zip`) on the WebSphere Commerce 5.5 machine. If the migration is *in-place*, and if you set `WORK_DIR` to the same value in the backup and migration steps, this step to copy the ZIP file is not required.

Migrating a Commerce Suite 5.1 instance

Variables you specify in `wcimenv`

Prior to running `wcim.sh` for migrating your instance, you need to set up the environment variables correctly in `wcimenv.sh` for your environment. In addition, you need to set the variables differently depending on whether you are performing a backup of your instance and instance related file assets, or actually migrating them.

The following table summarizes the variables and their values:

Table 5. Variables set in `wcimenv.sh`

Variable	Value	Description
WCIM_BACKUP	true	The WCIM tool will <i>backup</i> your WebSphere Commerce instance related file assets.
	false	The WCIM tool will <i>migrate</i> your WebSphere Commerce instance and instance related file assets.
WCIM_INPLACE	true	Always set <code>WCIM_INPLACE=true</code> (except when <code>WCIM_BACKUP=true</code> and the migrated instance will be moved to a different machine — remote machine)
	false	Set <code>WCIM_INPLACE=false</code> <i>only</i> when <code>WCIM_BACKUP=true</code> and the migrated instance will be moved to a different machine — remote migration.



The above `WCIM_BACKUP` and `WCIM_INPLACE` variables are used in the following combinations:

- For backup:
 - `WCIM_BACKUP` is always set to true.
 - `WCIM_INPLACE = true` when Commerce Suite 5.1 is on the same machine as WebSphere Commerce 5.5 (in-place).
 - `WCIM_INPLACE = false` when Commerce Suite 5.1 is not on the same machine as WebSphere Commerce 5.5 (remote).
- For migration:
 - `WCIM_BACKUP` is always set to false.
 - `WCIM_INPLACE = true` when Commerce Suite 5.1 is on the same machine as WebSphere Commerce 5.5 (in-place).
 - `WCIM_INPLACE = true` when Commerce Suite 5.1 is not on the same machine as WebSphere Commerce 5.5 (remote).

Table 5. Variables set in *wcimenv.sh* (continued)

Variable	Value	Description
WCIM_MIGRATE_FROM	51	Backup or migrate from Commerce Suite 5.1
	WPM221	Backup or migrate from Payment Manager 2.2.1. See “Migrating your Payments instance and database” on page 83 for details.
	WPM312	Backup or migrate from Payment Manager 3.1.2. See “Migrating your Payments instance and database” on page 83 for details.
	WCP313	Backup or migrate from WebSphere Commerce Payments 3.1.3. See “Migrating your Payments instance and database” on page 83 for details.
DB_TYPE	db2	The database type for DB2 databases.
WAS_PATH	<i>WAS35_installdir</i> or <i>WAS50_installdir</i>	The WebSphere Application Server installation path: <ul style="list-style-type: none"> For backups (where <i>WCIM_BACKUP</i> is true), specify the previous WebSphere Application Server installation path. For migration (where <i>WCIM_BACKUP</i> is false), specify the WebSphere Application Server 5.0 installation path.
JAVA_EXE	java	The SDK for Java executable.
WC_PATH	<i>WCS51_installdir</i> or <i>WC55_installdir</i>	The WebSphere Commerce installation path: <ul style="list-style-type: none"> For backups (where <i>WCIM_BACKUP</i> is true), specify the Commerce Suite 5.1 installation path. For migration (where <i>WCIM_BACKUP</i> is false), specify the WebSphere Commerce 5.5 installation path.
WEBSERVER_PATH	<ul style="list-style-type: none"> For IBM HTTP Server: <i>HTTPServer1312_installdir</i> or <i>HTTPServer1326_installdir</i> 	<p>The Web server installation path:</p> <ul style="list-style-type: none"> For backups (where <i>WCIM_BACKUP</i> is true), specify the previous Web server path. For migration (where <i>WCIM_BACKUP</i> is false), specify the current Web server installation path. <p>Note: <i>WEBSERVER_PATH</i> referenced here is the production Web server path (QIBM/ProdData/HTTP) as opposed to the user Web server path (QIBM/UserData/HTTP). It is the same value for both <i>WCIM</i> backup and <i>WCIM</i> migration.</p> <p>Note: If you are migrating from Payment Manager 2.1.2, and specify <i>PM221</i> for <i>WCIM_MIGRATE_FROM</i>, you need to specify the <i>HTTPServer1312_installdir/htdocs</i> directory.</p>
ANT_PATH	<i>Ant_installdir/lib</i>	The ANT library path on your machine. You can set this to <i>WAS50_installdir/lib</i> since ANT 1.4.1 is shipped with WebSphere Application Server 5.0.
WORK_DIR	user-defined	Working directory for <i>WCIM</i> tool. (For example, <i>WC55_userdir/temp</i>)

Table 5. Variables set in `wcimenv.sh` (continued)

Variable	Value	Description
MIG_FROM_WAS_INSTANCE		The previous WebSphere Application Server instance with which the Commerce instance was originally associated.
MIG_TO_WAS_INSTANCE		The WebSphere Application Server instance that the Commerce instance is to be associated with.
WPM_PATH	<i>Payments_installdir</i>	This variable is used when using WCIM to migrate your previous version of WebSphere Commerce Payments. The process to migrate Payments is described in detail in “Migrating your Payments instance and database” on page 83.
LOG_FILE	user-defined	Log file for WCIM. You can specify any file name under any directory. If no directory is specified, the log file is generated in the working directory specified by <code>WORK_DIR</code> . For example: <ul style="list-style-type: none"> • For backups, <code>wcimbackup.log</code> • For migration, <code>wcimmigration.log</code>
INSTANCE	Commerce Suite 5.1 <i>instance_name</i>	The name of the instance that WCIM will either migrate, or whose file assets WCIM will backup.

Running WCIM for remote instance migration

The following sections describe the steps for instance migration when WebSphere Commerce 5.5 is installed on a separate machine from Commerce Suite 5.1.

Pre-requirements for remote instance migration: If you are installing your WebSphere Commerce 5.5 software components on a machine that is remote from Commerce Suite 5.1, you should have completed the steps described in “Migrating your previous WebSphere Application Server configuration” on page 24.

Ensure WebSphere Application Server 5.0 is installed on both the WebSphere Commerce 5.5 and the Commerce Suite 5.1 machines. WebSphere Application Server 5.0 needs to be uninstalled from the Commerce Suite 5.1 machine after you have successfully completed the instance migration.

Running WCIM for remote backup: Before running WCIM for migration, you should run WCIM to create a backup of your current instance and instance related files:

1. Create a temporary work directory on your Commerce Suite 5.1 machine, for example `WC51_userdir/temp`. In addition, you should also create the following subdirectories under the temp working directory:
 - `lib`
 - `bin`
 - `xml/migration`
2. Copy the following files:
 - From the `WC55_installdir/lib` directory on your WebSphere Commerce 5.5 machine to the `lib` subdirectory under your work directory on your remote Commerce Suite 5.1 machine (`WC51_userdir/temp/lib`):
 - `Utilities.jar`
 - `ConfigManager.jar`

- xerces.jar
- From the *WC55_installdir/xml/migration* directory on your WebSphere Commerce 5.5 machine, copy the *instbackup51.xml* file to the *xml/migration* subdirectory under the working directory on your Commerce Suite 5.1 machine (*WCS51_userdir/temp/xml/migration*).

Note: If migrating Payments, instead of copying *instbackup51.xml*, copy either *instbackupwpm221.xml*, *instbackupwpm312.xml*, or *instbackupwcp313.xml*, depending on the Payments version of the instance are migrating.

- From the *WC55_installdir/bin* directory on your WebSphere Commerce 5.5 machine to the *bin* subdirectory under the working directory on your Commerce Suite 5.1 machine (*WCS51_userdir/temp/bin*), copy the following files:
 - *wcim.sh*
 - *wcimenv.sh*
- From the *WCS51_installdir/xml* directory in the Commerce Suite 5.1 machine, copy the *product.xml* and *product.dtd* files to the *xml* subdirectory under the migration working directory (*WCS51_userdir/temp/xml*).
- For Payments instance backup:

From the *WC55_installdir/payments/lib* directory on your WebSphere Commerce 5.5 machine, copy the *eTillConfig.jar* file (for Payments instance backup) to the *lib* subdirectory under your work directory on your remote Commerce Suite 5.1 machine (*WCS51_userdir/temp/lib*).

3. On your Commerce Suite 5.1 machine, before running the *wcim.sh* script, update the environment variables in the *wcimenv.sh* file (that you copied to the working directory on your remote machine) for backup.

The *wcimenv.sh* file contains the following entries:

```
...

WCIM_BACKUP="true"
WCIM_INPLACE="false"
WCIM_MIGRATE_FROM="51"
DB_TYPE="db2"

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...

WAS_PATH="WAS35_installdir"
WC_PATH="WCS51_installdir"
WEBSERVER_PATH="HTTPServer1312_installdir"
ANT_PATH="WCS51_installdir/temp/lib"
WORK_DIR="WCS51_installdir/temp"
MIG_FROM_WAS_INSTANCE="default"
MIG_TO_WAS_INSTANCE="default"
export MIG_FROM_WAS_INSTANCE
export MIG_TO_WAS_INSTANCE
WPM_PATH="/QIBM/ProdData/CommercePayments/V55"
export WPM_PATH

...

LOG_FILE="wcimbackup.log"
INSTANCE="instance_name"
PATH=$WAS_PATH/java/jre/bin:$PATH
```

```
export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH
...
```

Note that the log file set by the LOG_FILE variable gets generated in the WORK_DIR/logs/ directory.

4. On your Commerce Suite 5.1 machine, run the `wcim.sh` script as follows:
 - a. From an iSeries command line, enter QSH to open a QShell window.
 - b. Change to the directory where `wcim.sh` is stored.
 - c. Type:

```
wcim.sh [wcimenv_path]
```

where:

- *wcimenv_path* is the path where the `wcimenv.sh` is located on your machine. If this file is located in the same directory as `wcim.sh`, this parameter is not required.
5. Copy the `wcbackup51.zip` file from the working directory on the remote Commerce Suite 5.1 machine (for example, `WCS51_userdir/temp/zip`), to the `zip` subdirectory of the working directory on the WebSphere Commerce 5.5 machine (for example, `WC55_userdir/temp/zip`).

If it does not exist on your WebSphere Commerce 5.5 machine, you will have to create the `WC55_userdir/temp/zip` directory.

Running WCIM for remote migration:

Notes:

1. Ensure that you have copied all necessary files from the WebSphere Commerce 5.5 machine to the remote Commerce Suite 5.1 machine as documented in step 2 on page 35.
2. The WCIM_INPLACE variable is, in fact, set to true when performing a remote migration.
3. The Web server member in QUSRSYS/QATMHINSTC for the Commerce Suite 5.1 instance being migration need to be renamed prior to running the WCIM for migration to avoid errors with the creation of the new Apache Web server instance.

After running WCIM for backup, run WCIM to migrate your instance and instance related files:

1. Before running the `wcim.sh` script on your WebSphere Commerce 5.5 machine, search all the files in the `wcbackup51.zip` package for all occurrences of the Commerce Suite 5.1 machine's node name, IP address, and domain. Replace these occurrences with the WebSphere Commerce 5.5 machine's node name, IP address, and domain name respectively. For example, the files that you will likely need to update include:
 - `product.xml`
 - `product.xml.migration`
 - `instance_name.xml`
 - `instance_name_was.xml`

- *instance_name_was_DB.xml*
- *instance_name_was_Start.xml*
- *instance_name_was._EJB.xml*
- *instance_name_was_VH.xml*
- *cache.properties*

You should check all files in the *wbackup51.zip* package except for the Web server configuration file, for example, *httpd.conf* if you are using IBM HTTP Server. The Web Server configuration file is not migrated by WCIM so you need not update it.

2. Update the environment variables in the *wcimenv.sh* file for your machine for migration.

The *wcimenv.sh* file typically contains the following entries:

```
...

WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="51"
DB_TYPE="db2"

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...

WAS_PATH="WAS50_installdir"
WC_PATH="WC55_installdir"
WEBSERVER_PATH="HTTPServer1326_installdir"
ANT_PATH="WAS50_installdir/lib"
WORK_DIR="WC55_userdir/temp"
MIG_FROM_WAS_INSTANCE="default"
MIG_TO_WAS_INSTANCE="default"
export MIG_FROM_WAS_INSTANCE
export MIG_TO_WAS_INSTANCE
WPM_PATH="/QIBM/ProdData/CommercePayments/V55"
export WPM_PATH

...

LOG_FILE="wcimmigration.log"
INSTANCE="instance_name"
PATH=$WAS_PATH/java/jre/bin:$PATH

export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH

...
```

Note that the log file set by the *LOG_FILE* variable gets generated in the *WORK_DIR/logs/* directory.

3. On your WebSphere Commerce 5.5 machine, run the *wcim.sh* script as follows:
 - a. From an iSeries command line, enter *QSH* to open a QShell window.
 - b. Change to the directory where *wcim.sh* is stored.

c. Type:

```
wcim.sh [wcimenv_path] [PM_password]
```

where:

- *wcimenv_path* is the path where the `wcimenv.sh` is located on your machine. If this file is located in the same directory as `wcim.sh`, this parameter is not required.
- *PM_password* is the WebSphere Commerce Payments instance password and is only used when migrating a WebSphere Commerce Payments instance. Note that usage of WCIM for WebSphere Commerce Payments migration is covered in detail in “Migrating your Payments instance using WCIM” on page 84.

Note: The *PM_password* parameter is required for migrating a Payments instance. You have to specify the `wcimenv` path even if it is in the current directory; in this case use `."` to specify the current directory. For example:

```
wcim.sh . mypassword
```

Running WCIM for an in-place instance migration

The following sections describe the steps for an in-place migration.

Pre-requirements for an in-place instance migration: If you are installing your WebSphere Commerce 5.5 software components on the same machine as Commerce Suite 5.1, you should have completed the steps described in “Manually migrating your previous WebSphere Application Server configuration” on page 26.

Running WCIM for in-place backup: Before running WCIM for migration, you should run WCIM to create a backup of your current instance and instance related files:

1. Before running the `wcim.sh` script for backup, update the environment variables in the `wcimenv.sh` file to reflect your environment.

The `wcimenv.sh` file typically contains the following entries:

...

```
WCIM_BACKUP="true"  
WCIM_INPLACE="true"  
WCIM_MIGRATE_FROM="51"  
DB_TYPE="db2"
```

```
export WCIM_BACKUP  
export WCIM_INPLACE  
export WCIM_MIGRATE_FROM  
export DB_TYPE
```

...

```
WAS_PATH="WAS50_installdir"  
WC_PATH="WCS51_installdir"  
WEBSERVER_PATH="HTTPServer1312_installdir"  
ANT_PATH="WAS50_installdir/lib"  
WORK_DIR="WC55_userdir/temp"  
MIG_FROM_WAS_INSTANCE="default"  
MIG_TO_WAS_INSTANCE="default"  
export MIG_FROM_WAS_INSTANCE  
export MIG_TO_WAS_INSTANCE  
WPM_PATH="Payments_installdir"  
export WPM_PATH
```

```

...
LOG_FILE="w cimbackup.log"
INSTANCE="instance_name"
PATH=WAS50_installdir/java/jre/bin:$PATH

export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH

```

...
 Note that the log file set by the LOG_FILE variable gets generated in the WORK_DIR/logs/ directory.

2. Run the w cim.sh script as follows:
 - a. From an iSeries command line, enter QSH to open a QShell window.
 - b. Change to the directory where w cim.sh is stored.
 - c. Type:

```
w cim.sh [w cimenv_path]
```

where:

- *w cimenv_path* is the path where the w cimenv.sh is located on your machine. If this file is located in the same directory as w cim.sh, this parameter is not required.

Running WCIM for in-place migration:

Note: The Web server member in QUSRSYS/QATMHINSTC for the Commerce Suite 5.1 instance being migration needs to be renamed prior to running WCIM for migration to avoid errors with the creation of the new Apache Web server instance.

After running WCIM for backup, run WCIM to migrate your instance and instance-related files:

1. Before running the w cim.sh script for migration, update the environment variables in the w cimenv.sh file.

The w cimenv.sh file contains the following entries:

```

...
WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="51"
DB_TYPE="db2"

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...
WAS_PATH="WAS50_installdir"
WC_PATH="WC55_installdir"
WEBSERVER_PATH="HTTPServer1326_installdir"
ANT_PATH="WAS50_installdir/lib"
WORK_DIR="WC55_userdir/temp"

```

```

MIG_FROM_WAS_INSTANCE="default"
MIG_TO_WAS_INSTANCE="default"
export MIG_FROM_WAS_INSTANCE
export MIG_TO_WAS_INSTANCE
WPM_PATH="Payments_installdir"
export WPM_PATH

...

LOG_FILE="wcimmigration.log"
INSTANCE="instance_name"
PATH=$WAS_PATH/java/jre/bin:$PATH

export WAS_PATH
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH

```

...
Note that the log file set by the LOG_FILE variable gets generated in the WORK_DIR/logs/ directory.

2. Run the wcim.sh script as follows:
 - a. From an iSeries command line, enter QSH to open a QShell window.
 - b. Change to the directory where wcim.sh is stored.
 - c. Type:

```
wcim.sh [wcimenv_path] [PM_password] [db_userid db_password]
```

where:

- *wcimenv_path* is the path where the wcimenv.sh is located on your machine. If this file is located in the same directory as wcim.sh, this parameter is not required.
- *PM_password* is the WebSphere Commerce Payments instance password and is only used when migrating a WebSphere Commerce Payments instance. Note that usage of WCIM for WebSphere Commerce Payments migration is covered in detail in “Migrating your Payments instance using WCIM” on page 84.
- *db_userid* and *db_password* are the valid user ID and password for the relational database. These parameters are only required if, after instance migration, the relational database will be on a remote system from the migrated Payments instance.

Note: The *PM_password* parameter is required for migrating a Payments instance. You have to specify the wcimenv path even if it is in the current directory; in this case use "." to specify the current directory. For example:

```
wcim.sh . mypassword
```

Verifying that WCIM ran successfully

WCIM generates a log file as specified in wcimenv.sh file by the SET LOG_FILE directive. For example, for SET LOG_FILE=wcimmigration.log, the log file wcimmigration.log is generated in the *WC55_userdir/temp/logs/* directory if WORK_DIR is set to *WC55_userdir/temp*. In this log file, search for the following statement:

Info: Instance migration is completed successfully.

WCIM also generates an `instanceXmlMigration.log` migration log file in the `WORK_DIR/logs` directory (for example, `WC55_userdir/temp/logs` if `WORK_DIR=WC55_userdir/temp`). In `instanceXmlMigration.log`, search for
Info: WebSphere Commerce instance configuration migrated successfully.
Info: WebServer configuration migrated successfully.

Note: Receiving this message does not guarantee that your WebSphere Commerce instance has been migrated successfully, since you have not migrated the database at this point. You migrate your database to the WebSphere Commerce 5.5 schema in Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59 and then you can verify that your instance is migrated in “Verifying successful instance and database migration” on page 79.

Recovering from WCIM failure: If WCIM fails for some reason, inspect the logs for the potential problem and make any fixes that are required. You can also consult Appendix H, “Troubleshooting,” on page 161 for any potential problems and fixes that are known. Otherwise, you should restore your system from previous backups and carefully review the steps in this section and redo the instance migration.

Note that for problems associated with instance XML migration, you can simply fix any problems and rerun the WCIM migration. For Web server migration, since WCIM only migrates a local IBM HTTP Server, you need to delete `httpd.conf`, before rerunning the WCIM migration (if the `httpd.conf` file already exists).



If you receive the following message during instance migration using WCIM, you will need to isolate the ANT statement that is failing and then take appropriate action:

```
Event: MethodId: invokeAnt - executing ant task:postmigrationcopy  
Error: Ant task has failed. Please check the log file.
```

1. Check the `wcimbackup.log` (if you are running WCIM for back up) or `wcimmigration.log` (if you are running WCIM for migration) for an error string or message. These log files get generated in the `logs` subdirectory under the working directory specified by `WORK_DIR` for the string “Error:” when you run WCIM. An example of an error message is as follows:

```
Event: MethodId: invokeAnt - executing ant task:postmigrationcopy  
Error: Ant task has failed. Please check the log file.
```

2. Refer to the **Debug:** statement immediately before the **Event:** or **Error:** statement. For example:

```
Debug: java -classpath classpath -buildfile ... migratejsp
```

3. Run the `java` command referenced by the **Debug:** statement at a command prompt while in your Java Runtime environment location on your machine, `QIBM/ProdData/Java400/jdk13/bin`. The error message will be displayed.
4. Take appropriate action based on the error message.
5. When you have resolved the error message, rerun WCIM.

Verifying WebSphere Commerce Payments cashier profiles

After running the instance migration, verify the following items for WebSphere Commerce Payments in the `Payments` directory:

```
WC55_userdir/instances/instance_name/xml/payment
```

1. Verify that the profiles for the Cassette for SET and the Cassette for CyberCash do not exist.

2. Verify the existing Commerce Suite 5.1 cashier profiles have been updated to remove all references to “Payment Manager”. Open the cashier profiles, for example, files such as WC51_CustomOffline_BillMe.profile and check that they have been updated. A good way to verify the update is to see if the references to “Payment Manager” have been changed to “Commerce Payments” as in the following example.

Before migration:

```
<!-- =====
Parameters required by Payment Manager for order creation
===== -->
```

After migration:

```
<!-- =====
Parameters required by WebSphere Commerce Payments for order creation
===== -->
```

3. Verify the new profile WC_Paymentech has been created.
4. Verify that the WebSphere Commerce 5.5 Payments profile.dtd file exists.
5. Verify that the following profile files exist:
 - WC_Paymentech (new)
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_OfflineCard.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_OfflineCard.profile

For details on migrating to WebSphere Commerce Payments, see Chapter 8, “Migrating to Commerce Payments,” on page 81.

After running the WCIM script

The following sections describe remaining tasks to complete before running the database migration script.



If you are using a non-default WebSphere Application Server instance, ensure that the instance is started so that you can access the WebSphere Application Server Administration Console for that instance.

Removing obsolete class paths in WebSphere Application Server

Before you deploy your migrated ear file in WebSphere Application Server 5.0, perform the following steps to remove all obsolete Commerce Suite 5.1 class paths on the JVM setting of the migrated application server (WebSphere_Commerce_Suite_-_instance_name).

1. Log on to the WebSphere Application Server Administration Console and start the WebSphere Application Server - server1 application.
2. Expand **Servers** and click **Application Servers** in the left frame.
3. From the Application Servers panel, choose *server_name* —> **Process Definition** —> **Java Virtual Machine** where *server_name* is your migrated application server (for example, WebSphere_Commerce_Suite_-_instance_name).

4. Remove all content from the Classpath field that reference Commerce Suite 5.1 paths, except the path for the `wcsjni.jar` file.
5. Save you updates (click **Apply** and **OK**).

The idea here is to remove references to any Commerce Suite 5.1 JAR files as they are not required in a WebSphere Commerce 5.5 environment. However, if you have added new `classpath` entries in your Commerce Suite 5.1 system, it is assumed that they are still required in a WebSphere Commerce 5.5 environment, for example, for customized commands.

Adjusting JVM properties in WebSphere Application Server

After you have run the WCIM script, perform the following steps to adjust JVM properties from the migrated application server (`WebSphere_Commerce_Suite_-_instance_name`) in order to avoid conflicts.

1. Log on to the WebSphere Application Server Administration Console.
2. Expand **Servers** and click **Application Servers** in the left frame.
3. From the Application Servers panel, choose `server_name` → **Process Definition** → **Java Virtual Machine** → **Custom Properties** (under Additional Properties), where `server_name` is your migrated application server (for example, `WebSphere_Commerce_Suite_-_instance_name`).
4. Remove the following obsolete properties (if they are present on your system):
 - `javax.rmi.CORBA.UtilClass`
 - `com.ibm.ivj.ejb.runtime.instancename`
5. Add the following new properties (if they are not present on your system). Click **New** to add them:

Name	Value
<code>com.ibm.servlet.file.esi.timeOut</code>	<code>0</code>
<code>os400.define.class.cache.file</code>	<code>/QIBM/ProdData/CommerceServer55/lib/classcache.jar</code>
<code>os400.define.class.cache.hours</code>	<code>9999</code>
<code>os400.define.class.cache.maxpgms</code>	<code>40000</code>

6. Save your updates (click **Apply** and **OK**).

Running the `wcimWasConfig.jacl` script

WCIM generates a file `wcimWasConfig.jacl` in the directory specified by `WORK_DIR` in `wcimenv.sh` file. After you complete the instance migration using WCIM, you need to run this JACL script file as it contains WebSphere Application Server 5.0 administrative commands that will minimize the manual WebSphere Application Server configuration during the migration.

Among other things, the script will add virtual hosts into WebSphere Application Server as required. The script by default adds the following virtual hosts for your migrated WebSphere Commerce instance:

- `VH_instance_name` on ports 80 and 443 (for your WebSphere Commerce instance)
- `VH_instance_name_Tools` on port 8000 (for tools such as WebSphere Commerce Accelerator)

- `VH_instance_name_Admin` on port 8002 (for the WebSphere Commerce Administration Console)
- `VH_instance_name_OrgAdmin` on port 8004 (for the WebSphere Commerce Organization Administration Console)

Before running the script:

- **For a remote migration scenario only**, edit the `wcimWasConfig.jacl` file generated in the `WCIM_work_dir` on the WebSphere Commerce 5.5 machine. Search for all occurrences of the Commerce Suite 5.1 machine's node name, IP address, and domain, and replace them with the WebSphere Commerce 5.5 machine's node name, IP address, and domain respectively.
 - In the `wcimWasConfig.jacl` script, ensure that the value of the `DB_IMPLE_CLASSNAME` variable is not null. If it is null, you need to assign the value according to the database management system you are using. For example, `COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource` for DB2.
 - Depending on whether you wish to deploy the migrated `.ear` file in a WebSphere Application Server single server configuration or WebSphere Application Server network deployment configuration, you need to verify or update the values in the script file before you run it.
- Any lines requiring your attention or modification are indicated in the script.
- Modify the server name in the `wcimWasConfig.jacl` file to apply the updates to the appropriate server that you are migrating (for example, `WebSphere_Commerce_Server_-_instance_name`) rather than `server1` (the default server). Refer to the following line in the script:

```
# Please verify the server name
set SERVER_NAME server1
```



You can check the migrated WebSphere Commerce server name from the WebSphere Application Server 5.0 Administration Console under **Servers** —> **Application Servers**.

- If required, start the `WebSphere_Commerce_Server_-_instance_name` application server. You can start it from the WebSphere Application Server Administration Console if you are using WebSphere Application Server ND, otherwise use the WebSphere Application Server `startServer` command.

Important

There is a possibility that the port numbers for these virtual hosts may conflict with existing port numbers. For example, port 443 of the default host if you use the default setting when you installed the previous WebSphere Commerce system. This port will conflict with WebSphere Commerce 5.5 and prevent you from successfully launching your migrated store.

Verify these port numbers or virtual host names do not already exist and in use on your system. If they are being used by your system, you should either change the name of the new virtual host or its predefined port number in the script prior to running it to an unused value. You can also remove the new virtual host if it is already defined. To see your existing virtual hosts, from the WebSphere Application Server 5.0 Administration Console, expand **Environment** and double-click **Virtual Hosts** to see them displayed in the WebSphere Application Server panel.

For example, suppose you have the following host alias definitions for default_host that are generated on the WebSphere Commerce 5.5 system from the Commerce Suite 5.1 backup, by the WCIM script:

```
<X.X.X.X> :80
<X.X.X.X> :443
localhost :80
localhost :443
host_name :80
host_name :443
fully_qualified_host_name :80
fully_qualified_host_name : 443
```

With the above entries, after completing the migration, you may have problems getting your home page to display. Simply removing these entries could enable the appearance made the store home page to appear. In effect, this default_host *grabs* the commerce requests and so in this case they never get to the right recipient — VH_instance_name. Since the WebSphere Commerce tooling is runs on other ports (for example, 8000, 8002, and 80004 by default) so the tools such as the Administration Console is unaffected.

To invoke the JACL script, enter:

```
WAS50_installdir/bin/wsadmin -f WCIM_work_dir/wcimWasConfig.jacl
```

If you want to deploy to a non-default WebSphere Application Server instance, invoke the JACL script as follows:

```
WAS50_installdir/bin/wsadmin -instance WAS_instance
-f WCIM_work_dir/wcimWasConfig.jacl
```

Checking the JDBC Provider

Before deploying the migrated .ear file to your application server, check the JDBC provider in the WebSphere Application Server Administration Console to verify if the data source is configured correctly for the current WebSphere Commerce 5.5 environment. If the data source is null, you need to add it manually prior to deploying the migrated .ear file.

1. Logon again to the WebSphere Application Server Administration Console. From the Administration Console, expand **Resources** and click **JDBC Providers**
> **Browse Server** and select your WebSphere Commerce instance. Click **OK**.

2. Back on the JDBC Providers panel, (from the Administration Console, expand **Resources** and click **JDBC Providers**) if the WebSphere Commerce 5.5 data source does not appear, proceed with the following steps:
 - a. From the JDBC Providers panel, click **New** and select the appropriate JDBC Provider from the drop down list (for example, DB2 JDBC provider for DB2[®] databases).
 - b. Configure JDBC Provider name, JDBC classpath fields. Click **Apply** and **OK** to save your changes
 - c. Under the Additional Properties section of the Configuration panel, click **Data Sources (Version 4) —> New**
 - d. Configure the Name, JNDI Name (for example, `jdbbc/WC_instance_name` WebSphere Commerce Suite DB2 Datasource for DB2 databases, the Database Name, and the Default user ID and Default Password fields for the database.
 - e. Click **Apply** and **OK**.
 - f. Save your changes.

Deploying the migrated .ear file

After you have migrated your instance using WCIM and run the `wcimWasConfig.jacl` script, you need to deploy the migrated .ear file as follows:

1. Logon to the WebSphere Application Server Administration Console.
2. Click on **Applications —> Install new application**. Since the .ear file on your iSeries machine, select **Server**, and enter the path to the migrated .ear file (for example, `WCIM_work_dir/migrate/WebSphere_Commerce_Suite_-_instance_name.ear`) and click **Next**.
3. Select **Default bindings for EJB 1.1 CMPS** (not the **Generate Default Bindings** check box at the top) and enter the JNDI name, database user name, database password, and verify the password in the section "*EJB1.1 CMP bindings*" of this panel, then click **Next**. (For example, **JNDI name:** `jdbbc/WC_instance_name` WebSphere Commerce Suite DB2 Datasource, **username:** `db2admin`, **password:** `db2admin`)

Note: The above values are an example. You need to get the correct data source value for your system which you can find in the `instance_name.xml` file for your WebSphere Commerce instance. You need to prepend "`jdbbc/`" to the data source value from the `instance_name.xml` and enter it above.

4. Accept the default values until you reach the Step 3 panel by clicking the **Next** button.
5. In the Step 3 panel:
 - a. Expand **Apply Multiple Mappings**
 - b. Select **EJB Module**
 - c. Provide your user name and password and click **Apply**
 - d. In **Specify existing Resource JNDI name** select the JNDI name for the Application Server Name.
 - e. Click **Apply** for the JNDI name and click **Next**.
6. In the Step 4 panel:
 - a. Select all the listed EJB files
 - b. Choose the JNDI name (*node_name:* `jdbbc/WC_instance_name` WebSphere Commerce Suite DB2 Datasource from the **Specify existing resource jndi name**) drop down list. Click **Apply**, and then **Next**.

7. Accept the default values until you reach the Step 6 panel by clicking the **Next** button.
8. In the Step 6 panel, map the virtual host for Web modules similar to:
 - commerceAccelerator
 VH_instance_name_tools
 - OrganizationAdministration
 VH_instance_name_Orgadmin
 - SiteAdministration
 VH_instance_name_Admin
 - Stores VH_instance_name

Press **Next**.

9. In the Step 7 panel:
 - a. Select all modules
 - b. Select the WebSphere Commerce server (long format name) in **Clusters and Servers** and click **Apply**
 - c. Accept the default configuration and click **Next** for the remaining windows.
10. Click **Finish**, and wait until you receive the update successful message.
11. Click on *node_name* → **Environment** → **Update Web server plugin** → **OK**. Ensure the update successful message appears.
12. Save your changes.
13. Change the Classloader mode and WAR Classloader policy:
 - a. Expand **Applications**
 - b. Click **Enterprise Applications**
 - c. Click on application prefaced by WC_
 - d. Change ClassLoader Mode to **PARENT_LAST**
 - e. Change WAR Classloader Policy to Application
 - f. Click **OK**
 - g. Save your changes
14. Restart WebSphere Application Server.

Note: If you have deleted the virtual hosts defined in the JACL script so they do not conflict with the existing ones as described in “Running the wcmWasConfig.jacl script” on page 44, during EAR deployment, you have to map the existing ones to the WAR module.

For more information on deploying your applications using the WebSphere Application Server 5.0 see the WebSphere Application Server 5.0 for iSeries site <http://www.ibm.com/servers/eserver/iseries/software/websphere/wsappserver/>

Selecting the WebSphere Application Server pass by reference setting

After deploying the migrated .ear file, select the WebSphere Application Server 5.0 pass by reference service setting. When this setting is enabled, the WebSphere Application Server Object Resource Broker passes parameters by reference instead of by value which avoids making an object copy. If you do not enable pass by reference, the parameters are copied to the stack before every remote method call is made, which can be expensive.

To enable the setting:

1. Open the WebSphere Application Server Administration Console and logon.
2. On the left hand panel, click **Application Servers** -> *WC_instance_name* -> **ORB Service**
3. Select **Pass by reference** for the migrated instance and
4. Click **OK** and save the changes.

Creating JNDI names for JCA connectors

The JNDI names of all JCA connectors are created during the installation of the EAR. When an existing EAR or instance is migrated, these names are not created. For example, the JNDI name for the J2C Connection Factory of JMS Connector is "eis/JCAJMS". It is the name found under the resource adapters under the EAR configuration.

To add in the JNDI names for the JCA connectors, do the following:

1. Open the WebSphere Application Server Administration Console and logon.
2. On the left hand panel, click **Applications** -> **Enterprise Applications** -> *WC_instance_name*.
3. At the bottom of the page, click **Connector Modules** -> *Connector Module Name* -> **Resource Adapter**
4. At the bottom of the page, click **J2C Connection Factories** -> **New**
5. Use the following table to enter:
 - The name in the **Name** field.
 - The JNDI name in the **JNDI Name** field.
 - The description in the **Name** field.

Connector Module Name	Name	JNDI name	Description
Enablement-JCAEmailConnector.rar	Enablement-JCAEmailConnector.rar	eis/JCAEmail	<i>WC_instance_name</i> .JCA EMail Connector
Enablement-JCACrossWorldsConnector.rar	Enablement-JCACrossWorldsConnector.rar	eis/CWConnectionFactory	<i>WC_instance_name</i> .IBM CrossWorlds Resource Adapter for InterChange Server
Enablement-JCAFileConnector.rar	Enablement-JCAFileConnector.rar	eis/JCAFile	<i>WC_instance_name</i> .JCA File Connector
Enablement-JCAJMSConnector.rar	Enablement-JCAJMSConnector.rar	eis/JCAJMS	<i>WC_instance_name</i> .JCA JMS Connector
Enablement-JCASampleConnector.rar	Enablement-JCASampleConnector.rar	eis/JCASample	<i>WC_instance_name</i> .Sample Connector

6. Select **None** for the Authentication Preference.
7. 8. Click **OK** at the bottom of the page.
8. 9. Click the newly created **JCA Connection Factories** again and modify the connection pool properties by clicking **Connection Pool** at the bottom of the page.
9. 10. Update **Max Connection** to be 30 and click **OK** twice to return back to **J2C Connection Factories** page.
10. 11. Save the configuration by clicking **Save** on the top menu and click **OK** to confirm saving the configuration on the next page.
11. 10. Repeat steps 3 through 10 above for each connector module name in the table above.
12. 12. Restart WebSphere Commerce.

Updating file and directory permissions

After you have run the WCIM and `wcimWasConfig.jacl` scripts, you need to run the following script to change the permission on the files and directories for the migrated `.ear` file:

1. Log in to QSH.
2. Run the following script from the `WC55_installdir/bin` directory:

```
chgaut_iseries_ear.sh cell_name wc_instance was_instance
```

For example, from the `/QIBM/ProdData/CommerceServer55/bin` directory run:
`chgaut_iseries_ear.sh TORASCSD demo default`

Transitioning and deploying your custom code

As part of your transition to WebSphere Commerce 5.5, you need to transition your custom code and business logic to the WebSphere Commerce 5.5 level. For information on how to upgrading your code to the WebSphere Commerce 5.5 level, see the section "Transitioning your customized or extended code" in the *WebSphere Commerce Studio Migration Guide*.

For detailed information about deployment of customized code assets, see the sections on the deployment of business logic or a controller command, or a task command as required in the *WebSphere Commerce Programming Guide and Tutorials* document.

Updating deployed instance XML files

For your migrated WebSphere Commerce instances, update the instance XML files to the WebSphere Commerce 5.5.0.2 level by running the following steps. If you have more than one WebSphere Commerce instance, use only one of your instance names for the variable `instance_name`. The script will update all existing instances.

1. Navigate to the `WC55_installdir/bin` directory.
2. Run the following command:

```
./config_ant.sh -buildfile WC55_installdir/xml/config/updateInstances.xml  
-DupdateCEP=no -Duninstall=no -DinstName=instance_name
```

Customizations in `httpd.conf`

During the instance migration, WCIM reads the Web server configuration information from the previous `instance.xml` file and reconfigures the WebSphere Commerce 5.5 `httpd.conf` file from WebSphere Application Server 5.0 installation with that information. Any customized configurations in the previous version of the native iSeries HTTP configuration file are not migrated since these configurations are out of the scope of the WCIM. If you have customized your configurations, you will have to migrate the customized configurations manually after instance and Web server migration.

Reconfiguring Web servers

In order to reconfigure your Web servers, follow the Web server installation section in the *WebSphere Commerce Installation Guide*. Complete the necessary preconfiguration steps in that section. Some additional user configuration is required in WebSphere Commerce 5.5 to configure any remote Web Server including IBM HTTP Server.

To perform this additional configuration, log on to Configuration Manager after migration and complete the necessary fields on the Web server panel to configure the Web server. Click on **Apply** on the panel to configure the Web server. For

information on configuring panels in Configuration Manager, see the chapter on configuring WebSphere Commerce instances in the *WebSphere Commerce Installation Guide*.

Enabling WebSphere Application Server EJB Security

If you had WebSphere Application Server EJB Security enabled in Commerce Suite 5.1, you need to re-enable it for WebSphere Commerce 5.5. See the section "Enabling WebSphere Application Server security" in the *WebSphere Commerce Security Guide* for the steps to redeploy the EJB security.

Note any additional machine recommendations for EJB Security in the *WebSphere Commerce Security Guide*.

Copying files with static content to the documentation root directory

To migrate Web assets that are not located under the WebSphere Commerce WAR directory, you need to copy files with static content (such as image and HTML files) to the document root directory.

Copy all files with static content in the `WC51_userdir/instances/instance_name/web` directory to the `WC55_userdir/web` directory.

Chapter 5. Before migrating your database

This chapter describes the pre-database migration actions and consideration you need to take before you actually migrate your database as described in Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59.



- Ensure that you have performed the steps in Chapter 1, “Before you migrate from Commerce Suite 5.1,” on page 3 including the steps in “Ensuring no new messages are stored in MSGSTORE table” on page 4 as well as the other sections in “Premigration actions” on page 3, while you had an operational Commerce Suite 5.1 system.
- For details on the database schema differences between Commerce Suite 5.1 and WebSphere Commerce 5.5 refer to the WebSphere Commerce Production and Development online help. Once you have launched the online help, select **WebSphere Commerce Development information > Reference > Data > Database schema**. From here, select **Database tables** to view an alphabetical list of all database schema information. Select **Database changes in this release** to view the schema changes for Version 5.5. The online help also provides data model information. Ensure that you visit the WebSphere Commerce Technical Library (<http://www.ibm.com/software/commerce/library/>), for the latest versions of the WebSphere Commerce documentation, including any updates to the database schema information in the online help.

Note: Before running any database migration or premigration procedure, you must sign on to your iSeries machine with your instance profile that has *SECOFR authority. Use the following command:

```
CHGUSRPRF USRPRF() USRCLS(*SECOFR) SPCAUT(*USRCLS)
```

Then sign back on with this profile. After migration is complete, you should return the user profile to its original state:

```
CHGUSRPRF USRPRF() USRCLS(*USER) SPCAUT(*NONE)
```

Ensuring database journal receivers are large enough

WebSphere Commerce 5.5 uses CLOB’s (character large objects) as part of the schema. When migrating from previous versions of OS/400, the default journal receiver attached to the journal created for your database schema may not be of sufficient size for such entries. Your journal receiver must have RCVSIZOPT(*MAXOPT2) specified. Use the WRKJRNA command to determine your current receiver size options. If *MAXOPT2 is not specified, you will need to modify your journal with a command similar to the following before attempting to migrate your database:

```
CHGJRN JRN(instance/QSQJRN) JRNRCV(*GEN) RCVSIZOPT(*MAXOPT2)
```

where *instance* is the name of your database schema, and QSQJRN is the name of the journal attached to your database tables.

Updating database system views to the V5R2 level

If you are migrating a schema that was created at the V4R5 level, your system views may not contain the SYSTRIGGERS view. This view is needed to successfully complete the migration to WebSphere Commerce 5.5. In this case, you will need to clear your system tables and then repopulate them.

To determine if your schema was created prior to V5R1, run the following query:

```
Select * from SCHEMA_NAME/systriggers
```

where *SCHEMA_NAME* is the name of the Commerce Suite 5.1 instance you are migrating to WebSphere Commerce 5.5. (Note that you must query the schema's copy of the view and not the system wide copy of the view.)

- If the query returns a result, you may skip the remainder of this step to repopulate your system view.
- If the query returns the following error message:
SYSTRIGGERS in *SCHEMA_NAME* type *FILE not found.

you must repopulate your system views as described below.

To repopulate your system views, be sure that all database operations against this schema are complete, and then sign on as the user profile for your Commerce Suite 5.1 instance, and run the following two commands:

```
CALL PGM(QSYS2/QSQXRLF) PARM(DLT SCHEMA_NAME)  
CALL PGM(QSYS2/QSQXRLF) PARM(CRT SCHEMA_NAME)
```

where *SCHEMA_NAME* is the name of the Commerce Suite 5.1 instance you are migrating to WebSphere Commerce 5.5.

Sign off the user profile for your instance before continuing.

Considering the order of columns

The order of the columns in the migrated WebSphere Commerce 5.5 database tables are not guaranteed to be the same as the tables created when you create a new WebSphere Commerce 5.5 instance. It is recommended that when you query a table using `select *`, that you retrieve the value of a column by explicitly specifying the column name, and not by column position. For example, in a JDBC program:

```
resultSet rs= statement.executeQuery("select * from address");  
    (while rs.next())  
    {  
        p= rs.getObject(1); <---- not recommended  
  
        p=rs.getLong("address_id"); <---- recommended  
    }
```

Conversely, when you insert data, you should explicitly specify the column name. For example, consider the following insert statements:

```
insert into address values( 10001,10002,'myaddress'); <-- not recommended  
  
insert into address (address_id,addrbook_id, displayname)  
    values (10001,10002,'myaddress'); <-- recommended
```

Checking the unique indexes in the MSGTYPES table

In WebSphere Commerce 5.5, there are new rows that are created in the MSGTYPES table *out-of-the-box* (that is, for a non-migrated system). The MSGTYPE_ID and NAME columns are primary key and unique index respectively. If you have created your own message types in Commerce Suite 5.1, ensure that they are not in conflict with the new ones. If they are in conflict, you can modify the MSGTYPE_ID values or modify the NAME or change both. The key is that both MSGTYPE_ID and NAME must be unique in the table. You will need to recompile the commands that refer to the custom message types. Otherwise, you may encounter problems when migrating your data from Commerce Suite 5.1.

The following table lists the newly added message types for WebSphere Commerce 5.5, compared with Commerce Suite 5.1 products. In that case, you can check if you have anything in the MSGTYPES table that shares the same MSGTYPE_ID or NAME as listed.

Table 6. MSGTYPE_ID and NAME additions to MSGTYPES table in WebSphere Commerce 5.5

MSGTYPE_ID	NAME
130	OrderCancelForMerchant
150	OrderSummaryReportNotification
151	StoreUsageReportNotification
152	StoreCommerceReportNotification
153	SiteCommerceReportNotification
221	PriceAndAvailabilityCheck
222	BatchAvailability
223	ShoppingCartTransfer
224	CheckInventoryAvailabilityBE
250	CustomerMessage
251	CouponsSavedNotification
310	ResellerRegistrationApprovedNotification
320	ResellerRegistrationRejectedNotification
400	CollabEmail
501	RFQSubmitMessage
502	RFQCloseMessage
503	RFQCompleteMessage
600	ResponseNotification

Note that not all the above entries will show up after the data migration. CollabEmail, RFQSubmitMessage, RFQCloseMessage, RFQCompleteMessage, and ResponseNotification are only available under Business Edition.

Running the database preparation script

The database preparation script modifies the access control tables in your Commerce Suite 5.1 database so that they can be properly migrated. Running this script is required for proper database migration.

The database preparation script analyzes your existing Commerce Suite 5.1 databases and reports any data with the following characteristics:

- All items that do not have a parent product
- All members that do not have a parent in the organization entity.

The database preparation script also attempts to remove database triggers and reapplies all Commerce Suite 5.1 constraints. The script flags any triggers or constraints that it cannot apply properly. You need to manually apply any such flagged items before running the database migration script.

The report lists the items described in “Mandatory database premigration items” on page 136 and “Optional database premigration items” on page 138.

For the mandatory items, you need to take the requested actions before proceeding with the database migration. After taking the requested actions, you need to rerun the database preparation script in order to verify that all the mandatory actions have indeed been addressed.

For the optional items, you are not required to complete the recommended actions before proceeding with the database migration. It is recommended, however, that you correct any of the data that gets flagged or if you do not want the default assignments of the migration script. After correcting your data, you should rerun the database preparation script to ensure all items are fixed.

Once you have fixed all necessary items, back up your updated Commerce Suite 5.1 databases once again as described in Chapter 2, “Backing up Commerce Suite 5.1,” on page 17, in order that you have an up-to-date copy of your databases.

Run the WebSphere Commerce 5.5 database preparation script as described below.

DB2 databases

1. Open a QShell window by running the STRQSH command.
2. Switch to the bin subdirectory where WebSphere Commerce 5.5 is installed. For example, *WC55_installdir/bin*.
3. Run the database preparation script as follows:

```
premigratedb51.sh db2 db_name db_userID password
```

where:

- *db2* represents the database management system you are using and should be *db2* for DB2 databases.
- *db_name* is the Commerce Suite 5.1 database that you are migrating, as displayed in the relational database directory.
- *db_userID* is the name of the instance user profile of the Commerce Suite 5.1 database that you are migrating. This should be the same as your database schema name.
- *password* is the password for the instance user profile used to connect to the Commerce Suite 5.1 database that you are migrating.

For example, to run the database preparation script for Commerce Suite 5.1 databases, you can use the following command:

```
premigratedb51.sh db2 dbname instance instance_password
```

- ___ 4. Check the `premigratedb51.log` file generated in the `instance_name` subdirectory under `WC55_userdir/instance/instance_name/migration/logs` for additional messages or errors. You need to resolve any errors that are flagged before proceeding.
- ___ 5. Check the `premigratedb51.log` file generated in the directory for additional messages or errors. You need to resolve any errors that are flagged before proceeding.

The return codes output by the database preparation script are listed in “Database preparation script return codes” on page 135.

If you have fixed the errors or warnings that are flagged by the database preparation script or made any changes to your database, it is recommended that you take another back up of your database as described in “Backing up your databases” on page 19.

In addition, if the data preparation script flags certain columns that contains data that exceeds the WebSphere Commerce 5.5 limits, see “Ensuring column lengths are not exceeded” on page 140.

Dropping custom constraints

If you have customized tables that contain foreign key links to the shipped Commerce Suite 5.1 tables, the data migration script may fail when it attempts to drop these referential integrity constraints (such as foreign keys, primary keys, and indexes) during data migration.

For DB2 databases, you need to drop these constraints using the SQL statements in the sections below. You will later restore the constraints after migrating your data to the WebSphere Commerce 5.5 schema as described in “Restoring custom constraints” on page 61.

1. Identify all new tables that you have added, as well as all the Commerce Suite 5.1 tables.
2. Identify all constraints and triggers (views, summary tables, triggers, SQL functions, SQL methods, referential constraints) that point from the new tables to the Commerce Suite 5.1 tables, or from Commerce Suite 5.1 tables to the new tables, or all new constraints you have added to the Commerce Suite 5.1 tables.

To determine the constraints that you have added to the Commerce Suite 5.1 tables that point to your custom tables:

Run the following query for DB2 databases:

```
db2 select CONSTNAME from SYSCAT.TABCONST
      where TABNAME='your_table_name'
```

3. For DB2 databases, drop all identified constraints. If there are any SQL errors due to constraints in the `migratedb_database_name.log` after running the database migration script, you will need to restore the database and drop or disable these constraints before attempting to migrate the database again.

For information on running database scripts, see “Running database scripts” on page xvi. Run the following SQL statement for each constraint you wish to drop:

```
ALTER TABLE instance_name.table_name DROP constraint constraint_name
```

where

instance_name

Is the name of your database schema or WebSphere Commerce instance.

table_name

Is the name of the customized table containing the constraint.

constraint_name

Is the name of the referential integrity constraint you wish to drop.

Note the constraints that you drop, as you will need to add them back after the data migration.

Chapter 6. Migrating your Commerce Suite 5.1 database

This chapter describes the steps to migrate your Commerce Suite 5.1 database schema to the WebSphere Commerce 5.5 schema level. Prior to performing the steps to migrate your schema, it is recommended that you have completed the steps in Chapter 1, “Before you migrate from Commerce Suite 5.1,” on page 3, Chapter 2, “Backing up Commerce Suite 5.1,” on page 17, Chapter 3, “Upgrading your software,” on page 21, and Chapter 5, “Before migrating your database,” on page 53. In addition, it is mandatory that you run the database preparation script as documented in “Running the database preparation script” on page 55 prior to the actual database migration described in this chapter. Note that the database preparation script modifies your database in order for proper migration to WebSphere Commerce 5.5.

After you have completed running the scripts to migrate your data, you perform some additional post database migration steps to verify the success of your database migration.

Important

Before proceeding with the migration steps described in this chapter and the remainder of this Guide:

- To run the following database migration utilities, the DB2 database must appear in your relational database directory. Use the `WRKRDBDIRE` command to determine the databases that are currently in your relational database directory.
- If you have extended the Commerce Suite 5.1 schema (for example, by adding a column to a table) you should perform the steps documented in Appendix D, “Extending the data migration script,” on page 127.
- Ensure that you have enough free disk space on your machine to accommodate a backup of the Commerce Suite 5.1 database plus space to accommodate logs and temporary disk space to run the migration script. This additional disk space should be at least two times the size of your current Commerce Suite 5.1 database.
- You can consider backing up your database before you attempt to migrate it. Note that the database migration script will perform a backup of your database in any case. If you wish to backup your current database, see “Backing up your databases” on page 19.

Migrating your database

After you have eliminated all items flagged by the database preparation script, you can run the database migration script to update your Commerce Suite 5.1 schema to the WebSphere Commerce 5.5 level. For an overview of what the migration script does when migrating Commerce Suite 5.1 databases, see Appendix E, “Overview of the WCIM tool and the data migration script,” on page 129. For a summary of the database schema changes between Commerce Suite 5.1 and WebSphere Commerce 5.5, see the topic “Database schema changes in this release” in the WebSphere Commerce Production and Development online help.

Note that you can only run the migration script once on the database.

Before migrating the database, ensure that the user who runs the database migration script has the authority to access the files in the *WC55_installdir* directory.

Note: Before running any database migration procedure, you must sign on to your iSeries machine with your instance profile that has *SEC0FR authority. Use the following command:

```
CHGUSRPRF USRPRF() USRCLS(*SEC0FR) SPCAUT(*USRCLS)
```

Then sign back on with this profile. After migration is complete, you should return the user profile to its original state:

```
CHGUSRPRF USRPRF() USRCLS(*USER) SPCAUT(*NONE)
```

DB2 databases

The following steps show you how to run the migration script for DB2 databases:

- ___ 1. From an iSeries command line, open a QShell window by running the STRQSH command.
- ___ 2. Switch to the bin subdirectory where WebSphere Commerce 5.5 is installed. For example, *WC55_installdir/bin*.
- ___ 3. Run the database migration script as follows:

```
migratedb51.sh db2 db_name db_userID password atp_option host_name backuplib
```

where:

- *db2* specifies that you are migrating a DB2 database.
- *db_name* is the Commerce Suite 5.1 database that you are migrating, as displayed in the relational database directory.
- *db_userID* is the instance user profile of the Commerce Suite 5.1 database that you are migrating. This should be the same as your schema name.
- *password* is the password for the instance user profile of the Commerce Suite 5.1 database that you are migrating.
- *atp_option* can either be *atp* or *noatp* as follows:
 - *atp* specifies that you wish to migrate your existing Commerce Suite 5.1 inventory data to the available-to-promise (ATP) inventory representation required for WebSphere Commerce 5.5.
 - *noatp* specifies that you do not wish to migrate your existing Commerce Suite 5.1 inventory data to the available-to-promise (ATP) inventory representation.

See “Considerations for ATP migration” on page 140 for more details and considerations on this important option.

- *host_name* is the fully qualified host name of your machine, where the database you are migrating resides.
- *backuplib* is an existing iSeries native library where the database will be backed up using the SaveFile command.

For example, to run the database migration script for a DB2 database, and to convert your inventory representation to ATP, you can use the following command:

```
./migratedb51.sh db2 MY_DB MY_INSTANCE PASS atp myhost.montreal.ca MYBACKDIR
```

Press PF6 to print your QShell output to a spooled file.

Additional scripts and files are generated when running the migration script. They are stored in the *WC55_userdir/instances/instance_name/migration* directory to

avoid file naming collisions or file overwrites, if you are migrating multiple databases which are associated with different Commerce Suite 5.1 instances.

After you run the database migration script, you should check the log files as documented in “Checking the database migration log files” on page 141. The section provides some guidance for verifying that you have successfully migrated your database.

Performance consideration with ID Resolver

If you experience poor performance in completing the ID Resolver phase of the data migration with DB2 Version 8.x, you will need to change several DB2 configuration parameters. Normally, each ID Resolver phase should take no more than several minutes to complete. If the time to complete an ID Resolver phase approaches an hour or so, you should consider the following actions to improve performance:

- Change the DB2 configuration parameter CPUSPEED to a value of -1. Enter the following DB2 update configuration command:

```
db2 -v update dbm cfg using cpuspeed -1
```

The CPU speed, in milliseconds per instruction, is used by the DB2 SQL optimizer to estimate the cost of performing certain operations. The value of this parameter is set automatically when you install the database manager based on the output from a program designed to measure CPU speed. For more information on this parameter see the DB2 Administration information at the DB2 Technical Library.

- Either increase the buffer pool size used for your database, or assign a dedicated buffer pool for the SYSCAT tablespace as follows:

```
CREATE BUFFERPOOL NEWBUFPOOL SIZE 20000 PAGESIZE 4K  
ALTER TABLESPACE SYSCATSPACE BUFFERPOOL NEWBUFPOOL
```

ID Resolver performs many intensive metadata fetches, which is very SYSCAT tablespace intensive. Allocating a new buffer pool alleviates the bottleneck. After making the above changes, you should notice a performance improvement in completing the ID Resolver phase when running the database migration script.

Restoring custom constraints

For DB2 databases, after migrating your Commerce Suite 5.1 database using the migration script, you need to restore any referential constraints that you dropped in “Dropping custom constraints” on page 57.

To run SQL statements, see “Running database scripts” on page xvi. Run the SQL statements as described in the sections below.

DB2 Databases

For DB2 databases, run the SQL statement as follows:

1. Enter the following SQL statement:

```
ALTER TABLE table_name  
  ADD CONSTRAINT constraint_name FOREIGN KEY (column_name)  
  REFERENCES foreign_table_name ON DELETE CASCADE
```

where:

- *table_name* is the name of the customized table where you need to restore the constraint.

- *constraint_name* is the name of the referential integrity constraint you wish to restore.
- *column_name* is the name of the column on which the referential constraint is being applied.
- *foreign_table_name* is the foreign table for which the referential constraint is being applied.

For examples of the SQL statements that add referential constraints, see the file `wcs.referential.sql` in the `WC55_installdir/schema/os400` directory.

Updating distinguished names

WebSphere Commerce 5.5 requires that the distinguished name (DN) column be populated for the ORGENTITY and USERS tables. You can use the `migrateDN` script to fill these tables with the recommended values except for guest users (type G). This script calls the `fillorgDN.sql` script which populates the DN column in the ORGENTITY table, and then populates the user's DN column in the USERS table. The `fillorgDN.sql` script is generated when you run the database migration script.

Before running the `migrateDN` script, do the following:

- Locate and view the contents of the `fillorgDN.sql` script to see the values for the DN column, and the values to which they will be updated. The `fillorgDN.sql` script is located in the `WC55_userdir/instances/instance_name/migration/db2` directory.
- If you do not wish to make any of the suggested updates, you will have to appropriately customize the script.

Note: If you are using an LDAP server, you may need to edit the generated `fillorgDN.sql` to ensure the distinguished names (DN) of the organizational entities are what you want. When the organizational entities are created on the LDAP server later on by WebSphere Commerce 5.5, the DN values in the ORGENTITY table are used. For example, if you would like the organizational entity named Root Organization to be created under an LDAP entry with a DN value of `c=US`, you would change the DN for the Root Organization in `fillorgDN.sql` from `o=Root Organization` to `o=Root Organization,c=US`. Distinguished names for other organizational entities will also have to be modified accordingly. For instance, change the DN entry `o=YourOrganization,o=Root Organization` to `o=YourOrganization,o=Root Organization,c=US`.

Once you are satisfied with the content of the `fillorgDN.sql` script, run the `migrateDN` script as described in the section below.

DB2 databases

To run the `migrateDN` script for DB2 databases:

1. From an iSeries command line, open a QShell window by running the `STRQSH` command.
2. Switch to the `WC55_installdir/bin` subdirectory.
3. Run the `migrateDN` script on the migrated database as follows.

```
./migrateDN.db2.sh db_name db_userID password
                    host_name
```

where:

- *db_name* is the Commerce Suite 5.1 database that has been migrated to the WebSphere Commerce 5.5 database schema level (for example, mall).
- *db_userID* is the instance user profile of the migrated database (for example, mydbuser).
- *password* is the password for the user ID to connect to the migrated database (for example, mypasswd).
- *host_name* is the fully qualified host name of the machine where the database resides.

For example, to run the script on the migrated Commerce Suite 5.1 mall database, you can use the following command:

```
./migrateDN.db2.sh mall mydbuser mypasswd myhost.montreal.ca
```

4. Check the migrateDN.log file which is generated in *WC55_userdir/instances/instance_name/migration/logs* directory. Ensure that no errors have occurred in using the script before proceeding.

Removing case sensitive search

In WebSphere Commerce 5.5.0.2, the new implementation to uniquely identify organizational entities is to ensure that the distinguished names stored for organizational entities in the database are always stored in lower case. If your distinguished names are not in lower case, you will run into problems when searching for distinguished names. The new implementation also improves the performance of finders on this organizational entities. The migration script to convert to lower case is for existing customers with organizational entities in their current database.

You must run the script to convert your distinguished names to lower case. For detailed information on running the script for your platform, see the section “Performance improvement by removing case sensitive search” in the *WebSphere Commerce Installation Guide* for WebSphere Commerce 5.5.0.2 (fix pack 2).

Assigning a master catalog

WebSphere Commerce 5.5 requires that you have a master catalog designated for each of your stores. To assign a master catalog for your store, you can run the *choosemc.sql* script that is generated when you run the database migration script. The migration script detects if you have more than one catalog in the database.

If you have only one catalog in the database, the migration script assigns this catalog as the master catalog. In this situation, the *choosemc.sql* script is not generated, and you do not need to follow the steps below.

Prior to running the *choosemc.sql* script, you will need to edit it. Locate and replace the string *MASTERCATALOG_ID* with the reference number (primary key) corresponding to the catalog that you wish to designate as your master catalog. For example, locate the following statements in the script:

```
--please replace MASTERCATALOG_ID with one of the catalog of the store you want to
designate as MasterCatalog
update storecat set mastercatalog='1' where catalog_id=MASTERCATALOG_ID
and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
values (0,MASTERCATALOG_ID,10006);
```

To choose catalog ID 6000 as your master catalog, you would update the statement as follows:

```

update storecat set mastercatalog='1' where catalog_id=6000
    and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
    values (0,6000,10006);

```

The following is an example of the output after you run the script:

```

-store :10001 has 20 catalogs.
--catalog:311000
--catalog:321000
--catalog:341000
--catalog:6000
--catalog:361000
--catalog:371000
--catalog:322000
--catalog:391000
--catalog:411000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:501000
--catalog:611000
--catalog:612000
--catalog:10001
--please replace MASTERCATALOG_ID with one of the catalog of the store you want to
designate as MasterCatalog
update storecat set mastercatalog='1' where catalog_id=6000
    and storeent_id=10001;
insert into catgrptpc (catgroup_id,catalog_id,tradeposcn_id)
    values (0,6000,10006);

```

DB2 Databases

To run the choosemc.sql script for DB2 databases:

1. Refer to “Running database scripts” on page xvi for information on running SQL scripts.
2. Run the choosemc.sql script which is located in the `WC55_userdir/instances/instance_name/migration` directory.

Changing the status of orders and order items

It is recommended, but not required that you change all your orders and order items that have a status of C to a status of S in order for you to work with orders and order items using the Commerce Accelerator tools. The considerations you need to make are described in “Orders and order items” on page 7. To change the status, you can use the ctos.sql script that is generated when you run the database migration script.

To run the script, refer to the following sections.

DB2 Databases

To run the ctos.sql script for DB2 databases:

1. Launch Operations Navigator. Refer to “Running database scripts” on page xvi.
2. Run the ctos.sql script, which is located in the `WC55_userdir/instances/instance_name/migration` directory.

Running the database consistency checker

After you have successfully run the generated migration script for your database, run the consistency checker to check the state of the migrated database.

The checker generates a report and lists the items described in “Database consistency checker output” on page 144.

Note: The consistency checker tool checks to see if any known integrity problems are present in the migrated database. Even if you run the consistency checker without receiving any error message, this does not necessarily guarantee that integrity issues are not present in the database. The tool may not cover all possible cases, especially for customized databases.

DB2 databases

To run the consistency checker for DB2 databases:

- ___ 1. From an iSeries command line, open a QShell window by running the STRQSH command.
- ___ 2. Switch to the bin subdirectory where WebSphere Commerce 5.5 is installed. For example, *WC55_installdir/bin*.
- ___ 3. Run the consistency checker on the migrated database as follows.

```
./dbchecker.sh db2 db_name db_userID password
```

where:

- *db2* represents the database management system you are using and should be *db2* for DB2 databases.
- *db_name* is the Commerce Suite 5.1 database that has been migrated to the WebSphere Commerce 5.5 database schema level, as displayed in the relational database directory.
- *db_userID* is the instance user profile of the migrated database. This is the same as the schema name.
- *password* is the password for the instance user profile used to connect to the migrated database.

For example, to run the consistency checker on the migrated Commerce Suite 5.1 database, you can use the following command below:

```
./dbchecker.sh db2 MYDB mydbuser mypasswd
```

- ___ 4. Check the *dbchecker_dbname.log* which is generated in the *WC55_userdir/instances/instance_name/migration/logs* directory. Ensure that no errors have occurred in using the script before proceeding.

Remigrating your database

If you have completed database migration and wish to remigrate the database for some reason, ensure that you remove all the files previously generated by the database migration script before you attempt to the remigrate your original database.

If the files *wcs.stage.cascade.delete.trigger.sql* or *wcs.droptrigger.sql* exist in the *WC55_userdir/instances/instance_name/migration/db2* directory, **DO NOT DELETE** it until after your migration is successfully completed. Other than these two files, you should delete the entire contents of the *WC55_userdir/instances/instance_name/migration* directory and its subdirectories. To begin the migration again, delete your migrated database, and use RSTLIB to

restore the database to its Commerce Suite 5.1 version. Be sure that your instance user profile has ownership of the restored database library and tables. Never restore a Commerce schema with a different name that it was originally created.

When remigrating the database, you need to perform all the steps in this chapter including rerunning the database preparation script as described in “Running the database preparation script” on page 55.

Chapter 7. Post instance and database migration

The following sections describe migration actions you need to take to migrate your Commerce Suite 5.1 instance after you have migrated your data and WebSphere Commerce instance to the WebSphere Commerce 5.5 level. It includes the following:

- “iSeries Apache HTTP Server”
- “Migrating your security configuration”
- “Migrating store assets” on page 70
- “Updating custom code for discount, shipping, or taxation” on page 74

iSeries Apache HTTP Server

Commerce Suite 5.1 utilized the original iSeries HTTP Server. WebSphere Commerce 5.5 uses the iSeries Apache HTTP Server. During the migration to WebSphere Commerce 5.5, WCIM creates a new Apache HTTP server (that is, a new `httpd.conf` file). It will set up any properties required by WebSphere Commerce, however it will not migrate any customized settings from the previous iSeries HTTP server. Thus, if you require any customized settings, you will need to manually retrofit them into the new Apache server after migration.

Migrating your security configuration

After you have migrated your Commerce Suite 5.1 instance configuration to the WebSphere Commerce 5.5 level, you are required to run the merchant key migration utility:

- `MigrateEncryptedInfo.sh`

This utility does the following:

1. Optionally, changes the merchant key of a specified instance and updates the encrypted data in the store’s database according to the associated configuration file, if you are not using the Commerce Suite 5.1 default merchant key.
2. Changes the way in which logon passwords are stored in the database to conform to the WebSphere Commerce 5.5 requirements.

The encrypted data that gets updated includes encrypted passwords and credit card data. The utility updates encrypted data in the following tables:

- USERREG
- PATTRVALUE
- ORDPAYINFO
- ORDPAYMTHD

The configuration file (`DBUpdate.txt`) is located in the `WC55_installdir/schema/os400/migration` directory.

The configuration file contains the information about the database access, database tables and update class used for data update. There can be multiple tables per job as long as the updated columns need to be updated with the same update class which also limits the column data types.

Java virtual machine heap size consideration

The `MigrateEncryptedInfo` script sets the maximum Java virtual machine heap size (the maximum size of memory a JVM can use for its heap in megabytes) to be 512 MB. If you notice that the script is running out of JVM heap memory, you can edit the script and increase the two occurrences of the `-Xmx512m` setting. For example, to increase the maximum heap size to 1024 MB, you would change both occurrences of this setting to `-Xmx1024m`. The maximum allowable value depends on how much physical memory your database server machine has and how much memory is used by other applications. Usually, setting the JVM maximum heap size to one-third of your physical memory is a reasonable target. There is no value to setting this parameter to less than 64m (that is the default value).

Confirm encryption setting

Before running the merchant key migration utility, confirm that the PDI encryption setting in the `instance_name.xml` file for your WebSphere Commerce instance is set correctly. Locate the `instance_name.xml` file in the `WC551_userdir/instances/instance_name/xml` directory. Edit the `instance_name.xml` file (for example, `demo.xml`), search for the string, `PDIEncrypt`, and set its value to either `on` or `off`. Setting `PDIEncrypt="on"` means that sensitive information, such as credit card data, is stored in an encrypted format using the merchant key. Setting `PDIEncrypt="off"` means that sensitive information is stored in plain text format (unencrypted).

Running the MigrateEncryptedInfo utility

Run the `MigrateEncryptedInfo` utility as follows:

- ___ 1. Stop the `instance_name` – WebSphere Commerce Server application server in the WebSphere Application Server console.
- ___ 2. From an iSeries command line, enter QSH to open a QShell window.
- ___ 3. Switch to the `WC55_installdir/bin` directory.
- ___ 4. Run the `MigrateEncryptedInfo.sh` utility as follows:
`MigrateEncryptedInfo.sh db2 [instance_name] [old_key new_key]`

where:

- `instance_name` is an optional parameter for the name of the instance which will be updated. If there is only one instance installed, this parameter can be omitted. If there are multiple instances installed, specify the name of the instance to update.

Note:

If your system has multiple instances, and you will not change the current merchant key, you must ensure that the WebSphere Commerce instance you want to migrate is the entry on the first line of the `[instance]` section in the `WC55_userdir/instances/wcs_instances` file. If you have to update the file to move the instance to the first line of the section, make, and save the changes in the `wcs_instances` file before running the `MigrateEncryptedInfo` utility.

- `old_key` is an optional parameter for the current merchant key, in clear text (ASCII) form. You should specify this parameter only if you are currently using a non-default Commerce Suite 5.1 merchant key. If you are using the default Commerce Suite 5.1 merchant key, do not specify this

parameter as the utility will detect that you are using the default key. If you specify *old_key*, you need to specify *new_key* as well.

- *new_key* is an optional parameter for the new merchant key, in clear text (ASCII) form. It must conform to the following rules:
 - Its length must be 16 hexadecimal characters. Characters can be one of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, or f.
 - It must contain a minimum of one alphabetic character.
 - It must contain a minimum of one numeric character.
 - It must be in lowercase.
 - It cannot contain more than 4 consecutive characters.

For example, `aaaa1aaaa1aaaa12` and `abcdeaaaa3aaaa12` are allowed but `aaaaabaaaa1aaaa1` is not allowed.

For examples of how to specify the command in different scenarios, see the sections below. If you do not wish to change your merchant key, omit the *old_key* and *new_key* parameters. The merchant key stored in the *instance.xml* file will be used.

If you specify *new_key*, you need to specify *old_key* as well.

- ___ 5. Restart your Web server and WebSphere Application Server:
 - a. Stop WebSphere Application Server.
 - b. Stop your Web server.
 - c. Restart your Web server.
 - d. Restart WebSphere Application Server
- ___ 6. Start the *instance_name* – WebSphere Commerce Server application server in the WebSphere Application Server console.

The tool generates three log files in the `WC55_userdir/instances` directory:

- `CCInfoMigration.log`
- `MKChangeUserAndCCInfoMigration.log`
- `MigrateEncryptedInfoError.log`

You should review the information in these log files and ensure they do not contain any error messages. Errors similar to the following can be ignored, if and only if, there is some existing data that was not encrypted with the specified *current_key*. If your data has been encrypted using different keys, you will need run the `MigrateEncryptedInfo` script multiple times, specifying a different *current_key* each time. In this case, only a subset of the encrypted data would be migrated during each pass, so there would be several of these errors in the log which you can ignore. However, the presence of such errors could also indicate a different problem, in which case the messages should not be ignored. Unintentionally specifying the incorrect key may result in these error messages, and customers might not be able to log on. :

```
%3DES-F-CIPHERINIT; Exception caught while initializing the cipher object.  
; java.lang.ArrayIndexOutOfBoundsException: 16
```

```
%3DES-F-DCRYPT; Exception caught while decrypting  
; javax.crypto.IllegalBlockSizeException: Input length (with padding)  
not multiple of 8 bytes
```

+
+
+
+

If you are migrating a Commerce Suite 5.1.1.2 or later system that contains encrypted credit card data, you can ignore the following type of error in the `MigrateEncryptedInfoError.log` file, generated when you run `MigrateEncryptedInfo`:


```
+ %3DES-F-DCRYPT; Exception caught while decrypting
+ ; javax.crypto.BadPaddingException: Given final block not properly padded
```

```
+ The number of entries should correspond with the number of encrypted credit
+ card records that exist in the ORDPAYMTHD and ORDPAYINFO tables.
```

Note that you can use this utility to update your merchant key only at migration time. If you wish to change your merchant key at a later time after migration, use the Configuration Manager to update the key. See the WebSphere Commerce 5.5 online help sections on using Configuration Manager for more information.

Examples

- If you are using a non-default Commerce Suite 5.1 merchant key, that is, you have already changed your merchant key to be unique, and you wish to update the key for WebSphere Commerce 5.5, specify both the old key and the new key:
`MigrateEncryptedInfo.sh db2 myinstance 0123456789abcdef abcdef0123456789`
- If you are using a non-default Commerce Suite 5.1 merchant key, that is, you have already changed your merchant key to be unique, and you *do not* wish to update the key for WebSphere Commerce 5.5, you do not specify either key:
`MigrateEncryptedInfo.sh db2 myinstance`

(Note that if you specify the old key and the new key with same value, you will receive an error message to that effect.)

Migrating store assets

This section describes how to migrate a published Commerce Suite 5.1 store to WebSphere Commerce.

Important

- Migrated stores will not have a `index.jsp`, which is new for WebSphere Commerce 5.5. Consequently, the WebSphere Commerce Administration Console launch store button will not work for migrated stores. In order to launch your store using the Administration Console launch store button, you need to create your own `index.jsp`. See the sample store provided with WebSphere Commerce for an example of an `index.jsp` file and refer to the WebSphere Commerce online help describing the supplied sample stores.
- You can launch your store with the Web address you used in Commerce Suite 5.1, prior to migration. For example:
`http://hostname/webapp/wcs/stores/servlet/StoreCatalogDisplay?storeId=storeId&langId=-1&catalogId=catalogId`

where *storeId* is the store ID number for your store and *catalogId* is the catalog number. See "Publishing a store archive using the command line" in the *WebSphere Commerce Production and Development online help* for more information.

Updating the Commerce Suite 5.1 JSP files

There are several changes you need to make to your Commerce Suite 5.1 JavaServer Pages (JSP files) in order to make them operational for WebSphere Commerce. The WCIM tool automatically updates the JSP files for some items. See "Instance migration using WCIM" on page 129.

Following is a list of additional changes you can make to your Commerce Suite 5.1 JSP files. Note that these changes are optional:

- You can replace some Commerce Suite 5.1 commands with the current WebSphere Commerce 5.5 version of the command. Note, that these changes are not mandatory, but are recommended to be completed at some point in time.
 - You can replace all occurrences of the `getCalculatedPrice()` method with the `getCalculatedContractPrice()` method. These methods occur in the `ItemDataBean` and `ProductDataBean` classes. For more information on these methods and classes, see the WebSphere Commerce 5.5 online help.
 - You can update the usage of the `CatalogEntryMPE` bean to the `EMarketingSpot` bean. For example, in the Commerce Suite 5.1 `newarrivals.jsp` for the WebFashion store, the `CatalogEntryMPE` bean was used as follows:

```
<%
// create the e-Marketing Spot
CatalogEntryMPE productSpot = new CatalogEntryMPE();

//LOOK: Set the right spot name.
productSpot.setName("NewArrivalsPage");
productSpot.setMaximumNumberOfItems(new Integer(20));

//Set the default list of promoted products to the
//contents of the HOMEPAGE_PROMO category.
List defaultCatalogEntryIdList = new ArrayList();
if (newArrivalCategoryId != null )
{
    CategoryDataBean subCategories[];
}
%>
```

The following is an example of the same function in the WebSphere Commerce 5.5 `newarrivals.jsp` for the WebFashion store:

```
<!-- START PROMO -->
<%
// create the e-Marketing Spot
EMarketingSpot eMarketingSpot = new EMarketingSpot();

// IMPORTANT - set the correct name here
eMarketingSpot.setName("StoreHomePage");

// instantiate the bean
DataBeanManager.activate(eMarketingSpot, request);

EMarketingSpot.CatalogEntry[] productResults = eMarketingSpot.getCatalogEntries();
if (productResults != null && productResults.length > 0)
{
    for (int i = 0; i < productResults.length; i++) {
        EMarketingSpot.CatalogEntry catalogBean = productResults[i];
        CatalogEntryDescriptionAccessBean catalogDescriptionBean =
            catalogBean.getDescription();
    }
}
%>
```

- You can use `UsablePaymentTCListDataBean` instead of `ProfileCassetteAccountDataBean` and `UserRegistrationDataBean.findUser()` instead of `UserRegistrationDataBean.getRegisterType()` to check the user type.

To see how to create a new WebSphere Commerce 5.5 store, see the *WebSphere Commerce Store Development Guide*

Changes to `shipaddress.jsp`

If you wish to launch your migrated store through the WebSphere Commerce Administration Console and have your shoppers be able to complete the shopping flow in your store, you need to make the following update to the `shipaddress.jsp` that was shipped in Commerce Suite 5.1.

Locate and change the following line from the Commerce Suite 5.1 `shipaddress.jsp`:

```
if ( !addr.getAddress1().equals("-"))
```

to:

```
if (addr.getAddress1()!=null && !addr.getAddress1().equals("-"))
```

You can update your `shipaddress.jsp` in the following directory:

`store_dir`

where `store_dir` is the directory for your store, for example, `webfashion1`.

If you have used the code from the Commerce Suite 5.1 `shipaddress.jsp` in your own application, you will need to make the corresponding change.

Shoppers should launch the store from a clean browser (that is, newly launched) after shutting down all browsers used to run any of the tools such as Administration Console.

Changes to `register.jsp` and `account.jsp`

The `register.jsp` and `account.jsp` files shipped in Commerce Suite 5.1 for the sample InFashion store do not handle the new error codes, nor do they return any errors if there is a login failure to the store.

In order to have stores return the error when a customer fails to logon, you need to update the `register.jsp` and `account.jsp`. In addition, you need to update the `infashiontext_en_US.properties` file.

For a listing of the updated JSP files and the property file see at:

- “`register.jsp`” on page 150
- “`account.jsp`” on page 155
- “`infashiontext_en_US.properties`” on page 159

The required changes to these JSP files for WebSphere Commerce relative to the original files shipped with Commerce Suite 5.1 are indicated by bold font.

If you have used the code from the Commerce Suite 5.1 `register.jsp` and `account.jsp` from InFashion in your own applications, you will need to make the corresponding changes in your code.

User registration for sample WebFashion store: For the sample Commerce Suite 5.1 WebFashion store that is available as a download from the Commerce Suite 5.1 Web site, you need to make the following change to the `register.jsp` provided. Replace the command `RegisterNAddToMemberGroup` with the command `UserRegistrationAdd`. See the WebSphere Commerce online help for usage and syntax information for the `UserRegistrationAdd` command.

This change is necessary as the `RegisterNAddToMemberGroup` command hard codes the `owner_id` to a value of 0 (zero). In WebSphere Commerce, the `owner_id` is actually a value of -2001. Without this change, you will not be able to register a new user in WebSphere Commerce.

After making the above change to `register.jsp` file, you need to change the customer profile in order to obtain discounts for the registered user. Using the Commerce Accelerator, change the `register10` customer profile for the migrated WebFashion store as follows:

1. Log into WebSphere Commerce Accelerator and select the WebFashion store.
2. Select **Marketing > Customer Profile**.
3. Select **register10** and click **Change** on the right navigation bar.
4. The Change Customer Profile page displays. On the left navigation bar, select **Registration > Registration status**.
5. Set the registration status to **Registered**.
6. Click **OK** on the bottom right corner of the screen to change the `register10` customer profile.

With Commerce Suite 5.1, when a new user is registered in the Commerce Suite 5.1 WebFashion store, the `RegisterNAddToMemberGroup` command automatically assigns the user under the member group `register10`. Thus, all the registered users for the Commerce Suite 5.1 WebFashion store belong to the `register10` member group in the `MBRGRPMBR` table. After changing the `register.jsp` file, the new registered users in the migrated WebFashion store no longer belong to the `register10` member group.

If you have used the code from the Commerce Suite 5.1 WebFashion `register.jsp` in your own applications, you will need to make the corresponding changes in your code.

Updating the Commerce Suite 5.1 JSP files

The following sections describe updates required to various JSP files from the Commerce Suite 5.1 sample stores, such as Infashion. If you reused the JSP files or your published Commerce Suite 5.1 stores, you need to make the updates as documented and redeploy the updated JSPs in your migrated stores.

Removing the `<jsp:root>` sections in JSP files

The `<jsp:root>` and `</jsp:root>` sections in your JSP files are only valid if your JSP files are XML documents. If your JSP files are not XML documents (that is they are in JavaServer Pages format), you should remove the `<jsp:root>...</jsp:root>` sections from these JSP files.

For more information on XML Documents, refer to section "JSP.5.2 JSP Documents" of the *JavaServer Pages Specification* (Version 1.2) available from Sun Microsystems.

Importing the "java.util.*" packages

In WebSphere Application Server 5.0, JSP files that use the "Vector" directive have to explicitly include `java.util.Vector` package.

If you have the following line in your JSPs, your JSPs will not require any changes to work in WebSphere Commerce 5.5.

```
<%@ page import="java.util.*" %>
```

If you do not import the `java.util` package and you use classes inside of the package, you will need to make the following changes to your JSP files. Common classes used in WebSphere Commerce are:

Enumeration

To import the specific class, use:

```
<%@ page import="java.util.Enumeration" %>
```

Vector To import the specific class, use:

```
<%@ page import="java.util.Vector" %>
```

ResourceBundle

To import the specific class, use:

```
<%@ page import="java.util.ResourceBundle" %>
```

For more information on classes provided with WebSphere Commerce 5.5, see the WebSphere Commerce Production and Development online help.

Other JSP changes

Other JSP changes include the following:

- The JSP 1.2 Specification states that the only language supported is "java". Thus, the following page language declaration in your JSPs are no longer valid.

```
<%@ page language="JAVA" %>
```

Note that the WCIM tool will convert all occurrences of `<%@ page language="JAVA" %>` to `<%@ page language="java" %>` on your behalf.

- If you are using `AbstractAccessBean.getInitContext()` to lookup the initial context in your JSPs, it is recommended that you change this to `AbstractAccessBean.getInitContext(null, null)`.

Tools XML Files Migration

The XML parser in WebSphere Commerce 5.5 enforces the XML 1.0 specifications more strictly than the previous version of the parser. As a result, you will need to update your XML files if they contain the following errors:

- `<XML>` element in `resources.xml`

The XML 1.0 specification does not allow an element name to be "XML", regardless of case-sensitivity. For example, if you use elements of the form `<XML name="xxx" file="xxx">` for XML file mappings, change it to `<resourceXML name="xxx" file="xxx">`

- Comments cannot contain two or more dashes ("--").

The XML 1.0 specifications do not allow "--" in comments. For example:

```
<!-- comment ---> <— Incorrect
```

```
<!-- comment --> <— Correct
```

```
<!-- comment ----- comment2 --> <— Incorrect
```

```
<!-- comment comment2 --> <— Correct
```

- XML headers

If your XML files do not have a correct XML header (for example, `<?xml`), or are missing the header, fix it as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Updating custom code for discount, shipping, or taxation

If you have any customized code for the discount, shipping, or taxation, you can use the following procedure to migrate your calculation code to WebSphere Commerce 5.5:

1. Identify whether there is any custom code for discount, shipping or tax.

Check the CMDREG table using the following query:

```
SELECT * FROM CMDREG WHERE INTERFACENAME LIKE '%.ApplyOrder%Cmd'
```

If there is any customized code for the discount, shipping or taxation subsystems, the result set will not be null. Check the column CLASSNAME in the CMDREG table for the class names referenced by the commands.

2. Identify whether the customized code needs to be migrated.

Check the custom implementations of the ApplyOrderXXXCmd task commands to see if they extend or reference the Commerce Suite 5.1 default implementations, that is:

- ApplyOrderAdjustmentCmdImpl
- ApplyOrderShippingChargesCmdImpl
- ApplyOrderTaxesCmdImpl
- ApplyCalculationUsagesCmdImpl — the base implementation

If they do not extend or reference the above implementations, the ApplyOrderXXXCmd task commands will continue to work in WebSphere Commerce 5.5. Note that migration of the task commands as described below is still recommended to maintain future compatibility.

3. Migrate the custom code:

a. Identify the calculation usage ID of the custom code:

discount	-1
shipping	-2
sales tax	-3
shipping tax	-4

b. Identify the corresponding default interface and implementation in WebSphere Commerce 5.5.

Check the CALMETHOD table using the following query:

```
SELECT * FROM CALMETHOD
WHERE STOREENT_ID=-1 AND
CALUSAGE_ID=calculationUsageID AND
SUBCLASS=12
```

Take note of the task command name, for example, com.ibm.commerce.order.calculation.ApplyCalculationUsageCmd for discount. The corresponding implementation is always the task command name appended with Impl, for example, com.ibm.commerce.order.calculation.ApplyCalculationUsageCmdImpl for discount.

c. Create a new interface that extends the interface in step 3b. For example:

```
package mypackage;
import com.ibm.commerce.order.calculation.*;
public interface MyApplyDiscountCmd extends ApplyCalculationUsageCmd {
    public static final String NAME = "mypackage.MyApplyDiscountCmd";
    public static final String defaultCommandClassName = NAME + Impl;
}
```

Note that the String defaultCommandClassName variable needs to match the name of the custom implementation, mypackage.MyApplyDiscountCmdImpl, in this example.

d. Modify the custom implementation to extend the implementation in step 3b and to implement the interface in step 3c. For example:

```
package mypackage;
import com.ibm.commerce.order.calculation.*;
public class MyApplyDiscountCmdImpl extends
    ApplyCalculationUsageCmdImpl implements MyApplyDiscountCmd {
}
```

- e. Modify the custom implementation to match the WebSphere Commerce 5.5 interface signature.

The `setItems()` method in WebSphere Commerce 5.5 is roughly equivalent to the `setOrderItems()` method in Commerce Suite 5.1 and there should be no need to override the default implementation of `setItems()`. Unlike Commerce Suite 5.1, the override is no longer necessary because in the `performExecute()` method, you can retrieve the list of (order) items by calling the protected method, `getItems()`.

The `Item` class is a wrapper for the `OrderItemAccessBean` class:

- To wrap an `OrderItemAccessBean` instance (or object):

```
Item item = new Item(abOrderItem);
```
- To unwrap an `OrderItemAccessBean` instance (or object):

```
OrderItemAccessBean abOrderItem = item.getOrderItem();
```
- It is recommended that your customized code interacts with `Item` instead of `OrderItemAccessBean`
- You need to commit changes to an `Item` so they can be picked up by the underlying `OrderItemAccessBean` and EJB cache:

```
item.commit();
```
- If you choose to update the `OrderItemAccessBean` directly, call `item.refresh()` to make sure that the `Item` is in synchronized with the underlying `OrderItemAccessBean` instance (or object).
- The `setCurrency()` method in WebSphere Commerce 5.5 is equivalent to the `setOrderCurrency()` method in Commerce Suite 5.1. You can retrieve the (order) currency by calling the protected method, `getCurrency()`.

Check the WebSphere Commerce 5.5 online help for the above referenced classes and interfaces for more details.

4. Register the custom interface in the CALMETHOD table. The following INSERT statement is one example of how you can do this:

```
INSERT INTO CALMETHOD
(CALMETHOD_ID, STOREENT_ID, CALUSAGE_ID, TASKNAME, DESCRIPTION, SUBCLASS, NAME)
VALUES
(calculationMethodID, storeID, -1, 'interfaceName', 'my method to apply discount',
12, 'custom ApplyCalculationUsage (discount)')
```

where:

- *calculationMethodID* is any positive number that does not conflict with another existing *calculationMethodId*, for example, 123456.
 - *interfaceName* is the one from step 3c on page 75, for example, `mypackage.MyApplyDiscountCmd`.
5. Update the STENCALUSG table to reference this calculation method. The following INSERT statement is one example of how you can do this:

```
INSERT INTO STENCALUSG
(STOREENT_ID, CALUSAGE_ID, ACTCC_CALMETHOD_ID, ACTRC_CALMETHOD_ID,
CALCODE_ID, CALMETHOD_ID_APP, CALMETHOD_ID_SUM, CALMETHOD_ID_FIN,
USAGEFLAGS, CALMETHOD_ID_INI, SEQUENCE)
VALUES
(storeID, -1, -1, -5, null, calculationMethodID, -203, null, 1, -201, 1.0)
```

where *calculationMethodID* is the one from step 4.

Migrating cache policies

WebSphere Commerce 5.5 supports caching through the dynamic cache feature of WebSphere Application Server 5.0. As a result, previous cache policies needs to be migrated. From an iSeries command line, enter QSH to open a QShell window. Run the cacheMigration script as follows, from the *WC55_installdir/bin* directory:

```
./cacheMigration.sh instance_name oldWCPath EARPath
```

where:

- *instance_name* is the instance to be migrated (for example, demo)
- *oldWCPath* is the previous Commerce Suite 5.1 installation path (typically /QIBM/UserData/CommerceSuite5 or /QIBM/UserData/WebCommerce, which holds your instance configuration files)
- *EARPath* is the WebSphere Commerce 5.5 EAR path (*WAS50_userdir/installedApps/cell_name/WC_instance_name.ear*).

For example:

```
cacheMigration.sh demo WCS51_userdir  
WAS50_userdir/installedApps/myhost/WC_demo.ear
```

Note: If you use the default WebSphere Application Server instance, the *cell_name* is the host name. If you use a non-default WebSphere Application Server instance, the *cell_name* is the *hostname_wasinstancename* by default.

The cache policy migration script generates a `cachePolicyMigration.log` file in the *WC55_userdir/instance_name/logs* directory.

Enabling WebSphere Application Server dynamic cache

In Commerce Suite 5.1, WebSphere Commerce cache pages were invalidated by a cache cleanup worker process. The WebSphere Commerce dynamic page cache used database triggers as a notification mechanism to indicate when a cached object was invalidated. These database triggers populated the CACHLOG table to indicate which cache entry needed to be removed from the cache. Once the row was committed into the CACHLOG table, the cache cleanup worker process removed the indicated page or pages on its next iteration, provided that Automatic Page Invalidation was enabled.

WebSphere Commerce 5.5 uses the new CACHEIVL table for invalidation. If you need to continue to use the previous caching mechanism to update the CACHEIVL table, that is, continue to use database triggers, see the example file `cacheTriggers.sql` provided in the *WC55_installdir/samples/dynacache/triggers/database_type* subdirectory.

WebSphere Commerce 5.5 uses the dynamic cache feature of WebSphere Application Server 5.0 as its caching mechanism. For WebSphere Application Server dynamic caching, cache entries are removed under the following circumstances:

- The timeout for the cache entry has expired.
- The cache is full and a new entry must replace an old one, based on designated priority.
- One of the cache invalidation methods has been called:
 - Command-based and servlet pathinfo based invalidation
 - Using the cache monitor

- CACHEIWL-based invalidation

For more information on cache invalidation see the chapter on Caching in the *WebSphere Commerce Administration Guide*.

For more information on WebSphere Application Server dynamic caching, see the topic "Improving performance through the dynamic cache" in the WebSphere Application Server 5.0 InfoCenter.

Note: Since WebSphere Commerce 5.5 caching mechanism utilizes the WebSphere Application Server dynamic caching facility, CacheCommand is no longer supported. Thus, any of your previous JSP files that utilized CacheCommand will need to be updated for WebSphere Commerce 5.5 to perform the functions of CacheCommand. For details on how to write the JSP files, refer to the *WebSphere Commerce Store Development Guide*.

Enabling the dynamic cache service and servlet caching

To enable caching, you should enable the dynamic cache service and configure servlet caching.

From the WebSphere Application Server administration console, click **Servers > Application Servers** in the left panel

Then (in the right panel) click the WebSphere Commerce server > **Web Container** and check **Enable servlet caching**.

For information to perform these step, see the topics "Enabling globally the dynamic cache service" and "Configuring servlet caching" in the WebSphere Application Server Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>).

Web server plug-in consideration

If it is present, remove the following line from the bootstrap.properties file located in the *WAS50_installdir/properties* directory:

```
cache.lib=WCS51_installdir/bin/51cache.dll
```

Since WebSphere Commerce 5.5 uses the WebSphere Application Server dynamic caching feature, the above line is no longer required.

Restarting the Web server and WebSphere Application Server

After completing the instance and database migration, restart your Web server and WebSphere Application Server as follows:

1. Stop your Web server, if it is started.
2. Stop WebSphere Application Server, if it is started.
3. Start the Web server.
4. Start WebSphere Application Server
5. From a Qshell session, restart the *WC_instance_name* application server by running the startServer script in the *WAS50_installdir/bin* directory.

Changing your host name(remote migration only)

If you are performing a remote migration, before you launch your migrated WebSphere Commerce 5.5 site, you will want to modify your host name for your migrated system to be the same as it was for your Commerce Suite 5.1 system prior to migration. Note that you will need to either take down your Commerce Suite 5.1 system or isolate it from your network before you bring up your renamed WebSphere Commerce 5.5 host machine to avoid conflicts.

Assuming that:

- Your migrated WebSphere Commerce 5.5 system host name is myhost2.
- You wish to change this host name to the original Commerce Suite 5.1 host name, myhost1, and ensure that WebSphere Commerce uses that host name as well.

The following are the high-level steps to do this. You will need to refer to the documentation for your operating system and network in order to use the required tools.

1. Using the tools available for your network, add the myhost1 host name and IP address to the network configuration.
2. On the WebSphere Commerce 5.5 machine, manually update the Web server configuration file, `httpd.conf`, to use myhost1 instead of myhost2.
3. In WebSphere Application Server, from the Administration Console:
 - a. Modify the Virtual Hosts to use myhost1
 - b. Modify the endpoints and HTTP transports to use myhost1
4. Manually update the WebSphere Commerce instance configuration file, `instance_name.xml`, to use myhost1 instead of myhost2.
5. Update your domain name server to reference myhost1 instead of myhost2.
6. For systems using DB2 Universal Database, log in as the DB2 user and edit the `db2nodes.cfg` file and change the myhost2 host name to the myhost1 host name. This update is required in order to start the DB2 database manager process.
7. Edit the WebSphere Application Server `WAS50_installdir/config/cells/plugin-cfg.xml` file and change the myhost2 host name to the myhost1 host name in order to be able to access the WebSphere Application Server Administrative Console.

Verifying successful instance and database migration

After you have migrated your instance using WCIM as described in Chapter 4, “Migrating your Commerce instance configuration,” on page 29 and completed the database migration in this chapter, you should verify the migration status of your instance. Once the database is successfully migrated, you should be able to start your migrated WebSphere Commerce 5.5 instance. The migrated instance will use the new WebSphere Commerce 5.5 database schema in your migrated database.

As described in the previous section, restart WebSphere Application Server 5.0 and ensure your network status is enabled. To verify that your instance and database are properly migrated, first verify all logs as previously described, and then launch the WebSphere Commerce Accelerator by pointing your Web browser to:

`https://your_hostname:8000/accelerator`

If the WebSphere Commerce Accelerator logon page displays, and you can log on to the page, your previous instance and database has been migrated to WebSphere Commerce 5.5 successfully.

Chapter 8. Migrating to Commerce Payments

This chapter describes the procedures and scenarios for migrating the following previous versions Payment Manager to the current version of Commerce Payments provided with WebSphere Commerce 5.5

- Payment Manager 2.2.x

The migration process will utilize the WebSphere Commerce 5.5 WCIM migration tool described in Chapter 4, “Migrating your Commerce instance configuration,” on page 29 and a data migration script described in this chapter.



You only have to run the Payments migration commands documented in this section if you had a previous version of WebSphere Payment Manager 2.2.1 installed and operational.

Payments migration overview

The migration of the Payments component of WebSphere Commerce is part of the overall WebSphere Commerce 5.5 migration solution. This solution provides a common procedure for you to migrate Commerce on all platforms. The Payments component, as part of this migration solution uses the WebSphere Commerce Instance Migrator (WCIM) tool. This tool provides the infrastructure required to migrate components within WebSphere Commerce and migrates an instance to the WebSphere Commerce 5.5 level. For Payments, you can migrate an earlier Payment Manager instance to a new Commerce Payments instance using this tool. The other tool that is used in the migration is the database migration tool, that provides the means and infrastructure for moving data from a previous Payments database to a new WebSphere Commerce Payments database.

Components migrated

The components of Payments that you will migrate are:

- Payments Framework
- IBM cassettes
- Vendor cassettes

The vendor cassette migration are of two categories: backward compatibility and migration. Backward compatibility allows a version 2.2.x cassette to work with WebSphere Commerce 5.5 Payments. The migration tools move the necessary assets of the payment cassette to the appropriate places in the new EAR file. Cassette migration follows the same rationale used to migrate the IBM cassette; the vendor provides a new version of their cassette and follows the new packaging and deployment capabilities for their cassettes. As part of the deployment packaging capabilities, a migration interface is provided to cassette writers to implement their internal migration and update a cassette to a new version.

Transition methodology

The key principle is to use one methodology for transition (migration of an old instance and old data to a new instance and new data) for all of WebSphere Commerce. Given this principle, Payments makes use of the tools provided by WebSphere Commerce 5.5 and no longer provides its own migration tool and engine as in previous releases. This provides the following benefits:

- Use of a common set of tools and framework
- Further integration of Payments with Commerce.
- Payments Component provides additional utilities and helpers to help the overall framework as well as tools to perform Payments-specific migration.

In addition, the methodology for migrating the old Payments files and properties to WebSphere Commerce 5.5 are followed. Previous versions of Payments, are supported as follows:

- Without an EAR file (Payment Manager 2.2.x).

Third-party cassette considerations (custom code)

Third-party cassettes data are migrated with the overall database migration of a Payments instance. Payments can handle a cassette that is incompletely migrated, that is, a cassette that is missing the required files of which Payments has no knowledge. If a cassette does not work properly at runtime because of missing files for the cassette, Payments marks it as not loadable. In this case, you need to contact the appropriate cassette vendor to help you get the cassette up and running. The intent of the migration is to migrate the items that the Payments component knows needs to be migrated in a cassette.

This list includes:

- Java archive files
 - Typical java archives that are migrated are files that follow the recommended naming conventions for cassette writers. The Payments component expects the java archive file to look similar to `eTillClasses.zip` in previous releases.
- PSPL files

The migration of third-party cassettes is modeled after the LDBCard cassette installation since most third-party cassettes have written their cassette using this IBM service asset. This installation is the model that is documented for third-party vendors to follow for creating their own directory structure and cassette setup.

For each migration, Payments will attempt to obtain the third-party cassettes to a point where they can with an existing migrated Payments instance. However; this not guaranteed as the cassettes could have files of which the Payments component is not aware.

Notes:

1. While this document covers migration, it is important to understand that existing, installed cassettes may work with a migration. However, a fresh install of a version 2.2.x cassette to WebSphere Commerce 5.5 Payments will not work because those installations are not based on the WebSphere Commerce 5.5 implementation.
2. Previous version 2.2.x cassettes are recognized in a migrated Payments instance. However, if a new instance of Payments is created, the previous 2.2.x cassettes are not recognized as Payments has no knowledge as to the table layout for these cassettes.

Premigration considerations

Before migrating to WebSphere Commerce Payments, ensure that you have addressed the following:

- All prerequisite software has been migrated or updated.

- WebSphere Application Server 5.0 has been installed, either to a separate machine, or to the same machine where WebSphere Application Server 3.5.x resides.
- WebSphere Application Server 3.5.x has been updated to WebSphere Application Server 5.0, with installed products migrated by WebSphere Application Server, with the exception of Payments.
- The application server has been exported. This is only necessary if you have customized or updated your application server settings since you first created it. To export an application server on iSeries, create the following file (say `/tmp/new.xml`):

```
<?xml version="1.0"?>
<!DOCTYPE webspere-sa-config SYSTEM "$XMLConfigDTDLocation$dsep$xmlconfig.dtd" >
<webspere-sa-config>

<node name="NodeName" action="locate">

<application-server name="PYM_INSTANCE WebSphere Payment Manager" action="export">

</application-server>

</node>
</webspere-sa-config>
```

For Payment Manager 2.2, make a backup copy of `/QIBM/UserData/PymSvr/WPMInstance/instConfig.xml`, and then run the following command from the QShell:

```
/QIBM/ProdData/WebASAdv/bin/XMLConfig -adminNodeName NodeName
-export /QIBM/UserData/PymSvr/WPMInstance/instConfig.xml
-partial /tmp/new.xml -instance WebSphereInstanceName
-nameServiceHost WASAdminHost -nameServicePort WASAdminPort
```

where `/tmp/new.xml` is the file you created above.

For details on the syntax of the XMLConfig command, refer to the WebSphere Application Server 3.5 Information Center at <http://www.ibm.com/servers/eserver/iseries/software/webspere/wsappserver/docs/as400v35/docs/>.

- You can access the previous Payment Manager and Commerce databases. If the previous Payment Manager database library is on a remote machine from where WebSphere Commerce 5.5 is installed, add a relational database directory entry for the remote system on the system where WebSphere Commerce 5.5 is installed. Use the WRKRDBDIRE CL command to work with relational database directory entries.

Note: If your pre-migrated Payments instance name is the same as your WebSphere Commerce instance name, you have to choose a different name for the Payments instance so that they do not conflict in WebSphere Commerce 5.5. Refer to Appendix B, “Renaming your Payments instance,” on page 123 for additional considerations when performing the Payments instance and database migration.

Migrating your Payments instance and database

The Payments migration process will be a three-step process using:

- The WCIM tool as described in “Migrating your Payments instance using WCIM” on page 84.
- Removing virtual hosts if required as described in “Updating the Web server ports” on page 88.

- The Payments data migration script as described in “Migrating your Payments database” on page 89.

The process follows the general migration methodology for other WebSphere Commerce components.

Migrating your Payments instance using WCIM

As in “Migrating your instance configuration using WCIM” on page 31, when you invoke WCIM to migrate Payments, you first invoke WCIM for backup, and then invoke WCIM again for migration. The basic process is similar to that described in “Migrating your instance configuration using WCIM” on page 31 but for a few additional variables that are required for Payments.

Note: The WCIM scripts, `wcim.sh` and `wcimenv.sh`, require access to the `setenv.sh` file. If you move or copy `wcim.sh` and `wcimenv.sh` from the default WebSphere Commerce `bin` subdirectory to your working directory, ensure that you move or copy the `setenv.sh` file to that working directory as well.

The main steps are as follows:

Running WCIM for backup

Before running WCIM for migration, you should run WCIM to create a backup of your current instance and instance related files:

1. Customize `wcimenv.sh` to provide the necessary migration information such as the version of your previous Payments instance, installation path, working directory name, and so on.

The following are the changes you need to make to the WCIM scripts for Payments:

- In the `wcimenv.sh` script, set the `WPM_PATH` variable to the previous Payment Manager installation directory:


```
WPM_PATH="WPM221_path"
export WPM_PATH
```
- Set the line `WCIM_MIGRATE_FROM="PM_version"` where `PM_version` is one of the following depending on the version of your previous version of Payment Manager:
 - Payment Manager 2.2.1 — WPM221

For example:

```
...

WCIM_BACKUP="true"
WCIM_INPLACE="false"
WCIM_MIGRATE_FROM="WPM221"

DB_TYPE="db2"

...

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...

WAS_PATH="WAS35_installdir"
WAS_INSTANCE="default"
WC_PATH="WCS51_installdir"
WEBSERVER_PATH="HTTPServer1312_installdir/htdocs/locale"
ANT_PATH="WAS50_installdir/lib"
```

```

WORK_DIR="WC55_userdir/temp"
MIG_FROM_WAS_INSTANCE="default"
MIG_TO_WAS_INSTANCE="default"
export MIG_FROM_WAS_INSTANCE
export MIG_TO_WAS_INSTANCE
WPM_PATH="WPM221_path"
export WPM_PATH

...

LOG_FILE="wcm_pm_backup.log"
INSTANCE="payments_instance"
PATH=WAS50_installdir/java/jre/bin:$PATH

export WAS_PATH
export WAS_INSTANCE
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH

```

2. For a remote Payments migration scenario only, you should change the following variables to your actual path on the remote Commerce Suite 5.1 machine. For example

```

WORK_DIR="WCS51_installdir/temp"
ANT_PATH="WORK_DIR/lib"
PATH=WAS35_installdir/java/jre/bin:$PATH

```

3. Run WCIM to backup the Payments instance, as follows:
 - a. From an iSeries command line, enter QSH to open a QShell window.
 - b. Change to the directory where wcm.sh is stored.
 - c. Type:

```
wcm.sh [wcmenv_path]
```

where:

- *wcmenv_path* is the path where the wcmenv.sh is located on your machine. If the wcmenv.sh path is the same as the wcmenv.sh path, use "." to specify the current directory. For example:

```
./wcm.sh .
```

4. The script backs up the version 2.2.x Payments instance files, including custom code, into the /backupInst directory on the Commerce Suite 5.1 (payments components) machine.

WCIM packages the files as a .zip file in the .../zip subdirectory of the working directory supplied in wcmenv.sh.

5. If your Payments machine is remote from WebSphere Commerce 5.5, copy the .zip file into the working directory's .../zip/ subdirectory in the WebSphere Commerce 5.5 machine. If you are backing up *in-place*, skip this step.

For a remote backup, you also need to copy the instbackupwpm221.xml file from the WC55_installdir/xml/migration directory to the .../xml/migration subdirectory of your temporary working directory.

You need to copy the following files to the .../lib subdirectory of your temporary working directory:

- eTillConfig.jar (from /QIBM/ProdData/Commerce55/Payments/V55/lib)
- xerces.jar, j2ee.jar and ant.jar (from /QIBM/ProdData/WebAS5/Base/lib)

- ConfigManager.jar and Utilities.jar (from *WC55_installdir/lib*)
- Copy *wcim.sh* and *wcimenv.sh* to your temporary working directory:

Note: WCIM requires Java version 1.2 or higher to run successfully. To check the version of Java that will be used when you run WCIM, enter `java -version` in a QShell session. If the version is below version 1.2, create a file called `SystemDefault.properties` in your home directory (as specified for your user profile -- for example, `/home/yourUserProfile`).

Add the line:

```
java.version=1.2
```

For example:

```
mkdir '/home/yourUserProfile'
edtf '/home/yourUserProfile/SystemDefault.properties
java.version=1.2
```

Press **PF3** twice to save and exit.

Running WCIM for migration

For a remote migration of your Payments instance, ensure that the new Payments database exists. This database can be empty — the data can be migrated later.

After running WCIM to backup your Payments instance, run WCIM to migrate your Payments instance and instance-related files:

1. Customize *wcimenv.sh* to provide the necessary migration information. For example:

...

```
WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="WPM221"
DB_TYPE="db2"
```

```
export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE
```

...

```
WAS_PATH="WAS50_installdir"
WC_PATH="WC55_installdir"
WEBSERVER_PATH="HTTPServer1326_installdir/htdocs/locale"
ANT_PATH="WAS50_installdir/lib"
WORK_DIR="WC55_installdir/temp"
MIG_FROM_WAS_INSTANCE="default"
MIG_TO_WAS_INSTANCE="default"
export MIG_FROM_WAS_INSTANCE
export MIG_TO_WAS_INSTANCE
WPM_PATH="/QIBM/ProdData/CommercePayments/V55"
export WPM_PATH
```

...

```
LOG_FILE="wcim_pm_migration.log"
INSTANCE="new_payments_instance_name"
PATH=$WAS_PATH/java/jre/bin:$PATH
```



```
export WC_PATH
export WEBSERVER_PATH
export ANT_PATH
export WORK_DIR
export LOG_FILE
export INSTANCE
export PATH
```

...

Notes:

- a. Add the RELATIONAL_DB_NAME to wcmenv.sh for all iSeries Payments migrations, (even when the database collection is not moved to a remote system):

```
RELATIONAL_DB_NAME=new_Relational_DB_Name
export RELATIONAL_DB_NAME
```

Additionally, in WebSphere Commerce 5.5, your schema or your Web server instance can be named differently than your Payments instance. If you wish to do this, add the appropriate lines below:

```
HTTP_INSTANCE=some_other_instance_name
export HTTP_INSTANCE
WPM_SCHEMA_NAME=some_other_instance_name
export WPM_SCHEMA_NAME
```

- b. Selecting your new instance name and schema name:

If you move your database library from one system to another during migration, you are required to keep the previous schema (library) name. There are two ways to do this:

- Use the previous Payments instance name for your new, migrated Payments instance name. INSTANCE is the same in wcmenv.sh for both backup and migration. Do not specify WPM_SCHEMA_NAME as it will default to the instance name. Specify a different instance name for migration, but specify WPM_SCHEMA_NAME to be the name of the previous Payments instance.

This note only applies when you move the database library from one system to another.

2. When running WCIM to migrate the Payments instance, run the wcm.sh script as follows:

- a. From an iSeries command line, enter QSH to open a QShell window.
- b. Change to the directory where wcm.sh is stored.
- c. Type:

```
wcm.sh wcmenv_path PM_password db_userid db_password
```

where the following are *required* parameters for Payments migration:

- *wcmenv_path* is the path where the wcmenv.sh is located on your machine. If the wcmenv.sh path is the same as the wcmenv.sh path, use "." to specify the current directory. For example:

```
./wcm.sh . mypassword
```

- *PM_password* is the WebSphere Commerce Payments instance password and is used when migrating a WebSphere Commerce Payments instance.

Note: This parameter must be in upper case. If it is not, the instance will not start and the configurator log (/QIBM/UserData/CommerceServer55/instances/Configurator.x.log) will show "ERROR: Commerce Payments instance could not be

started. Payment Servlet return codes are (56, 632)." For more information on WebSphere Commerce Payments return codes, see the *WebSphere Commerce Payments Programming Guide and Reference*.

- The *db_userid* and *db_password* parameters correspond to the previous Payment Manager database user ID and password. The parameters must be valid on both the system with the previous Payment Manager database library and the system where the migrated Payments database library will reside. (This can be the same system.)
3. When you run WCIM for migration, it does the following:
- Copies the WebSphere Commerce 5.5 instance base files into the working directory's *.../migrate* subdirectory. (This is the new EAR template to add the pre-WebSphere Commerce 5.5 EAR information, for Payments = cassettes related files.)
 - Unpacks the copied jar file into the working directory's *.../migrate* subdirectory.
 - Performs the instance migration (this is Payments instance specific):
 - Updates the WebSphere Commerce 5.5 deployment descriptors. That is, it adds unknown *.jar* files from the migrated instance to the Payments Manifest.MF file.
 - Starts collecting the configuration information from the existing Payments instance to be used later when the Payments migration creates the migrated instance. The migration process reads the current Payments information and uses it to create the new instance and the WebSphere Application Server information. No tables are created and no EAR file is deployed at this point.
 - Migrates the instance file.
 - Displays the remaining migration steps, if any, that are required after running WCIM.

Updating the Web server ports

After executing the Payments instance migration with WCIM, you should modify the Web server ports through the WebSphere Commerce Configuration Manager to avoid any port conflicts. There are two ports: the SSL port (port 5433 by default) and the non-SSL port (port 5432 by default). Port numbers used by the premigrated instance (SSL or non-SSL) will be carried forward to the new instance. If the SSL port was not set for the pre-migrated instance, the default SSL port will be used. Make sure that both of the port numbers that are assigned to the Web server for the Payments instance are not in use by other jobs. For example, if you will continue to use the premigrated instance or its Web server, and the instance shares the same host name as your migrated instance, you will need to change the port number that gets carried forward to the new instance in order to avoid a port conflict.

All Payments Web server ports should be changed through Configuration Manager to make sure that all properties and files are updated with the same information. To change WebSphere Commerce Payments ports, do the following:

1. Start Configuration Manager.
2. Click on the **Payments** component.
3. Click **Instance List** → *Instance Name* → **Instance Properties**.
4. Click on the **Web server** tab.

5. Update the desired ports. To change the SSL port, select **Use SSL** to display the SSL port. After you change the SSL port you may clear **Use SSL** if desired. The SSL port will not be displayed, but it will remain changed.
6. Click **Apply** to apply your changes.

Note: The host name will also be carried forward from the premigrated instance to the new instance. If you would like to change the host name, follow the steps above, updating Hostname in step 5, instead of the ports.

Migrating your Payments database

Before running the Payments data migration script (`migratepaymentsdb`) on your previous Payments database, it is recommended that you back it up as described in “Backing up your databases” on page 19.

Rename the database to your migrated instance name before migrating the database. Use the `RSTLIB` command, specifying an additional parameter to restore to a different library name. For example:

```
RSTLIB SAVLIB(previousPaymentsLibrary) DEV(*SAVF)
SAVF(saveFile) RSTLIB(newMigratedLibraryName)
```

where *newMigratedLibraryName* is the name that was specified for `WPM_SCHEMA_NAME` in `wcimenv.sh` during the instance migration. If `WPM_SCHEMA_NAME` was not specified, *newMigratedLibraryName* is the same as the name of your new migrated Payments instance.

If you want to move your database collection to another machine, move the *saveFile* to that machine and restore the library there. The library must be restored with the same library name as the original library:

```
RSTLIB SAVLIB(PaymentsLibraryName) DEV(*SAVF) SAVF(saveFile)
```

You must restore your database collection on a machine that is at V5R2M0 or above. This is a requirement of WebSphere Commerce 5.5. For the detailed steps to restore the library under a different library name, see Appendix C, “Restoring a Payments library under a different name when the Payments instance has been moved to a new system,” on page 125.

When you run the Payments database migration script (`migratepaymentsdb`) against your previous Payments database (as described below) the script:

1. Runs the premigration steps based on version, release level and type
2. Updates the Payments Schema to the WebSphere Commerce 5.5 level.
3. Based on the version and release, runs the Payments data migration command. This will be done for the Payments Framework and IBM Cassettes, and also for any third-party cassette that is installed before running the migration script.

DB2 databases

The following steps show you how to run the migration script for DB2 databases:

- ___ 1. From an iSeries command line, open a QShell window by running the `STRQSH` command. (You always run this command on the system where WebSphere Commerce 5.5 is installed.)
- ___ 2. Switch to the `bin` subdirectory where WebSphere Commerce 5.5 is installed. For example, `WC55_installdir/bin`.
- ___ 3. Migrate the previous Payments database by running the following command from the `WC55_installdir/bin` directory as follows:

```
./migratepaymentsdb.sh relationalDatabaseName databaseUserId  
databasePassword schemaName version
```

Set *version* to one of the following: 2.2.1.0.

Press PF6 to print your QShell output to a spooled file.

Post instance and database migration steps

The following describe migration actions you need to take to successfully migrate your Payments instance after you have completed data and instance migration.

- Grant appropriate ownership to the instance library and its tables by running the following CL commands:

```
CHGOBJOWN(schemaName) OBJTYPE(*LIB) NEWOWN(owner)
```

```
CHGOWN OBJ('/QSYS.LIB/schemaName.LIB/*') NEWOWN(owner)
```

Where

- *schemaName* is the schema name of the migrated instance. If you did not specify `WPM_SCHEMA_NAME` during instance migration, the schema name is equal to the migrated Payments instance name.
- *owner* is one of the following:
 - If your instance library resides on the system where WebSphere Commerce is installed, set the owner to `QPYMSVR`.
 - If you are using a remote database (the instance library and the Payments instance are on separate machines) set the owner to the database user ID specified during instance and database migration.

Changing to WCSRealm (recommended)

Using WCSRealm is recommended for the complete integration of WebSphere Commerce with Payments. For example, a single signon is possible when using the WebSphere Commerce Administration Console to administer Payments when using WCSRealm. Also, a WebSphere Commerce user can avoid authorization problems when placing orders in a published store when the WCSRealm is used for the Payments instance. Using WCSRealm also prevents a WebSphere Commerce user from having to maintain duplicate users in the WCSRealm and on the system.

If you would like to change the realm of your migrated Payments instance to WCSRealm, do the following:

1. From your WebSphere Administrative Console, navigate to **Application Servers** → *yourPaymentsInstance*_Commerce_Payments_Server → **Process Definition** → **Java Virtual Machine** → **Custom Properties** → **wpm.RealmClass**.
2. Set the value for `wpm.RealmClass` to `com.ibm.commerce.payment.realm.WCSRealm`.
3. Click **Apply**.
4. Save your changes to the master configuration.
5. Restart your Payments instance for the change to take effect.

To change the WCSRealm properties, such as the WebSphere Commerce host name and Web Server port, use the Configuration Manager. Note that the users defined in your previous realm will not be defined in WCSRealm. One user has been predefined for use in the WCSRealm called `wcsadmin`. You can create additional users through the WebSphere Commerce Organization Administration Console.

If you wish to continue using the change to the PSOS400Realm, see the next section.

Continuing to use PSOS400Realm (optional)

If you had been using Payment Manager without WebSphere Commerce, you were using the PSOS400Realm instead of the WCSRealm for Payment Manager users. If you would like to continue using the PSOS400Realm for Payments users, you must do the following:

1. Change the user profile QPYMWEB to be *ENABLED with a password:
`CHGUSRPRF USRPRF(QPYMWEB) STATUS(*ENABLED) PASSWORD(password)`
2. Grant QPYMWEB read authority to all the PSOS400Realm user profiles you want to have Payments access:
`GRTOBJAUT OBJ(User_Profile) OBJTYPE(*USRPRF) USER(QPYMWEB) AUT(*READ)`
3. Create an OS/400 user profile for the WebSphere Commerce site administrator. For migrated instances, this will be wcsadmin. Enable the user profile and assign the same password that is used for the wcsadmin user in the Commerce realm. Using the Payments user interface, make wcsadmin a Payments administrator.

Starting the Payments instance

To start the migrated 5.5 Payments instance, you require the Payments instance password. Therefore, you need to use Configuration Manager or the IBMPayServer script.

To start the instance from Configuration Manager:

1. Start Configuration Manager.
2. Expand **Payments** —> **Instance List**
3. Select your instance name
4. Click **Action** —> **Start Payments Instance**

When using Configuration Manager, you may clear **Password Required for startup** to change the password requirement for your migrated Payments instance. Then you may start your migrated Payments instance using the WebSphere Application Server startServer command. For more information on starting Payments instances, see the *WebSphere Commerce Installation Guide*.

To start using IBMPayServer, run the following QShell command from the /QIBM/Proddata/CommercePayments/v55/bin directory:

```
IBMPayServer Payments_instance_name Payments_instance_password
```

Payments migration usage scenarios

The following scenarios reflect the Payments-only flow that occurs during the transition to WebSphere Commerce 5.5.

Migrating the previous Payment Manager in-place

Description

To migrate Payment Manager 2.2.x to WebSphere Commerce Payments, with an in-place migration.

Performed by

Site administrator

Preconditions

- You have installed WebSphere Application Server 5.0.
- WCIM is available on the machine.
- You have started all required services.

- You can access the previous Payment Manager and Commerce databases.

Main flow

1. Copy `wcimenv.sh` to another directory in order to edit it.
2. Edit `wcimenv.sh` to supply the information necessary back up the old Payments instance. Specify `WCIM_BACKUP=true`. See “Running WCIM for in-place backup” on page 39 for more details on the information necessary for a successful backup.
3. Run `wcim.sh` from the command line, to back up the previous instance.
4. If WCIM displays errors during the instance migration process, take the corrective action antecessor and restart WCIM.
5. Edit `wcimenv.sh` on the WebSphere Commerce 5.5 machine. Specify `WCIM_BACKUP=false` and `WCIM_INPLACE=true`. See “Running WCIM for in-place migration” on page 40 for more details on the information necessary for a successful migration.
6. Run the WCIM script from the command line, passing in the Payments instance password (and optionally specifying `db_userid` and `db_password` if your migrated instance will use a remote database):

```
wcim.sh wcimenv_path payments_password [db_userid db_password]
```
7. Invoke the Payments data migration script to migrate the previous Payments database to the current level. See “Migrating your Payments database” on page 89 for the syntax to invoke the script.
8. Complete the post migration tasks in “Post instance and database migration steps” on page 90.

Post condition

The previous Payment Manager 2.2.x is successfully migrated to WebSphere Commerce Payments.

Migrating the previous Payment Manager on a remote machine

Description

To migrate Payment Manager 2.2.x to WebSphere Commerce Payments when WebSphere Commerce 5.5 is installed remotely from the previous Payment Manager.

Performed by

Site administrator

Preconditions

- You have installed WebSphere Application Server 5.0.
- WCIM is available on both the previous Payment Manager and the WebSphere Commerce 5.5 machines.
- You have started all required services.
- You can access the previous Payment Manager and Commerce databases.

Main flow

From the previous Payment Manager machine (remote system):

1. Copy `wcimenv.sh` and `wcim.sh` from the `WC55_installdir/bin` directory on the WebSphere Commerce 5.5 machine to your temporary working directory on the Commerce Suite 5.1 machine.

2. For a remote backup, you need to copy the `instbackupwpm221.xml` file from the `WC55_installdir/xml/migration` directory to the `.../xml/migration` subdirectory of your temporary working directory. You also need to copy the following files to the `.../lib` subdirectory of your temporary working directory:
 - `eTillConfig.jar` (from `/QIBM/ProdData/Commerce55/Payments/V55/lib`)
 - `xerces.jar`, `j2ee.jar` and `ant.jar` (from `/QIBM/ProdData/WebAS5/Base/lib`)
 - `ConfigManager.jar` and `Utilities.jar` (from `WC55_installdir/lib`)
3. Edit `wcimenv.sh` to supply the information necessary to back up the old Payments instance. Specify `WCIM_BACKUP=true` and `WCIM_INPLACE=false`. See “Running WCIM for remote backup” on page 35 for more details on the information necessary for a successful backup.
4. Run the `wcim.sh` command to backup the instance.
5. Copy the `backup.zip` file into the working directory’s `zip` subdirectory in the WebSphere Commerce 5.5 machine.

From the WebSphere Commerce 5.5 machine:

1. Copy `wcimenv.sh` to a temporary directory on the WebSphere Commerce 5.5 machine, before you edit it.
2. Edit `wcimenv.sh` on the WebSphere Commerce 5.5 machine. Specify `"WCIM_BACKUP=false"` and `"WCIM_INPLACE=true"`. See “Running WCIM for remote migration” on page 37 for more details on the information necessary for a successful migration.
3. Run the WCIM script from the command line, passing in the Payments instance password (optionally passing in the database user ID and password if your migrated instance will use a remote database):


```
wcim.sh wcmenv_path payments_password [db_userid db_password]
```

Note: The `payments_password` parameter must be in upper case. If it is not, the instance will not start.
4. When WCIM displays the migration log file, you should inspect the log to verify the migration.
5. WCIM displays any remaining migration steps required for Payments.
6. Invoke the Payments data migration script to migrate the previous Payments database to the current level. See “Migrating your Payments database” on page 89 for the syntax to invoke the script.
7. Complete the post migration tasks in “Post instance and database migration steps” on page 90.

Post condition

The previous Payment Manager 2.2.x is successfully migrated to WebSphere Commerce Payments.

Multiple WebSphere Commerce instances pointing to one instance of Payments

Description

To migrate Payment Manager 2.2.1 to WebSphere Commerce Payments, when multiple WebSphere Commerce instances point to one Payments instance.

Performed by

Site administrator

Preconditions

- You have installed WebSphere Application Server 5.0.
- You have installed WebSphere Commerce Payments as part of the WebSphere Commerce 5.5 installation.
- WCIM is available on both the previous Payment Manager and the WebSphere Commerce 5.5 machines (Note: They may be installed on the same machine).
- You have started all required services.
- You can access the previous Payment Manager and Commerce databases.

Main flow

1. Migrate Payments — depending on the scenario, refer to appropriate use case above.
2. For all WebSphere Commerce instances, point to the newly migrated Payments instance.
3. Choose which WebSphere Commerce instance to migrate.

Post condition

The previous Payment Manager 2.2.x is successfully migrated to WebSphere Commerce Payments.

Payments subsystem migration considerations

WebSphere Commerce 5.5 has introduced the following changes relative to Commerce Suite 5.1:

- All SET profiles (WCS51_SET_MIA and WCS51_SET_Wallet) will be in the WebSphere Commerce 5.5 database tables, but are disabled to the runtime. To enable these profiles, you need contact IBM.
- Cybercash is no longer supported and it will be disabled in WebSphere Commerce 5.5. Contact IBM for additional information.
- A new policy, Paymenttech, has been introduced:

Table 7. Paymenttech profile

Policy ID	Policy name	Profile name
-9980	Paymenttech	WC_Paymenttech

Note that all previously supplied Payments cassette profiles that continue to be supported in WebSphere Commerce 5.5 have been updated:

- All references to “Payment Manager” have been removed.
- The DTD path in the profiles now reference the DTD in the current directory rather than in *WC55_installdir/xml/PaymentManager/profile.dtd* for consistency across platforms.

If you have modified the profiles supplied in Commerce Suite 5.1, you should reapply your changes to the updated WebSphere Commerce 5.5 profiles of the same name, if you still require the change for your migrated Payments instance.

What the data migration does

The data migration script performs the following steps when you run it against your previous database:

1. The database migration copies all the standard profiles, plus the `profile.dtd` file from the `WC55_installdir/instances/default/xml/payment` directory into the `WC55_installdir/instances/instance_name/xml/payment` directory. The names of the profiles in the `.../default/xml/payment` directory for WebSphere Commerce 5.5 are:
 - WC_Paymentech (new in WebSphere Commerce 5.5)
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_OfflineCard.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_OfflineCard.profile
2. Copies all the files from the original Commerce Suite 5.1 `.../instances/instance_name/xml/payment` directory into the same directory in the WebSphere Commerce 5.5 installation, except for the following files:
 - WC51_BankServACH.profile
 - WC51_CustomOffline_BillMe.profile
 - WC51_CustomOffline_COD.profile
 - WC51_CyberCash.profile
 - WC51_OfflineCard.profile
 - WC51_SET_MIA.profile
 - WC51_SET_MIA_PCard.profile
 - WC51_SET_Wallet.profile
 - WC51_VisaNet.profile
 - WC51_VisaNet_PCard.profile
 - WCS51_CustomOffline.profile
 - WCS51_CyberCash.profile
 - WCS51_OfflineCard.profile
 - WCS51_SET_MIA.profile
 - WCS51_SET_Wallet.profile
3. Copies `profile.dtd` from the Commerce Suite 5.1 `.../xml/PaymentManager` directory to the same directory in the WebSphere Commerce 5.5 installation tree, if any files were copied in step 2, above.

That is because the Commerce Suite 5.1 versions of the profiles expect the `profile.dtd` to reside in `.../xml/PaymentManager`. The updated WebSphere Commerce 5.5 profiles expect the `profile.dtd` file to reside in the same directory as the profiles.
4. When migrating the Payment Business Policies in the POLICY table (`PolicyType_Id = 'Payment'`), the script sets the `ENDTIME` column to the `CURRENT TIME` for payment policy entries that include the String `"cassetteName=SET"` or `"cassetteName=CyberCash"` in the `PROPERTIES` column. (This effectively marks the policy as inactive.)

```

UPDATE POLICY SET ENDTIME = CURRENT TIME
WHERE PolicyType_Id = 'Payment'
AND ( Properties LIKE '%cassetteName=SET%' OR
      Properties LIKE '%cassetteName=CyberCash%' )

```

- When migrating the PAYMTHDSUP table, the script leaves out the entries with PayMthd_Id in the set.

```

SELECT PayMthd_Id FROM PAYMTHD
where PAYMTHD.ProfileName like '%SET%'
or PAYMTHD.ProfileName like '%CyberCash%'

```

The Migration script should log the PAYMTHDSUP entries that were left out from this step.

The migration to WebSphere Commerce 5.5 does *not* copy the following profiles as WebSphere Commerce 5.5 no longer supports the Cassette for SET and Cassette for CyberCash. If these cassettes and profiles are still required, contact IBM service for support:

- WC51_CyberCash.profile
- WC51_SET_MIA.profile
- WC51_SET_MIA_PCard.profile
- WC51_SET_Wallet.profile
- WCS51_CyberCash.profile
- WCS51_SET_MIA.profile
- WCS51_SET_Wallet.profile

Additional Payments subsystem migration considerations

This section documents migration considerations and actions for the Payment component.

Migrating from PAYMTHD table to payment policies

Commerce Suite 5.1 uses three database tables to define payment methods supported by a store or store group. These tables are:

PAYMTHD

The Payment Method table is a site-wide table that lists all the cashier profiles used in the mall. Each profile has a unique integer ID and a name.

PAYMTHDDSC

The Payment Method Description table is a site-wide table that contains a short description of each of the Commerce Suite 5.1 cashier profiles in each of the supported language

PAYMTHDSUP

The Payment Method Supported table lists all the profiles support by the store or store Group.

Commerce Suite 5.1 shipped with five entries in the PAYMTHD table and five cashier profiles. The five entries in the PAYMTHD table are summarized in the following table.

PAYMTHD_ID	PROFILENAME	Remark
100	WCS51_CustomOffline	Standard Commerce Suite 5.1 profile for the CustomOffline cassette.

PAYMTHD_ID	PROFILENAME	Remark
200	WCS51_OfflineCard	Standard Commerce Suite 5.1 profile for the OfflineCard cassette.
300	WCS51_SET_MIA ¹	Standard Commerce Suite 5.1 profile for the Cassette for SET (Secure Electronic Transactions) using the MIA (Merchant Initiated Authorization) SET extension.
400	WCS51_SET_Wallet ¹	Standard Commerce Suite 5.1 profile for the Cassette for SET using Wallet.
500	WCS51_CyberCash ²	Standard Commerce Suite 5.1 profile for the Cassette for CyberCash.
600	WCS51_VisaNet	Standard Commerce Suite 5.1 profile for the VisaNet cassette.

Notes:

- ¹ All SET profiles will be in the WebSphere Commerce 5.5 database tables, but are disabled to the runtime. To enable these profiles, you need contact IBM.
- ² Cybercash is no longer supported and it will be disabled in WebSphere Commerce 5.5. Contact IBM for additional information.

These payment methods are limited to payment methods supported by WebSphere Commerce Payments.

Business policies and business policy commands

WebSphere Commerce 5.5 has the the concept of *business policies* and *business policy commands* that was first introduced in WebSphere Commerce 5.4.

One category of business policies is the payment business policy (or in short, payment policy). Payment policy defines a set of business policy command interfaces that WebSphere Commerce 5.5 calls to perform business functions related to the business policy. Different payment policies may have different business policy command implementation.

WebSphere Commerce 5.5 payment policies are not limited to payment methods supported by WebSphere Commerce Payments and thus are more general than the payment methods defined for Commerce Suite 5.1.

When migrating from Commerce Suite 5.1 to WebSphere Commerce 5.5, you have to change from specifying `payMethodId` on the `OrderProcess` command to specifying `policyId` if you wish to use new WebSphere Commerce 5.5 functionality or payment behavior. Use the following table to help you make the appropriate change:

Commerce Suite 5.1		WebSphere Commerce 5.5		
Pay method ID	Profile name	Policy ID	Policy name	Profile name
100	WCS51_CustomOffline	-9991	CustomOffline_COD	WC51_CustomOffline_COD
		-9990	CustomOffline_BillMe	WC51_CustomOffline_BillMe
200	WCS51_OfflineCard	200	OfflineCard	WC51_OfflineCard

300	WCS51_SET_MIA ¹			
400	WCS51_SET_Wallet ¹			
500	WCS51_CyberCash ²			
600	WCS51_VisaNet	600	VisaNet	WC51_VisaNet
		601	VisaNet_PCard	WC51_VisaNet_PCard
		700	BankServACH	WC51_BankServACH
		-9980	Paymentech ³	WC_Paymentech ³

Notes:

- ¹ All SET profiles will be in the WebSphere Commerce 5.5 database tables, but are disabled to the runtime. To enable these profiles, you need contact IBM.
- ² Cybercash is no longer supported and it will be disabled in WebSphere Commerce 5.5. Contact IBM for additional information.
- ³ New in WebSphere Commerce 5.5.

Note that the `policyId` of the predefined payment policies is the same as the `payMethodId` of `PayMethods` used in Commerce Suite 5.1, so you can use the same value. You can choose to implement either the Commerce Suite 5.1 or WebSphere Commerce 5.5 payment behavior by using the appropriate parameters when calling the `OrderProcess` command:

- Using the `payMethodId` parameter invokes the `DoPaymentMPFCmdImpl` class to conform to the Commerce Suite 5.1 behavior using the `PAYMTHD`, `PAYMTHDDSC`, and `PAYMTHDSUP` tables. This assumes that you will not use the new functions or commands such as ATP inventory, Release To Fulfillment, Balance[®] Payment, and so on that are available in WebSphere Commerce 5.5. To use the new functions or commands, you must switch from using the `payMethodId` parameter to use the `policyId` parameter.
- Using the `policyId` parameter invokes the `DoPaymentMPFCmdImpl` class to conform to WebSphere Commerce 5.5 behavior using the `POLICY` and `POLICYCMD` tables. For example, in the migrated InFashion store, if you use ATP, replace the following line in `OrderDisplayPending.jsp`:

```
<input type=hidden name="<%= ECConstants.EC_PAYMTHDID %>" value="200">
```

with

```
<input type=hidden name="policyId" value="200">
```

Otherwise, some functions will not work such as `PickPatches` in Commerce Accelerator.

In addition, if you are using the `ProfileCassetteAccountDataBean` data bean in Commerce Suite 5.1, you should switch to use the `UsablePaymentTCListDataBean` data bean in WebSphere Commerce 5.5. The `ProfileCassetteAccountDataBean` data bean uses the Commerce Suite 5.1 tables, `PAYMTHD`, `PAYMTHDDSC`, and `PAYMTHDSUP`. The `UsablePaymentTCListDataBean` data bean uses the new WebSphere Commerce 5.5 tables, `POLICY` and `POLICYCMD`.

Processing an order using the `OrderProcessCmd` controller command invokes the `DoPaymentMPFCmdImpl` class, which is the standard implementation class for the `DoPaymentCmd` task command for WebSphere Commerce 5.5. For detailed information on how payments are processed in WebSphere Commerce 5.5, see the

sections on WebSphere Commerce Payments in the online help. For detailed information on the above interfaces, see the online help.

Note:

Business If you are migrating to a B2B store that uses contracts with specified payment terms and conditions, you can use the WebSphere Commerce 5.5 Commerce Accelerator to create the account, contracts, and the payment terms and conditions. For such cases, the `tcId` parameter is also required to identify the Payment terms and conditions to be used. You can use the `UsablePaymentTCListDataBean` data bean to return the `tcId` associated with the Payment terms and conditions.

See the WebSphere Commerce 5.5 online help for additional information on the payment policies. WebSphere Commerce 5.5 includes several other payment policies in addition to the ones listed here.

Migrating to WebSphere Commerce 5.5 business policy commands for payment

WebSphere Commerce 5.5 specifies a set of business policy command interfaces for the payment policy:

- `DoPaymentPolicyCmd`
- `CheckPaymentAcceptPolicyCmd`
- `DoDepositPolicyCmd`
- `DoRefundPolicyCmd`
- `DoCancelPolicyCmd`

Different Payment Policies may have different implementations for these commands.

WebSphere Commerce 5.5 defines two sets of implementation classes for the payment policies included with WebSphere Commerce. One set supports the payment policies that are based on the WebSphere Commerce Payments, and the other set supports the credit line payment policy which is non-WebSphere Commerce Payments based. The business policy command implementation classes for payment policies that are based on WebSphere Commerce Payments are:

- `DoPaymentPMCmdImpl`
- `CheckPaymentAcceptPMCmdImpl`
- `DoDepositPMCmdImpl`
- `DoRefundPMCmdImpl`
- `DoCancelPMCmdImpl`

The business policy command implementation classes for the credit line payment policy are:

- `DoPaymentCLCmdImpl`
- `CheckPaymentAcceptCLCmdImpl`
- `DoDepositCLCmdImpl`
- `DoRefundCLCmdImpl`
- `DoCancelCLCmdImpl`

The appropriate business policy command implementation classes are called, depending on the payment policy used.

For more information on the above implementation classes, see the WebSphere Commerce 5.5 online help.

Changing CMDREG entry for DoCancelCmd

If you use the InFashion Store in Commerce Suite 5.1 or have created your own store that uses the WebSphere Commerce Payments, you need to change the CMDREG entry for DoCancelCmd to make it work in WebSphere Commerce 5.5. In Commerce Suite 5.1, DoCancelCmd is assigned to the class `com.ibm.commerce.payment.commands.DoCancelPMCmdImpl` if WebSphere Commerce Payments is used. With WebSphere Commerce 5.5, DoCancelCmd should be assigned to class `com.ibm.commerce.payment.commands.DoCancelCmdImpl`. The `DoCancelCmdImpl` interface routes the call to `DoCancelPMCmdImpl` or `DoCancelCLCmdImpl` depending on the payment policy used for payment.

Sample JavaServer Pages file - PayStatusPM.jsp

The sample JavaServer Pages file `PayStatusPM.jsp` in the `/QIBM/ProdData/WebCommerce/samples/web/payment` directory has been updated slightly for WebSphere Commerce Payments 3.1.3. The changes are required for the JSP to generate a correct status message related to the state of the payment of an order.

In the previous release, the `PayStatusPM.jsp` treats the WebSphere Commerce Payments order state of `Refundable` as an indication that the payment for the order has progressed beyond the approved state. Therefore, it generates the status message assuring a shopper that their order "has been authorized.....". This was true with the Payment Manager cassettes shipped with Commerce Suite 5.1.

However, with WebSphere Commerce Payments 3.1.3, some of the Cassettes set the WebSphere Commerce Payments order state to the `Refundable` state as soon as the WebSphere Commerce Payments order has been created, and before any payment has been approved. This has necessitated the change in `PayStatusPM.jsp` to enable it to generate the correct status message.

The WebSphere Commerce Payments Cassettes that exhibit this behavior are:

- Cassette for CyberCash
- Cassette for VisaNet
- CustomOffline Cassette
- OfflineCard Cassette

If you have your own version of the `PayStatusPM.jsp` and you plan to use the above Cassettes with WebSphere Commerce 5.5, you should make the corresponding change to your JSP so the correct status message is displayed to the shopper when they view the page. (Remember that the WebSphere Commerce Payments order state of `Refundable` may not mean that the payment has already been *approved*.)

Refer to the `PayStatusPM.jsp` file that is supplied with WebSphere Commerce 5.5 for details. It is located in the `/QIBM/ProdData/WebCommerce/samples/web/payment` directory.

Part 2. Additional migration steps

The chapters in this part of the Migration Guide describe additional migration considerations and scenarios that may not apply to all Commerce Suite 5.1 customers. In most cases, these sections should be considered as optional steps. They include:

- Chapter 9, “Member subsystem migration,” on page 103
- Chapter 10, “Access control subsystem considerations,” on page 107
- Chapter 11, “Migrating other WebSphere Commerce components,” on page 109

Chapter 9. Member subsystem migration

This chapter describes the migration scenarios depending on how a user decides to integrate LDAP with the WebSphere Commerce 5.5 database, when migrating the Member subsystem from Commerce Suite 5.1 to WebSphere Commerce 5.5.

Important

You should always back up your database before migration or before switching from using the database to using a directory server as the Member repository. For information on backing up databases, see “Backing up your databases” on page 19.

Overview of migration procedures

This section provides an overview of the migration procedures for the Member subsystem.

The conventions when reading the table below are:

DB->DB

Migrate from a Commerce Suite 5.1 database to a WebSphere Commerce 5.5 database.

DS->DS

Migrate from a Commerce Suite 5.1 directory server to a WebSphere Commerce 5.5 directory server.

DB->DS

Migrate from a Commerce Suite 5.1 database to a WebSphere Commerce 5.5 database first, then switch to using a directory server.

The migration from using a database to using a directory server can be done at any time after release to release migration. Hence, the proper way to read the DB->DS scenario is to first read the DB->DB column, then read the DB->DS column following the number sequence.

In Commerce Suite 5.1, the `wcsadmin` user is provided in the bootstrap data for the database. However, if you used a directory server in Commerce Suite 5.1, no corresponding bootstrap was provided for `wcsadmin` in the directory server. So it is unclear in Commerce Suite 5.1 who uses a directory server where the entry for `wcsadmin` is located in the directory server. The migration procedure below does assume such an entry exists in the directory server. However, regardless of where the `wcsadmin` user actually resides in the directory server, in the `MBRREL` table which captures the membership hierarchy, the parent member of `wcsadmin` within WebSphere Commerce 5.5 will be set to the Root Organization.

The tasks done by automated migration below is only a subset of all the tasks done for the Member subsystem. There are other tasks (mainly the migration of roles and member groups) which are not listed below. More information on the **DB->DS** migration can be found in the WebSphere Commerce 5.5 online help.

Table 8. Overview of member subsystem migration procedures

Migration Procedures	DB->DB	DS->DS	DB->DS	Remarks
Create records in the BUSPROF table for business users, filling in ORG_ID and ORGUNIT_ID properly.	manual	manual	See Remarks	Already done during DB->DB migration
Fix PROFILETYPE for users in the USERS table. Refer to "Member subsystem" on page 11				
Run the automated migration script:				
Add the STATE column to the MEMBER table and populate it.	auto	auto	See Remarks	(*) Everything done by the automated script for DB ->DS has been completed during DB->DB migration.
Add Root Organization to the database.	auto	auto	see (*)	
Change ORG_ID for the wcsadmin user in the BUSPROF table from -2000 to -2001.	auto	auto	see (*)	
Change PROFILETYPE for wcsadmin in the USERS table from C to B				
Set MEMBER_ID in the ORGENTITY table for those that were null before to -2001 (Root Organization).	auto	auto	see (*)	
Create and populate the MBRREL table.	auto	auto	see (*)	
Examine DNs for organizational entities generated by automated data migration. Change the DNs (distinguished name) in the script if necessary.	n/a	manual	manual	
Run script to populate DNs for organizational entities in the ORGENTITY table and users in the USERS table	manual	manual	manual	
Create all necessary suffixes in the directory server. These are the suffixes that will be needed when organizational entities are created automatically by Commerce Suite 5.1 in the directory server.	n/a	manual	manual	
Create the ldapentry.xml file.	n/a	manual	manual	See the note below.
For DS->DS migration, ldapentry.xml should be based on the content of the ldapmap.xml file in Commerce Suite 5.1.				
For DB->DS migration, using Configuration Manager, switch to using the directory server. Manually edit the instance_name.xml file to set the MigrateUsersFromWCSdb option to ON. By default, it is set to OFF.	n/a	n/a	manual	

Note: For DS->DS migration, in creating the ldapentry.xml, you have to do something slightly different in terms of specifying the user search bases depending on how you want your users to log on. The description below differentiates between two types of users:

- *DS users* refer to those users who exist in your directory server and are users that you want WebSphere Commerce 5.5 to recognize. However, these users have never logged on to Commerce Suite 5.1 and have never been referenced in Commerce Suite 5.1. Therefore, they do not have entries in the Commerce Suite 5.1 database yet.
- *WCS DS users* refer to those users who exist in the your directory server and they are already recognized by Commerce Suite 5.1 because they already logged on to the Commerce Suite 5.1 site. Such users have entries in the Commerce Suite 5.1 database.

If you want both the DS users and WCS DS users to log on using RDN (relative distinguished name), both types of users must have RDN values such that they are all unique in the directory server. You then specify search bases such that both types of users can be found. During a search of the directory server for a user, WebSphere Commerce 5.5 expects only one user to be found. If multiple users are found, it is an error situation.

If you want your DS users and WCS DS users to have the same RDN (for example, one DS user may have DN 'uid=john,o=IBM,c=US', another WCS DS user may have DN 'uid=john,o=CompanyA,o=Root Organization; notice that both users have RDN value 'john'):

- You can have your WebSphere Commerce 5.5 DS users continue to log on with the same logon ID they used in WebSphere Commerce Suite 5.1. Have your DS users log on using a DN. You need to specify the search bases for your WCS DS users such that the search bases do not overlap with where your DS users reside.

Refer to the LDAP integration section of the WebSphere Commerce 5.5 online help for more information on DB->DS migration.

Existing Commerce Suite 5.1 user with existing directory server

In this scenario, you already have an existing directory server with some entries. You are using Commerce Suite 5.1 but not with your existing directory server. After migrating to WebSphere Commerce 5.5, you wish to use your existing directory server with WebSphere Commerce 5.5. Since you are a Commerce Suite 5.1 user, you have code running that uses only access beans to retrieve MEMBER data from the WebSphere Commerce 5.5 database.

For more information, see the section on "LDAP scenario: a database as the member repository" in the WebSphere Commerce 5.5 online help.

Continue using the Commerce Suite 5.1 directory server in WebSphere Commerce 5.5

In this scenario, you are already using a directory server with Commerce Suite 5.1. Now you migrate to WebSphere Commerce 5.5 and continue to use the directory server. As a Commerce Suite 5.1 customer, you have code running which uses only access beans to retrieve member data. Since you have used a directory server in 5.1, the registered users recognized by Commerce Suite 5.1 exist in the directory server and the users' data is replicated to the Commerce Suite 5.1 database, but organizational entities and member groups data reside only in the Commerce Suite 5.1 database.

In this scenario you need to do the following:

1. To ensure business users (B2B users) are properly placed within the membership hierarchy in WebSphere Commerce 5.4, you should ensure records exist in the BUSPROF table where necessary, with ORG_ID and ORGUNIT_ID set properly. In addition, set the profileType for users appropriately. You need to manually do the following:
 - If a business user's parent and ancestor organizational entities already exist in the Commerce Suite 5.1 database:
 - If the business user does not have a BUSPROF record, create a BUSPROF record for the business user and set ORG_ID and ORGUNIT_ID to point to the appropriate organizational entities.
 - If the business user already has a BUSPROF record, ensure the ORG_ID and ORGUNIT_ID are set properly in the BUSPROF record.
 - If a business user's parent and ancestor organizational entities do not exist in the Commerce Suite 5.1 database and you cannot create those organizational entities, consider setting the business user's profileType to C (B2C user).
2. Run the data migration script in "Migrating your database" on page 59 to:
 - Add the STATE column to the MEMBER table.
 - Change the ProfileType for the wcsadmin user from C to B.
 - Change ORG_ID in the BUSPROF table for wcsadmin from -2000 to -2001.
 - Add Root Organization to the database.
 - Fill in MEMBER_ID in the ORGENTITY table for those member Ids that were null in Commerce Suite 5.1.
 - Create the MBRREL table and populate it.
3. As part of automated data migration, a script is provided which will populate the ORGENTITY table with distinguished name (DN) values as described in "Updating distinguished names" on page 62. You should examine the DN values of the organizational entities to ensure they are what you want, and change any DN values as necessary. Then run the fillorgDN.sql script to populate the organizational entity DN values and also to populate the USERS table with DN values for registered users. Note that DN values are populated for registered users only and the DN values for the users will be replaced by WebSphere Commerce 5.5 logic later on.
4. Ensure that all suffixes that are required are properly created in the Directory server. These suffixes are needed for when WebSphere Commerce 5.5 automatically creates organizational entities in the Directory server.
5. Create an ldapentry.xml file based on ldapmap.xml, and add the mapping for organizational entity attributes into ldapentry.xml.
6. Ensure that the MigrateUsersFromWCSdb option in the instancename.xml file is OFF.

Chapter 10. Access control subsystem considerations

The access control model for WebSphere Commerce 5.5 is based upon the enforcement of access control policies. Access control policies are enforced by the access control Policy Manager. In general, when a user attempts to access a protectable resource, the access control policy manager determines if the user is allowed to perform the requested operation on the given resource.

In addition, note the following:

- The following Commerce Suite 5.1 access control database tables are deprecated in WebSphere Commerce 5.5:
 - ACCMBRGRP
 - ACCCMDGRP
 - ACCCUSTEXC
 - ACCCMDTYPE

These tables are no longer used by the server runtime to determine access control. They have been replaced with several new access control tables. See "Access Control" in the WebSphere Commerce 5.5 online help for more details.

- Any changes made to the default Commerce Suite 5.1 bootstrap access control policies (ACCCMDGRP table) will be lost. However, any additions to this table are preserved and migrated appropriately to WebSphere Commerce 5.5 by the data migration process.
- Data in the Commerce Suite 5.1, ACCCUSTEXC table has to be manually migrated. If you are using this table, see the steps in "Migrating the ACCCUSTEXC table."
- In WebSphere Commerce 5.5 the two levels of access control are:
 - Command level (coarse-grain).
Determines whether a user has access to a Controller command or view.
 - Resource level (fine-grain) — this is also known as role-based access control.
Determines whether a user can perform an action on a specific instance of a resource.

A key difference between Commerce Suite 5.1 access control and WebSphere Commerce 5.5 access control is that Commerce Suite 5.1 uses programmatic resource level access control, while WebSphere Commerce 5.5 uses policy-based, resource level access control. For more information on access control code migration considerations, see the *WebSphere Commerce Studio Migration Guide*.

Migrating the ACCCUSTEXC table

In Commerce Suite 5.1, the ACCCUSTEXC table was used to specify controller commands and views that were not to be accessible by customers in the specified organizations' stores. These commands and views were otherwise accessible by all users in the site. By default, the policy that allows all users to execute commands is AllUsersExecuteAllSiteUserCmdResourceGroup. It has the following resource group: AllSiteUserCmdResourceGroup. Assuming that the command to be excluded is part of this resource group, the following steps must be followed to maintain the same behavior after migration:

1. Create a new resource group, similar to `AllSiteUserCmdResourceGroup`, but exclude the commands (resources) that should not be accessible in this organization.
2. Create a new policy, similar to `AllUsersExecuteAllSiteUserCmdResourceGroup`, but use the new resource group from step 1.
3. Create a new action group, similar to `AllSiteUsersViews`, but exclude the views (actions) that should not be accessible in this organization.
4. Create a new policy, similar to `AllUsersExecuteAllSiteUsersViews`, but use the new action group from step 3.
5. Create a new policy group, similar to `ManagementAndAdministrationPolicyGroup`, but exclude the policies `AllUsersExecuteAllSiteUserCmdResourceGroup` and `AllUsersExecuteAllSiteUsersViews`, and include the new policies from steps 2 and 4.
6. Have this organization unsubscribe from any policy groups that include the unwanted policies: `AllUsersExecuteAllSiteUserCmdResourceGroup` and `AllUsersExecuteAllSiteUsersViews`. By default, these policies belong to the following policy group: `ManagementAndAdministrationPolicyGroup`. Thus, you would likely have to unsubscribe from that policy group.
7. Subscribe this organization to the new policy group from step 5.

Chapter 11. Migrating other WebSphere Commerce components

The following sections describe migration actions you may decide to take depending on your requirements, after you have migrated your data to the WebSphere Commerce 5.5 level. It includes the following:

- “Configuring user roles”
- “Migrating the staging server” on page 110
- “Reconfiguring the database cleanup utility” on page 110
- “Migrating the rule server configuration” on page 110
- “Auctions” on page 112
- “Business accounts and contracts” on page 113
- “Shipping Calculation Codes” on page 113
- “Format for logonID in the edit_registration page (when LDAP is used)” on page 113
- “Product Advisor migration” on page 114
- “ATP inventory rounding” on page 115
- “Default currency behavior” on page 115

Configuring user roles

Commerce Suite 5.1 used site-level registration. That is, whenever a user registered to any store on the site, they were implicitly registered to all other stores on the site. WebSphere Commerce 5.5 uses roles to determine whether the user is registered in a particular store. The customer can setup their role assignment policies so that the role gets assigned at individual stores, or for all stores under a given organization in the hierarchy.

To obtain the equivalent site level registration behavior in WebSphere Commerce 5.5, all users can simply be assigned the ‘Registered Customer’ role in the root organization. The migration process explicitly assigns all customers in the database this role. Thus, when existing customers attempt to shop at stores on the migrated site, they are allowed to access the store. For new customers, the role assignment works based on the configuration defined in the `MemberRegistrationAttributes.xml` file, located in `WC55_installdir/instances/instance_name/xml/member`.

For migration, a special version of this file is provided by WebSphere Commerce 5.5 in the `WC55_installdir/migration/wcs.security` directory. This version assigns the ‘Registered Customer’ role for a new customer into the root organization on any user registration. You can modify this file to change the role assignment policies if you wish. For more details on how to setup this file, see the topic “MemberRegistrationAttributes XML and DTD files” in the WebSphere Commerce Production and Development online help.

In order to get the correct behavior for WebSphere Commerce 5.5, you need to copy the version of `MemberRegistrationAttributes.xml` from the `WC55_installdir/migration/wcs.security` directory to the `WC55_installdir/instances/instance_name/xml/member` directory.

Migrating the staging server

In order to migrate the staging server to WebSphere Commerce 5.5, you should have already completed the following:

1. While you were at the Commerce Suite 5.1 level, you should have run either the Stage Propagate or Stage Copy utilities as documented in “Preparing the staging server for migration” on page 5 to synchronize your staging server and your production server.
2. You should have successfully migrated your Commerce Suite 5.1 production database as documented in Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59.
3. You should have successfully migrated your Commerce Suite 5.1 staging server database, also following the steps documented in Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59.

To complete the migration of your staging server:

1. Reapply any customized triggers that you dropped prior to migrating the staging server.
2. After the data migration is complete, you need to reconfigure the staging server. The migration process does not migrate the previous Commerce Suite 5.1 configuration.

Note that in the migration process, the Commerce Suite 5.1 staging server tables, are renamed by appending `_WCS51` to the original name. Thus, the Commerce Suite 5.1 staging server tables are preserved as:

- `STGSITETAB_WCS51`
- `STGMERTAB_WCS51`
- `STGMRSTTAB_WCS51`
- `STAGLOG_WCS51`

You can view the contents of these renamed tables for reference.

To reconfigure your staging server, see the section “Configuring the staging server for customized tables” in the *WebSphere Commerce Administration Guide*.

3. To copy data from the production database back to the staging server database, run the Stage Copy utility (`stagingcopy`) using the `-scope_all_` option. For details on how to run the Stage Copy utility, see the information in the section “Staging server commands” in the *WebSphere Commerce Administration Guide*.

Reconfiguring the database cleanup utility

The database migration renames the Commerce Suite 5.1 database cleanup utility table, by appending `_WCS51` to the original name. Thus, the Commerce Suite 5.1 database cleanup utility table is preserved as `CLEANCONF_WCS51`. You can view the contents of this renamed table for reference.

To reconfigure the database cleanup utility for customized database tables, see the section “Adding a new configuration to the Database Cleanup utility” in the *WebSphere Commerce Administration Guide*.

Migrating the rule server configuration

You can skip this section if:

- You had no rule services configured under Commerce Suite 5.1.

- All your rule services were created by the Campaign tools. Campaign migration is already complete as a result of completing the previous migration steps in this Guide.

The file `wcs.server`, now located in the `WC55_installdir/instances/your_instance/xml/rules` directory, was used to configure the Advisor Rule Server in Commerce Suite 5.1. In WebSphere Commerce 5.5, this configuration information is stored in the database.

To configure your rule services the same way they were configured in Commerce Suite 5.1, do the following:

1. Ensure that you have migrated your instance as documented in Chapter 4, “Migrating your Commerce instance configuration,” on page 29.
2. To view the `wcs.server` file, open it in a text editor. This is a file in XML format that is similar to the following figure:

```
<?xml version="1.0" ?>

<DeployRulesServerConfig>
  <Name>Stateless Event Poster server</Name>
  <ServerFactory>
    <JavaName>com.blazesoft.server.deploy.NdStatelessServer</JavaName>
  </ServerFactory>
  <ServiceManagerFactory>
    <JavaName>com.blazesoft.server.local.NdLocalServiceManager</JavaName>
  </ServiceManagerFactory>

  <DeployRulesServiceConfig>
    <Name>Loan Event Poster Argument Service</Name>
    <RulesServiceAgentFactoryFactory>
      <JavaName>com.blazesoft.server.rules.NdScriptRulesServiceAgentFactory</JavaName>
      <RulesProjectLoaderFactory>
        <JavaName>com.blazesoft.server.rules.NdRulesProjectFileLoader</JavaName>
        <Project>/demo/rules/ConsumerCredit_POSTER.adv</Project>
      </RulesProjectLoaderFactory>
    </RulesServiceAgentFactoryFactory>
    <DeploymentType>Java</DeploymentType>
    <DeployRulesServiceClientContextFactory>
      <JavaName>com.blazesoft.server.deploy.rules.NdDeployPosterRulesServiceClientContext</JavaName>
      <Sr1MappingClass>ScoredLoanApplication</Sr1MappingClass>
    </DeployRulesServiceClientContextFactory>
    <NumAgents>2</NumAgents>
    <RecyclePolicy>0</RecyclePolicy>
  </DeployRulesServiceConfig>

  <DeployRulesServiceConfig>
    <Name>Loan Event Poster Argument Wrapping-Results Extractor Service</Name>
    <RulesServiceAgentFactoryFactory>
      <JavaName>com.blazesoft.server.rules.NdScriptRulesServiceAgentFactory</JavaName>
      <RulesProjectLoaderFactory>
        <JavaName>com.blazesoft.server.rules.NdRulesProjectFileLoader</JavaName>
        <Project>../..../data/rules/ConsumerCredit_POSTER.adv</Project>
      </RulesProjectLoaderFactory>
    </RulesServiceAgentFactoryFactory>
    <DeploymentType>Java</DeploymentType>
    <DeployRulesServiceClientContextFactory>
      <JavaName>com.blazesoft.server.deploy.rules.NdDeployPosterRulesServiceClientContext</JavaName>
      <Sr1ArgumentsObjectFactory>
        <Sr1Class>ScoredLoanApplication</Sr1Class>
        <Sr1ObjectInitializerFunctional>
          <Sr1Name>initServiceData</Sr1Name>
          <Sr1ArgumentType>string</Sr1ArgumentType>
        </Sr1ObjectInitializerFunctional>
      </Sr1ArgumentsObjectFactory>
      <Sr1ResultExtractorFunctional>
        <Sr1Name>extractServiceResult</Sr1Name>
      </Sr1ResultExtractorFunctional>
    </DeployRulesServiceClientContextFactory>
    <NumAgents>2</NumAgents>
    <RecyclePolicy>0</RecyclePolicy>
  </DeployRulesServiceConfig>
</DeployRulesServerConfig>
```

The top-level tag is `DeployRulesServerConfig`. This entire tag represents a rule server. It contains a number of tags called `DeployRulesServiceConfig`. Each one of these tags represents a rule service. You can safely ignore the rule server tag (`DeployRulesServerConfig`). You will need to extract four pieces of information from each rule service tag (`DeployRulesServiceConfig`).

3. Launch the WebSphere Commerce 5.5 Administrator’s Console.

4. Logon and select **store**. Choose the store name and click **OK**.
5. Navigate to the Rule Service Administration tool. (**Rule Services > Administration**).
6. For each DeployRulesServiceConfig tag, do the following:
 - a. From the Rule Service Administration tool, select **Add Service**
 - b. The system presents you with four input fields:

Name of Rule Service

Enter the value of the <Name>...</Name> tag. In this example, the first rule service's name is Loan Event Poster Argument Service.

Project File Name

Enter the value of the <Project>...</Project> tag. In this example, the first rule service's project path is:

```
/demo/rules/ConsumerCredit_POSTER.adv</Project
```

Use forward slashes (/) to separate the different folder names and enter the fully-qualified path name of the .adv file for the rule project.

You should have backed up your Commerce Suite 5.1.adv files as described in "Backing up directories and files" on page 17

Number of Agents

Enter the value of the <NumAgents>...</NumAgents> tag. In our example, the first rule service's agents value is 2.

Session Timeout

Enter the value of the <ServiceSessionTimeout>...</ServiceSessionTimeout> tag, if it exists. In this example, the first rule service's session time out value is not specified. In this case, you can use the default value of 30000, meaning 30000 milliseconds or 30 seconds.

- c. Click **OK**.

The system will show you the new list of rule services, which will include the rule service you have just added. Repeat these steps for each rule service in the file wcs.server.

7. When you have finished these steps, restart the WebSphere Commerce 5.5 server and return to the Rule Service Administration tool to verify that your rule services have been successfully migrated.

In addition, for migration considerations for Rule server administration commands, see the *WebSphere Commerce Studio Migration Guide*.

Auctions

If you have enabled auctions in Commerce Suite 5.1, you need to consider the following.

All auction order items in Commerce Suite 5.1 uses the default contract (CONTRACT_ID in the CONTRACT table). After migrating to WebSphere Commerce 5.5, the auction order items are asserted with the new default contract for WebSphere Commerce 5.5. The default contract in WebSphere Commerce 5.5 may have some terms and conditions and refund policies and so on which are not suitable for auction items.

You should review whether the default contract terms and conditions are what you require for auction order items. If they are not, you need to change the contract to one that is suitable for auctions by changing TRADING_ID in the ORDERITEMS table to point to the suitable contract. By default, the database migration script sets TRADING_ID to point to the default contract that is created during the data migration.

Business accounts and contracts

The migration script does not create any business accounts. The script associates all migrated order items with the default contract. If you wish to create an account after migrating to WebSphere Commerce 5.5, you either have to specify to use the default contract or you need to modify your order items. For more information about creating an account, see the topic "Creating a new business account" in the WebSphere Commerce 5.5 online help.

The owner of StoreGroup1 has changed from the user "wcsadmin" to the organization "Root Organization". In any account or contract XML files that you want to import, if a PolicyReference refers to StoreGroup1, the Member element has to change from:

```
<Member>
  <User distinguishName ="uid=wcsadmin,o=Root Organization"/>
</Member>
```

to:

```
<Member>
  <Organization distinguishName ="o=Root Organization"/>
</Member>
```

Shipping Calculation Codes

In Commerce Suite 5.1, shipping calculation codes were calculated separately for groups of order items with different shipping addresses. That is, shipping calculation codes grouped their order items by shipping address. Grouping by shipping address is now optional behavior. To ensure backward compatible behavior, the migration script sets the perAddress flag of the CALCODE.GROUPBY column during migration for all shipping calculation codes (those where CALCODE.CALUSAGE_ID = -2).

Format for logonID in the edit_registration page (when LDAP is used)

If you are using LDAP, the logon ID in the edit registration page will be shown in DN format rather than RDN format. In order to display it to RDN format, you can use a method that is provided in the UserRegistrationDataBean to properly retrieve the logon ID. In order to use this method you will need to make the following minor change in the JSP of each store. Note that the old method is still supported and so the migrated store will still work properly, if LDAP is not used.

Locate the following code in edit_registration.jsp:

```
<%
    strLogonID = jhelper.htmlTextEncoder(bnRegister.getLogonId());
    strPassword = bnRegister.getLogonPassword();
    strFirstName = jhelper.htmlTextEncoder(bnRegister.getFirstName());
    strLastName = jhelper.htmlTextEncoder(bnRegister.getLastName());
  }
%>
```

Update the code as indicated in bold as follows:

```
<%  
    // use getAttribute("RDN") here because getLogonId() will  
    // return the DN value when LDAP is used  
    strLogonID = jhelper.htmlTextEncoder(bnRegister.getAttribute("RDN"));  
    strPassword = bnRegister.getLogonPassword();  
    strFirstName = jhelper.htmlTextEncoder(bnRegister.getFirstName());  
    strLastName = jhelper.htmlTextEncoder(bnRegister.getLastName());  
%>
```

Product Advisor migration

If you are migrating your Product Advisor configuration from previous releases, observe the following items. It is assumed that you have created a WebSphere Commerce Suite 5.1 Product Advisor search space, and that Product Advisor is operational in your previous release of WebSphere Commerce.

1. In your script to create a Product Advisor search space (for example, in the *WC55_install/samples/pa/bin* directory) you also need to make the same data type package name changes as indicated in the steps on Product Advisor code migration documented in the *WebSphere Commerce Studio Migration Guide*.

You can refer to the sample provided in WebSphere Commerce 5.5, which has been updated with the package name changes.

Note that in WebSphere Commerce 5.5, a new method of search space creation using the scheduler command `PACreateSearchSpaceBatchCmd` has been introduced that creates a search space using an XML file as input. This command can be run from the WebSphere Commerce Administration Console. In addition, a new interface has been added in the WebSphere Commerce Accelerator with which you can create a search space, define product compares, explore products, and create guided sell metaphors for store categories. The `PACreateSearchSpaceBatchCmdscheduler` command helps to automate the process for search space creation but requires an input XML file. You need to create the XML file manually. You can find sample XML files in the *WC55_installdir/samples/pa/xml* directory. For more details on this process, refer to the topic, "Product Advisor", in the WebSphere Commerce Production online help.

2. Run the script to create Product Advisor search spaces in your migrated WebSphere Commerce 5.5 system. After running the script, verify that all package names were migrated successfully by checking the `DATATYPE` column of the `ICEXPLFEAT` table. All class names in this column should have the new package name, `com.ibm.commerce.pa.datatype`.
3. Update the XML file used as input to the `PAXMLExportBatchCmd` scheduler command for creating metaphors, with the same package name changes listed in the Product Advisor code migration documented in the *WebSphere Commerce Studio Migration Guide*. You can find sample XML files in the *WC55_installdir/samples/pa/xml* directory.
4. Run the `PAXMLExportBatchCmd` scheduler command.

ATP inventory rounding

If you migrate the Commerce Suite 5.1 database using the `atp` option to convert to the ATP inventory format, the quantities in the `RECEIPT` table may contain some rounding errors. When converting to ATP inventory, the quantities in the Commerce Suite 5.1 `INVENTORY.QUANTITY` field which are declared `DOUBLE`, are converted to `INTEGER` in the `QTY*` fields of the `RECEIPT` table in WebSphere Commerce 5.5.

You need to create a tool to correct the quantities as follows:

- Convert, using the `QuantityManager`, `INVENTORY.QUANTITY` from units `INVENTORY.QUANTITYMEASURE` to units `BASEITEM.QUANTITYMEASURE`. This will give you `X`.
- Divide `X` by `BASEITEM.QUANTITYMULTIPLE` giving you `Y`.
- Round `Y` to the nearest integer and you will have the correct `RECEIPT` quantity.

Following is an example pseudo-code for such a tool:

```
QuantityAmount qa = new QuantityAmount(value, oldUnits);
QuantityAmount quantityAmount = QuantityManager.getInstance().convert(qa, newUnits);

if (quantityAmount == null) {
    throw new ECSystemException(
        ECMessage._ERR_NO_CONVERSION,
        iClassName,
        methodName,
        new Object[] { oldUnits, newUnits });
}
```

Default currency behavior

To allow your customers to select their preferred currency in which to shop, you can include a list of currencies that you support for payment on store pages. The shopping currency is determined as follows:

- It is the customer's preferred currency, if it is supported by the store. A currency is supported by the store if it is in the `CURLIST` table for the store or its store group, or if it is a language-independent or language-dependent default currency for the store or its store group, as defined in the following sentences.
- Otherwise, it is a currency supported by the store that has the customer's preferred currency as a counter value in the `CURCVLIST` table, if there is such a currency.
- Otherwise, it is the store's language-independent default currency, specified in the `SETCCURR` column of the `STOREENT` table, if one is specified for the store or its store group. Note that the concept of language-independent default currency was introduced in WebSphere Commerce 5.4 and has been carried over to WebSphere Commerce 5.5.
- Otherwise, it is the store's language-dependent default currency for the current language, specified in the `SETCCURR` column of the `STORELANG` table, if one is specified for the store or its store group.

Notes:

1. Only customers with no preferred currency or a non-supported preferred currency (which is not the counter value for a supported currency) are affected by the new language-independent default currency mentioned above. If the customer has a supported preferred currency, they will always see this currency no matter what language they choose.

2. To maintain the Commerce Suite 5.1 language-dependent default currency in WebSphere Commerce 5.5, do not set the store's default currency in the STOREENT table. To implement the WebSphere Commerce 5.5 language-dependent default currency, set the store's default currency in the STOREENT table for the Store or its StoreGroup. You do not need to change the STORELANG table parameters from Commerce Suite 5.1. Thus, you can choose which migrated stores implement the new language-dependent default currency behavior, and which stores maintain the previous language-dependent default currency behavior.

Access control policy subscriptions

WebSphere Commerce 5.5 introduces the concept of a policy group. Each policy group contains policies that are specific to a particular business requirement or access control requirement. For example, the B2B Policy Group contains policies that would be needed specifically by an organization that owned a store that wanted B2B functionality and access control behavior.

Since stores in Commerce Suite 5.1 had access to all the available functionality, in addition to the policy groups with custom policies created by the migration process, the migration script will subscribe the organization who owns the store to the following policy groups by default:

- ManagementAndAdministrationPolicyGroup
- CommonShoppingPolicyGroup
- B2CPolicyGroup
- B2BPolicyGroup

This preserves the previous behavior of your store.

After migration, if you want the store to be a B2C store, remove the subscription to the B2BPolicyGroup. If you want the store to be a B2B store, remove the subscription to the B2CPolicyGroup. For information on subscribing and unsubscribing to access control policy groups using the Organization Administration console, see the *WebSphere Commerce Administration Guide*.

If you are creating other stores with any of the new WebSphere Commerce 5.5 store types, your store may need to subscribe to a different set of policy groups. For information on which policy groups to subscribe your store, refer to the Access Control portion of the *WebSphere Commerce Security Guide*.

Part 3. Appendixes

Appendix A. Where to find more information

More information about the WebSphere Commerce system and its components is available from a variety of sources in different formats. The following sections indicate what information is available and how to access it.

WebSphere Commerce information

The following are the sources of WebSphere Commerce information:

- WebSphere Commerce online help
- WebSphere Commerce technical library

WebSphere Commerce online help

The WebSphere Commerce online information is your primary source of information for customizing, administering, and reconfiguring WebSphere Commerce. After you have installed WebSphere Commerce, you can access topics in the online information by visiting the following URL:

`https://host_name:8000/wche1p/`

where *host_name* is the fully qualified host name of your WebSphere Commerce instance.

WebSphere Commerce technical library

The WebSphere Commerce technical library is available at the following URL:

`http://www.ibm.com/software/genservers/commerce/library/`

A copy of this book, and any updated versions of this book, are available as PDF files from the Library section of the WebSphere Commerce Web site. In addition, new and updated documentation may also be available from the WebSphere Commerce technical library Web site.

WebSphere Commerce Payments information

Help for WebSphere Commerce Payments is available by clicking the following help icon:



This help icon displays on the WebSphere Commerce Payments user interface within the WebSphere Commerce Administration Console and WebSphere Commerce Accelerator and in the standalone WebSphere Commerce Payments user interface at the following URL:

`http://host_name:http_port/webapp/PaymentManager`

or

`https://host_name:ssl_port/webapp/PaymentManager`

Where the variables are defined as follows:

host_name

The fully qualified TCP/IP host name of the Web server associated with WebSphere Commerce Payments.

http_port

The HTTP port used by WebSphere Commerce Payments. The default HTTP port is 5432.

ssl_port

The SSL port used by WebSphere Commerce Payments. The default SSL port is 5433.

If WebSphere Commerce Payments is SSL-enabled, you can use either URL. If IWebSphere Commerce Payments is not SSL-enabled, you can only use the non-secure URL (http).

Help is also available at the following URL:

`http://host_name:http_port/webapp/PaymentManager/language/docenter.html`

or

`https://host_name:ssl_port/webapp/PaymentManager/language/docenter.html`

Where the variables are defined as follows:

host_name

The fully qualified TCP/IP host name of the Web server associated with WebSphere Commerce Payments.

http_port

The HTTP port used by WebSphere Commerce Payments. The default HTTP port is 5432.

ssl_port

The SSL port used by WebSphere Commerce Payments. The default SSL port is 5433.

language

A language code for the language in which the help page will be displayed. It is two letters for most languages. The language codes are as follows:

Language	Code
German	de
English	en
Spanish	es
French	fr
Italian	it
Japanese	ja
Korean	ko
Brazilian Portuguese	pt
Simplified Chinese	zh
Traditional Chinese	zh_TW

More information about WebSphere Commerce Payments and the Payments Cassettes is available at the WebSphere Commerce Technical Library:

<http://www.ibm.com/software/genservers/commerce/library/>

IBM HTTP Server information

IBM HTTP Server information is available at the IBM HTTP Server Web site:

<http://www.ibm.com/software/webservers/httservers/>

The documents are in HTML format, PDF files, or both.

WebSphere Application Server information

WebSphere Application Server information is available at the WebSphere Application Server InfoCenter:

<http://publib.boulder.ibm.com/series/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/was.htm>

DB2 Universal Database information

DB2 documentation is available at the DB2 Technical Library:

<http://www.ibm.com/software/data/db2/library/>

For information about SQL statements, refer to the DB2 Universal Database for iSeries SQL Reference at the following URL:

<http://publib.boulder.ibm.com/html/as400/infocenter.html>

Other IBM publications

You can purchase copies of most IBM publications from your IBM authorized dealer or marketing representative.

Appendix B. Renaming your Payments instance

If your pre-migrated Payments instance has the same name as your WebSphere Commerce instance, you will need to rename the Payments instance and proceed with the Payments migration as described in the steps below. Note that for the WebSphere Commerce instance you would migrate the WebSphere Commerce instance and database migration as usual as described in Chapter 4, “Migrating your Commerce instance configuration,” on page 29, Chapter 5, “Before migrating your database,” on page 53 and Chapter 6, “Migrating your Commerce Suite 5.1 database,” on page 59.

1. Decide on a new and meaningful name for the Payments instance
2. Using the WCIM tool, backup the Payments instance using the *previous* Payments name as the instance parameter for the WCIM tool. See “Running WCIM for backup” on page 84 for details on running WCIM for backup.
3. Using the WCIM tool, migrate the Payments instance using the *new* Payments name as the instance parameter. See “Running WCIM for migration” on page 86 for details on running WCIM for migration.
4. Restore the Commerce Suite 5.1 library/schema to a new library/schema with the same name as the new Payments instance name.
5. Run the Payments database migration using the new Payments instance name and the previous Payments instance password. See “Migrating your Payments database” on page 89 for details on running the Payments database migration.
6. Verify that the ports used by the Payments instance do not conflict with the ports to the WebSphere Commerce instance. (You may have to change your ports if the previous WebSphere Commerce and Payments instance shared an HTTP server, and both WebSphere Commerce and Payments utilized port 80.)

Proceed with the remaining tasks in Chapter 8, “Migrating to Commerce Payments,” on page 81

Appendix C. Restoring a Payments library under a different name when the Payments instance has been moved to a new system

The following are the steps for restoring a library under a different name when the Payments instance has been moved to a new system:

1. Restore the database from the previous system to the new system using the same library name.

```
RSTLIB SAVLIB(Payments_library_name) DEV(*SAVF) SAVF(saveFile)
```

2. Create a user profile with the same name as the new instance you are creating.

```
CRTUSRPRF USRPRF(new_instance_name) PASSWORD(*****) USRCLS(*SECOFR)  
TEXT('migration profile')
```

3. Enter the QSH command window, and run the following WebSphere Commerce 5.5 command from the /QIBM/ProdData/CommerceServer55/bin directory:

```
copyDB.sh host_name database_name new_instance_name  
new_instance_passwd old_instance_name empty_native_lib
```

4. Check the log file from copyDB

5. Delete the user profile that was created, and change the owner for the library to QPYMSVR.

```
DLTUSRPRF USRPRF(New_instance_profile) OWNBJOPT(*CHGOWN QPYMSVR) PGPOPT(*NOCHG)
```

6. Migrate the Payments database as described in “Migrating your Payments database” on page 89.

Appendix D. Extending the data migration script

WebSphere Commerce 5.5 provides two scripts that you can customize for use during the migration of your database.

If you create the files in your instance directory, prior to calling the migration script, migration will use your files, instead of the empty ones shipped in the *WC55_installdir* directory.

If you wish, you may create these files, and add your own customization here:

```
WC55_userdir/instances/instance_name/migration/db2/delyour.constraint.sql  
WC55_userdir/instances/instance_name/migration/db2/addyour.constraint.sql
```

The migration will run the first file prior to modifying the Commerce Suite 5.1 tables. The migration will then modify the tables remaining in the database, and create new ones where necessary, until the tables contain all the fields necessary for WebSphere Commerce 5.5. The second will be run after the tables are at the WebSphere Commerce 5.5 level.

Appendix E. Overview of the WCIM tool and the data migration script

This section describes the WCIM (WebSphere Commerce Instance Migration) tool and the database migration script, and is included for your information. You are not required to take any action based on this information.

Instance migration using WCIM

The WCIM tool performs the following instance migration steps (assuming the `WORK_DIR` variable in `wcimenv` is set to `WC55_userdir/temp`; note that `WORK_DIR` is user-defined):

1. Copies the WebSphere Commerce 5.5 instance template files into the `WC55_userdir/temp` directory.
2. Unpackages the copied ZIP file into the `WC55_userdir/temp` directory.
3. Performs the instance migration as follows:
 - Migrates the instance configuration file, `instance_name.xml`
 - Updates the `wcs_instances` file with instance information.
 - Migrates the Web server configuration file
 - Migrates any customized store properties files
 - Performs required modifications to migrate JavaServer Pages levels to version 1.3. (Some additional manual modifications are also required.)
 - Copies unpackaged Commerce Suite 5.1 instance files into the proper location within the WebSphere Commerce 5.5 EAR file
 - Updates the WebSphere Commerce 5.5 deployment descriptors
4. Generates a JACL script file to assist with configuring WebSphere Application Server, for example, virtual hosts, classpaths, JVM properties, and so on.
5. Generates a log file specified by the `LOG_FILE` variable in `wcimenv` in the `WORK_DIR` directory. For example:

```
SET LOG_FILE=WCIMMigration.log
```

The following are the required changes that the WCIM tool will automatically make to your JSP files. This list is derived from migrating the InFashion store from Commerce Suite 5.1 to WebSphere Commerce 5.5.

- As Commerce Suite 5.1 uses the JavaServer Page 1.1 level, you need to change all occurrences of the following:

```
<jsp:include page="<%=incfile%">"/>
```

to

```
<jsp:include page="<%=incfile%">" flush="true"/>
```

The JavaServer Page templates used in stores running in Commerce Suite 5.1 were required to support the JavaServer Page 1.0 specification. When migrating your store to WebSphere Commerce 5.5, you must ensure that the JavaServer Page templates in your store comply to the JavaServer Page 1.1 specification created by Sun Microsystems. For information about the JavaServer Page 1.1 specification, refer to Sun Microsystems' Java web site at <http://java.sun.com>.

- Change all occurrences of the following link (if it exists):

```
<Form NAME="BillAddressForm" METHOD="POST" action="<%= "OrderCopy"%>">
```

or

```
<Form NAME=BillAddressForm METHOD="POST" action="<%= "OrderCopy"%>">
```

to

```
<Form NAME="BillAddressForm" METHOD="POST" action="OrderCopy">
```

- The JSP 1.2 Specification states that the only language supported is "java". Thus, the following page language declaration in your JSPs are no longer valid.

```
<%=@ page language="JAVA" %>
```

Note that the WCIM tool will convert all occurrences of `<%=@ page language="JAVA" %>` to `<%=@ page language="java" %>` on your behalf.

Data migration script

The database migration script makes a backup copy of your database into the library that you specify, and then migrates the following subsystem or components:

- Member
- Catalog
- ATP inventory
- Order items
- Contracts
- Campaigns
- Access control

For a summary of the database schema changes between Commerce Suite 5.1 and WebSphere Commerce 5.5, see the topic "Database schema changes in this release" in the WebSphere Commerce Production and Development online help.

Member migration

The migration script performs the following updates on the Member subsystem:

- The script sets the state of the MEMBER table as follows:
 - The script sets the state to null (no state) for:
 - Guest users (user registration type set to G)
 - member groups
 - The script sets the state to Approved (1), for:
 - Registered users (user registration type set to R)
 - Site administrators (user registration type set to S)
 - Administrators (user registration type set to A)
 - Organizational entities

The possible states of the MEMBER table are as follows:

0 Pending Approval

1 Approved

2 Rejected

- Sets the OWNER_ID column of the MBRGRP table to -2001 (that is, the Root Organization) if it is not set (that is, contains a value of 0).
- Sets the users' profile type.

- For users that are registered as user type S or A (site administrators or administrative role) in Commerce Suite 5.1, the script sets the PROFILETYPE to B
- For users that have a business profile (that is, the BUSPROF table is setup) and that have a null PROFILETYPE in Commerce Suite 5.1, the script sets the PROFILETYPE to B, as it assumes they are B2B users.
For example, if the user’s PROFILETYPE is set to C (a B2C user) the script does not reset the profile type.
- Checks the MBRGRP table.
If in the MBRGRPUSG table, a custom member group is associated with the MBRGRPTYPE_ID of AccessGroup (-2), the script creates a corresponding role in the ROLE table but with the following exception:
 - If the Order Clerk or Store Developer roles are used in Commerce Suite 5.1, the migration script migrates them to user-defined roles in WebSphere Commerce 5.5.
 The script adds a record to the MBRRROLE table for each role in the WebSphere Commerce 5.5 ROLE table and sets the MEMBER_ID for these records to a value of -2001 (Root Organization). The Root Organization can access all these roles.
- Checks the ORGENTITY table and sets the parent MEMBER_ID to -2001 (Root Organization) if the MEMBER_ID is null.
- Checks all users with a register type of S. The script:
 - Creates an entry in the MBRRROLE table with the role set to -1 (Site administrator).
 - Ensures that all the parents and ancestors have the same role assigned.
- For each record in the ACCMBRGRP table, the script:
 - Adds a record to the MBRRROLE table.
 - Adds an additional record the MBRRROLE table for the parent organizational entity to which the administrator belongs. If the OWNER_ID is 0, then the script sets it to -2001.
- Builds the MBRRREL table, but only for registered users, and not for guest users.
- Generates the SQL statements to fill the DN of the ORGENTITY table.
The script generates a file called fillorgDN.sql. You can either manually update the DN (distinguished name) column of the ORGENTITY table, or use this file to update the table. See “Updating distinguished names” on page 62 for more information.

Catalog migration

The migration script performs the following updates on the Catalog subsystem:

- Creates a parent product for items that do not have any parent.
 - It creates a CATENTRY table entry of the product type; the values for this entry are copied from the item CATENTRY.
 - Creates a CATENTDESC table entry (the catalog entry description).
 - Creates the CATENTREL table entry (the product-item relationship)
- Creates fulfillment entries for products and items.
 - Creates a BASEITEM, BASEITEMDSC, ITEMVERSN and STOREITEM table entry for each product.
 - Creates an ITEMSPC and VERSIONSPC table entry for each item.
- Creates fulfillment entries for packages.

- Creates a BASEITEM, BASEITEMDSC, ITEMVERSN and STOREITEM table entry for each package.
- Creates an ITEMSPC and VERSIONSPC table entry for each package.

ATP inventory migration

The migration script performs the following updates on the Inventory subsystem:

- If you choose to migrate to the new ATP inventory representation, the script updates ALLOCATIONGOODFOR to a value of 43200 in order to enable ATP support. If you choose not to migrate to the new ATP inventory representation, the script updates ALLOCATIONGOODFOR to a value of 0 to postpone ATP inventory migration and continue tracking inventory as Commerce Suite 5.1 does, using the INVENTORY table.

Refer to the database schema documentation for the STORE table in the WebSphere Commerce 5.5 online help to understand the meaning of this value. Any nonzero value will cause ATP inventory to be enabled.

- Creates the entries for the DISTARRANG table for each product. It sets the ENDDATE to a large value (59 years) and the STARTDATE to the current date.
- Creates the entries for the ITEMFFMCTR, RADETAIL, RECEIPT, and RCTAVAIL tables for each item.
- Create the entries for the ITEMFFMCTR, DISTARRANG, RADETAIL, RECEIPT, RCTAVAIL, tables for each package.

Note that when you migrate to WebSphere Commerce 5.5 specifying `atp`, the database migration script does not populate any vendors in the RECEIPT table during the migration as there be multiple vendors per store or there may be none. If you do not have a vendor assigned to your store, you should create one using the WebSphere Commerce Accelerator before you migrate to the ATP inventory representation.

Order items migration

The migration script checks all the orders that have a status of P, I, or M.

- Looks for all the order items under those orders
- Fills in the ITEMSPC field corresponding to the CATENTRY_ID in the ORDERITEMS table.
- Fills in the TRADING_ID field of all order items to use the store's default contract.
- Generates a script to convert all orders and order items whose status is C to a status of S.

Shipping calculation codes

In Commerce Suite 5.1, shipping calculation codes were calculated separately for groups of order items with different shipping addresses. That is, shipping calculation codes grouped their order items by shipping address. Grouping by shipping address is now optional behavior. To ensure backward compatible behavior, the migration script sets the `perAddress` flag of the CALCODE.GROUPBY column for all shipping calculation codes (those where CALCODE.CALUSAGE_ID = -2).

Discount data migration

The migration script determines if discount data exists that was created by the Commerce Accelerator tool in Commerce Suite 5.1. For the Discount subsystem:

- If the script detects that discount data was created by tools other than the Commerce Suite 5.1 Commerce Accelerator, it leaves the discount data as-is. It

does not migrate the discount data to the WebSphere Commerce 5.5 level. In this case, you need to use the same tools and methods you used in Commerce Suite 5.1 to access and display your discount data.

- If the script detects that discount data was created by the Commerce Accelerator tool, it migrates the discount data in the CALCODE and CALCODEMGP tables to that required by WebSphere Commerce 5.5.
- Deletes the previous discount data if it has been migrated. Non-migrated discount data remains as-is.

Contract migration

The migration script performs the following updates on the Contract subsystem:

- If you have not used store default contract in Commerce Suite 5.1, the script creates a store default contract as required for WebSphere Commerce 5.5.
If you have used store default contract in Commerce Suite 5.1, the script migrates your store default contract to a WebSphere Commerce 5.5 store default contract. It creates the seller participant and a contract level participant if no member group pricing is available. For each TRADEPOSCN entry, if the FLAGS column is set to zero, it creates standard price contract terms and conditions. If the FLAGS column is non-zero, the script creates custom price contract terms and conditions.
- For each MGPTRDPSCN entry, it creates a trading position container level buyer participant. If MBRGRP_ID is zero, the buyer participant, MEMBER_ID, is set to null to indicate that all buyers are qualified.
- For each custom price contract terms and conditions, the script:
 - Creates a corresponding custom product set for custom price list.
 - Populates the data in the product set using the data from the price list.
- If there are multiple contracts, the script ensures that only one is activated as a default contract by setting STOREDEF.CONTRACT_ID to the appropriate value.
- Creates an entry in the TRADING table for each contract.
- Adds entries to the new STORECNTR table.

Default contract

WebSphere Commerce 5.5, introduces terms and conditions to provide contract support (originally introduced in WebSphere Commerce 5.4). The migration process will create a default contract for your system which will have the same behavior and characteristics as the WebSphere Commerce Suite 5.1 business flows (for example, for shipping charges).

Normally, you do not need to take any actions during the migration process, since the default contract is automatically created. If you need to create additional contracts for your business processes, use the WebSphere Commerce Accelerator to do so after you have migrated to WebSphere Commerce 5.5. See the section "Creating a new contract" in the WebSphere Commerce 5.5 online help.

The database migration script creates the following items for a WebSphere Commerce 5.5 default contract:

- Policy (POLICY table) and policy description (POLICYDESC table) for the migrated store:
 - A standard price policy for the master catalog (per store) — Mastercatalog TC.
 - Shipping charged by seller policy (POLICY_ID=-7001 bootstrap data)
 - Shipping charged by carrier policy (POLICY_ID=-7002 bootstrap data)

- Return charge policy -- fractional charge by days (create one per store)
- Return approval policy -- approval by days (create one per store)

In addition, the script creates four policy commands for return charge and return approval (per store)

The migration script assumes that you will use the original payment policy (policy_id=-2001 bootstrap data), thus it does not create it.

- The terms and conditions created for the default contract for the store — one shipping TC (per contract)

The returns and refund terms and conditions are not created in order that you can have your store up and running without having to create new JavaServer Pages.

The details of the returns and refund are specific to each store and is new for WebSphere Commerce 5.5. If you need to deploy this feature, you will need to create a new contract terms and conditions for your store. See the section "Creating a new contract" in the WebSphere Commerce 5.5 online help.

- Contract participants:
 - Seller participant
 - One or more buyer participants (depending on MGPTRDPSCN)
 - One contract level buyer participant

Campaign migration

The migration script migrates campaign initiatives, e-Marketing Spots, and customer profiles to the WebSphere Commerce 5.5 format. The customer profiles that were stored in the SEGMENT table are moved to the MBRGRP table. Campaign initiative rules are mined out of the BZRPENTSTG table and stored in the RULE column of the INITIATIVE table. Each condition is stored as a separate initiative. e-Marketing Spots are moved from the MPE table to the EMSPOT table. The scheduling of the campaign initiatives is moved to the INTVSCHED table.

Access Control migration

The Access Control migration involves migrating the following components:

- Migrating the Commerce Suite 5.1 ACCMBRGP table to the WebSphere Commerce 5.5 MBRROLE table.
- Migrating the Commerce Suite 5.1 ACCCMDGRP table to the WebSphere Commerce 5.5 ACPOLICY table.
- Migrating the Commerce Suite 5.1 Order Clerk and Store Developer access group as required.

The details are as follows:

1. In Commerce Suite 5.1, users are given access privileges by being assigned to various access groups (member groups with the member group type set to AccessGoup) in the ACCMBRGRP table. In WebSphere Commerce 5.5, access control policies are also partially based on access groups. However, instead of being assigned to access groups directly, it is recommended that users be assigned to roles, in the MBRROLE table. Most of the bootstrap access groups implicitly refer to role assignments. For example, the Sellers access group includes everyone that has been assigned the Seller role through the MBRROLE table.

The following table maps the Commerce Suite 5.1 access groups to the WebSphere Commerce 5.5 roles and the WebSphere Commerce 5.5 access groups. The CONDITIONS column of the MBRGRPCOND table stores the implicit conditions for belonging to an access group.

Table 9.

Commerce Suite 5.1 access group	WebSphere Commerce 5.5 role	WebSphere Commerce 5.5 access group
Site Administrator (-1)	Site Administrator (-1)	SiteAdministrators (-1)
Customer (-2)	Not needed as a role since AllUsers access group implicitly includes everyone.	AllUsers (-2)
Customer Service Representative (-3)	Customer Service Representative (-3)	CustomerServiceRepresentatives (-3)
Merchant (-4)	Seller (-4)	Sellers (-4)
Order Clerk (-5)	Not used in bootstrap	Not used in bootstrap
Store Administrator (-6)	Store Administrator (-6)	StoreAdministrators (-6)
Store Developer (-7)	Not used in bootstrap	Not used in bootstrap
Merchandising Manager (-8)	Product Manager (-8)	ProductManagers (-8)
Marketing Manager (-9)	Marketing Manager (-9)	MarketingManagers (-9)

Note: Generally, the names of roles are singular and the names of access groups are plural.

Entries in the ACCMBRGRP table are migrated to the MBRROLE table, using the above mapping of Commerce Suite 5.1 access group IDs to WebSphere Commerce 5.5 role IDs. If you have created an access group on your Commerce Suite 5.1 system, the data migration script creates a corresponding role with the same name as the access group.

- The data migration script migrates any entries that you have added to the Commerce Suite 5.1 ACCCMDGRP table appropriately to the WebSphere Commerce 5.5 ACPOLICY table. If the entry referred to a bootstrap access group, the script updates an existing WebSphere Commerce 5.5 bootstrap access control policy to include the command or view. If the entry refers to an access group that you have created in Commerce Suite 5.1, the migration script creates a new policy, along with the appropriate components: Action, ActionDescription, ActionGroup, ResourceCategory, ResourceGroup, and so on.
- The Order Clerk and Store Developer roles are not used in WebSphere Commerce 5.5. As such, WebSphere Commerce 5.5 bootstrap access control policies do not reference these roles. However, if the customer has added some customized controller commands or views that are to be accessed by these roles, then the script creates appropriate access control policies in the ACPOLICY table.

The migration script also subscribes the organization who owns a store to the required policy groups in order to preserve the previous behavior. For more information, see “Access control policy subscriptions” on page 116.

Supplemental information for data migration

The following sections provide additional reference information for the database migration tools and additional considerations for steps in the data migration process.

Database preparation script return codes

The following sections describe the return codes and the actions that are required from the database preparation script:

Mandatory database premigration items

If the database preparation script generates the following return codes, you must take the following actions before proceeding with database migration.

Return Code	Explanation/Action
-------------	--------------------

99	The database premigration tool verifies that all Commerce Suite 5.1 constraints can be properly applied to your database. Commerce Suite 5.1 runs on OS/400 V4R5; with V4R5, it is not possible to have a cascade delete referential constraint and a delete action trigger on the same database table. When staging or caching triggers are added or removed, there is a period of time where data can get inserted that violates a Commerce Suite 5.1 constraint. If there is a problem during this verification process, the tool flags the SQL statement. If the exception is flagged as 23520, then you must fix the data that violates the constraint, either by deleting the orphaned records or by creating a parent record, depending on whether you decide you need the orphaned data. Fix the problem based on the information in the log file, and then run the premigration tool again before attempting to migrate your database.
----	---

122	The script checks if your Commerce Suite 5.1 database uses the following member IDs (0 to -8) for the corresponding member group (for example, whether the member ID is -1 for the Site Administrator member group).
-----	--

0	Site Owner
-1	Site Administrator
-2	Customer
-3	Customer Service Representative
-4	Merchant
-5	Order Clerk
-6	Store Administrator
-7	Store Developer
-8	Merchandising Manager

You are required to ensure that the above member IDs correspond to the above member groups.

Notes:

1. You are required to preserve all bootstrap values, that is, the member group IDs cannot change from the above values, or else the database migration will fail.
2. Return code 122 is only applicable for English systems. If you are using a national language version other than English, you can ignore this return code for your system. National language version users need to make sure that they have not modified the bootstrap data.
3. In WebSphere Commerce 5.5, the Merchant role has been renamed to Seller, and Merchandising Manager renamed to Product Manager.

200	The script checks REFCODE field for the AUCTION table in Commerce Suite 5.1. There is a REFCODE field which is supposed to be unique,
-----	---

but the Commerce Suite 5.1 schema does not enforce it. The WebSphere Commerce 5.5 schema specifies that it is a unique index.

You are required to make sure that the REFCODE field in the AUCTION table is unique.

318 The script checks that your contract name is not more than 200 characters in length.

You are required to make sure that the data in the NAME field in the AUCTION table does not exceed 200 characters.

319 The script checks that the ORGENTITY table description is no more than 512 characters in length.

You are required to make sure that the data in the DESCRIPTION field in the ORGGRP table does not exceed 200 characters.

323 The script checks whether there are any undelivered messages in the MSGSTORE table.

You are required to run the msgviewer tool prior to running the database migration script.

324 The script checks to ensure the product QUANTITYMULTIPLE field in the CATENTSHIP table is consistent with the QUANTITYMULTIPLE field in the BASEITEM table.

If they are different, you need to ensure they are the same. See the description of the QUANTITYMULTIPLE field for the BASEITEM table in the WebSphere Commerce Production and Development online help.

325 The script checks to ensure the product QUANTITYMEASURE field in the CATENTSHIP table is consistent with the QUANTITYMEASURE field in the BASEITEM table.

If they are different, you need to ensure they are the same. See the description of the QUANTITYMEASURE field for the BASEITEM table in the WebSphere Commerce Production and Development online help.

340 The script checks that the MBRGRP table description is no more than 512 characters in length.

You are required to make sure that the data in the DESCRIPTION field in the MBRGRP table does not exceed 200 characters.

500 The LANGUAGE_ID column for one of the stores is null. Assign the correct LANGUAGE_ID to the store. Issue the following query to determine the store entry that you need to update:

```
select store_id,language_id from store
```

501 The name in one of the contracts is null. Set the contract name for the contract. Issue the following query to determine the contract entry that you need to update:

```
select contract_id,name from contract
```

502 The MEMBER_ID column for one of the contracts is null. Set the MEMBER_ID for the contract. Issue the following query to determine the contract that you need to update:

```
select member_id, contract_id from contract"
```

- 503 The NAME column in one of the TRADEPOSCN entries is null. Set the NAME column for the trading position. Issue the following query to determine the TRADEPOSCN entry that you need to update:
- ```
select tradeposcn_id,name from tradeposcnt
```
- 504 The CODE column in one of the CALCODE entries is null. Set the CODE column in the calculation code. Issue the following query to determine the CALCODE entry that you need to update:
- ```
select calcode_id, code from calcode
```

Optional database premigration items

If the database preparation script generates the following return codes, it is recommended that you take the following actions before proceeding with database migration. Although these actions are not mandatory, you should carefully consider the implications on your migrated system.

Return Code Explanation/Action

- 103 The script checks your Commerce Suite 5.1 discount data. If you have custom discount data that was created manually, that is, discount data that was not created using the **Merchandise** menu of the Commerce Suite Accelerator, the script will issue a warning. The database migration script will not migrate the discount data to WebSphere Commerce 5.5 requirements, however this data will remain in the database as-is. The WebSphere Commerce 5.5 discount tool will not be able to view the data.
- If you wish to view this discount data (which is assumed to be created outside of the Commerce Suite 5.1 tools) you need to follow the same procedure you used before to handle your discount data in Commerce Suite 5.1.
- 305 The specified users have REGISTERTYPE of "A" (Administrative user), but are not assigned to any access groups in the ACCMBRGRP table. By belonging to access groups, the user may have privileged access control rights in the site. Most administrative users belong to one or more access groups in the ACCMBRGRP table. If you feel that these users should belong to one or more access groups in the ACCMBRGRP table, consider assigning the users to access groups in the ACCMBRGRP table before migration.
- 307 If a user has a REGISTERTYPE value of S and it does not appear in the ACCMBRGRP table to have a more specific role for that user, the database migration script assigns a Site Administrator role to the user automatically during migration. Specifically, the migration script creates an entry for these users in the MBRROLE table, and assigns the Site Administrator role to their ancestor organization entries. Since Site Administrator is a very powerful role, the script issues a warning to advise you of this. The Site Administrator role should not be assigned to any organizational entities or users without careful consideration.
- 309 The Customer access group in Commerce Suite 5.1 represented the group of all users. An AllUsers member group is shipped in WebSphere Commerce 5.5 which replaces the Customer access group. If a user was assigned to the Customer access group (-2) in Commerce Suite 5.1, the migration script will explicitly assign that user to the AllUsers member group in WebSphere Commerce 5.5.

The premigration script issues a warning message as such explicit assignment is not required in the WebSphere Commerce 5.5 design. Assess whether you require such explicit assignment.

310 Commerce Suite 5.1 supported an Order Clerk role which is no longer required nor used in WebSphere Commerce 5.5. The tasks that used to be performed by the Order Clerk role have either been automated or can be performed by the Customer Service Supervisor in WebSphere Commerce 5.5. If a user has an Order Clerk role (-5) in Commerce Suite 5.1 and an entry in the ACCCMDGRP table, the user is migrated as part of the access control migration and the role is treated as a user-defined role. If no users have the Order Clerk role, the role will not be migrated.

Review if you still require the Order Role clerk. If you do not require it, remove it since it is no longer supported in WebSphere Commerce 5.5

316 The script checks for presence of parent member identifier for organizational entities in the ORGENTITY table.

You may want to fill in the parent MEMBER_ID in the ORGENTITY table for the items that are flagged. Otherwise, the database migration script will assign a value of -2001, which is the Root Organization.

317 Commerce Suite 5.1 supported a Store Developer role which is no longer required nor used in WebSphere Commerce 5.5. The tasks that used to be performed by the Store Developer role have either been automated or can be performed by the Site Administrator in WebSphere Commerce 5.5. If a user has an Store Developer role in Commerce Suite 5.1 and an entry in the ACCCMDGRP table, the user is migrated as part of the access control migration and the role is treated as a user-defined role. If no users have the Store Developer role, the role will not be migrated.

Review if you still require the Store Developer clerk. If you do not require it, remove it since it is no longer supported in WebSphere Commerce 5.5

401 The script checks for all the items that do not have a parent product. WebSphere Commerce 5.5 requires that each item have one parent product.

For any items that do not have a parent, you should add the item to the CATGPENREL table and assign a CATALOG_ID and CATGROUP_ID.

You can create a parent product for the items that are flagged. If you do not create a parent product for the items which are flagged, the database migration script will create them for you.

404 The script checks for all the items have more than one parent product. WebSphere Commerce 5.5 requires that each item can only have one parent product.

If you want to use the WebSphere Commerce 5.5 Catalog tools to view catalog data, you need to remove all parent products except for one parent product.

414 There are problems related to access control. Locate and refer to

the detailed message in the log where Access Control checking messages are located. The following are examples of such messages:

- The user `xxxx` does not have an entry in the `BUSPROF` table. The user will be migrated to `MBRROLE` but will not actually be able to perform that role without being assigned to an organization or organization unit that can perform the role.
- The owner `xxxx` for the member group defined in `ACCMBRGRP` is not an organization and will not be migrated.

415 The script checks that your orders have a status code of M (Payment initiated - the customer has initiated payment. Authorization is in process).

Make sure that `STATUS` column in the `ORDERS` table for all orders is set to M.

416 The script checks that your order items have a status code of M (Payment initiated - the customer has initiated payment. Authorization is in process).

Make sure that `STATUS` column in the `ORDERITEMS` table for all order items is set to M.

Ensuring column lengths are not exceeded

The lengths of the following columns have changed from Commerce Suite 5.1 to WebSphere Commerce 5.5. In order to ensure that you do not lose any data contained in these columns during the migration process, make sure that your data in these columns does not exceed the new length for the column prior, to data migration. For example, ensure that `MBRGRP.DESCRPTION` is not greater than 512 characters. If the existing data in the columns exceeds the new length, it will be flagged by the database preparation script.

Table.Column	Commerce Suite 5.1	WebSphere Commerce 5.5
<code>MBRGRP.DESCRPTION</code>	<code>VARGRAPHIC(4000)</code>	<code>VARGRAPHIC(512)</code>
<code>ORGENCY.DESCRPTION</code>	<code>VARGRAPHIC(4000)</code>	<code>VARGRAPHIC(512)</code>
<code>CONTRACT.NAME</code>	<code>VARGRAPHIC(254)</code>	<code>VARGRAPHIC(200)</code>
<code>USRTRAFFIC.PREVURL</code>	<code>VARGRAPHIC(4000)</code>	<code>VARGRAPHIC(254)</code>
<code>USRTRAFFIC.REFURL</code>	<code>VARGRAPHIC(4000)</code>	<code>VARGRAPHIC(254)</code>

Considerations for ATP migration

For a summary of the new ATP inventory representation system that will help you decide whether you wish to migrate or not, please refer to “Migrating to ATP inventory” on page 6. If you migrate your inventory data to ATP representation, you will need to change your customized code related to inventory data. For more information, see the WebSphere Commerce 5.5 online help. If you decide to migrate your inventory data at a later time, you need to run the `migrateATP` script as described in Appendix F, “Converting to ATP inventory at a later time,” on page 147.

Notes:

1. If you migrate to WebSphere Commerce 5.5 and with `atp` specified, the database migration script does not populate any vendors in the `RECEIPT` table

during the migration as there may be multiple vendors per store or there may be none. If you do not have a vendor assigned to your store, you should create one using the WebSphere Commerce Accelerator before you migrate to the ATP inventory representation.

2. If you choose to migrate to ATP inventory at a later time using the migrateATP script, the script creates one base item (BASEITEM) per product category entry and one specified item (ITEMSPC) per item category entry. This is normally done by the migration script if you select to migrate to ATP inventory representation during data migration.

Checking the database migration log files

After you have run the migration script, check the migration log files as follows:

1. Check the `migratedb_database_name.log` file.

This log file is generated in the `WC55_userdir/instances/instance_name/migration/logs` directory.

Ensure that no errors have occurred in running the script before proceeding. Search for the error or fail string in the log file. Also, you should check for SQLSTATE when searching the log file. If any of these strings are present, you need to analyze the message, and if necessary, fix your data and then reattempt the data migration.

You can ignore the following types of messages in the log file:

- Messages such as the following where an object did not exist and therefore could not be dropped:

```
DROP TRIGGER QSYS_TRIG_DB____PKGATTRVAL_000003 *The SQLException caught *
SQLState: 42704 Message: QSYS_TRIG_DB____PKGATTRVAL_000003 in DB type *N not found.
Vendor Code: -204
```

- Messages similar to the following that may get generated when the migration script runs the `wcs.dropview.sql` script, if the view is an undefined name. If you did not use WebSphere Catalog Manager in Commerce Suite 5.1, these views may not exist and you will receive the messages. If they do exist, these views get dropped by the migration script:

```
drop VIEW STOREINV
*The SQLException caught *

SQLState: 42704
Message: STOREINV in DB type *N not found.

Vendor Code: -204
```

```
drop VIEW wcPRODDesc
*The SQLException caught *
```

```
SQLState: 42704
Message: STOREINV in DB type *N not found.

Vendor Code: -204
```

```
drop VIEW wcPRODship
*The SQLException caught *
```

```
SQLState: 42704
STOREINV in DB type *N not found.

Vendor Code: -204
```

```
.
```

```
.
```

```
.
```

- The `defaultAccessControlPoliciesout.error.xml` file if it gets generated in the `WC55_userdir/instances/instance_name/migration/logs` directory. If this file contains only messages about duplicate records, you can ignore the messages. These messages means that the records already exist in the database, and therefore will not be loaded again.

Database migration log and trace files

The database migration script generates various log files (with the .log extension) in the *WC55_userdir/instances/instance_name/migration/logs* directory.

The log files that are generated for Commerce Suite 5.1 databases are:

premigrate_dbname.log

This log file contains all messages generated when the premigration script was run against your Commerce Suite 5.1 database. You should ensure there are no errors or warnings flagged in this log.

migratedbscripts.log

The log file for the generation of the files to handle schema changes from Commerce Suite 5.1 to WebSphere Commerce 5.5. The files that are generated in the *WC55_userdir/instances/instance_name/migration/DB2* directory are:

- *wcs.schema.create.sql* — Contains statements to rename existing tables to temporary tables, create new tables, or alter existing table, add columns and so on, and copy the content of the temporary tables to the new tables.
- *wcs.temp.drop.sql* — Contains statements that drops all temporary tables, with the exception of staging tables (that you have to migrate manually).
- *delwcs.constraint.sql* — Contains statements that delete all constraints from schema.

migratedb_database_name.log

The log file for the overall data migration including populating bootstrap data for the instance. It contains a log of the actual schema changes that were conducted, for example, dropping constraints, renaming tables, altering tables, creating new schema tables, and copying data from temporary tables to the new tables. It also flags any manipulates business logic related data

OrigSchema.log

The detailed list of Commerce Suite 5.1 tables for the from release in the database. For example, if you are migrating from WebSphere Commerce Suite database level 5.1.0.1, this log will contain a listing of all 5.1.0.1 WebSphere Commerce Suite schema tables.

TargetSchema.log

The detailed list of WebSphere Commerce 5.5 schema tables in the database after successfully running the migration script. It includes unique indexes, column definitions, and constraints. Differences between the *TargetSchema.log* and *OrigSchema.log* files will yield any customization (for example, additional columns or tables you may have added) that you may have performed on the original shipped Commerce Suite 5.1 tables.

Other log files:

The location of the following files are determined by the contents of the *WCALoggerConfig.xml* file. If this file exists in the *WC55_userdir/instances/instance_name/xml/loader* directory, it is used to determine the location of these files. Otherwise, the location is determined by the contents of *WC55_installdir/xml/loader/WCALoggerConfig.xml*.

messages.txt

Contains the Mass Loader messages.

trace.txt

Contains the trace file for the Mass Loader. Note that the size of this file is usually very large.

RESWCSID.txt

Contains messages from ID Resolver.

Viewing the migratedb log file

After running the migration script, you should review the content of the database migration log file that is generated in

WC55_userdir/instances/instance_name/migration/logs:

- *migratedb_dbname.log* — The main migration log for Commerce Suite 5.1 (creation of bootstrap data, create new schema tables, constraints, manipulate business logic related data)

Search for the string, Migrating, and check to see if Total errors=0 is present. Note that you can ignore any warning. Your migratedb.log file should appear similar to the following:

```
Migration starts... Date:2003-06-30
Migrating Discount Data...
...
...
```

```
Summary
Total changed =
Total inserted =
Total queries =
Total warnings = 0
Total errors = 0
```

```
Migrating Contract Component..
```

```
...
...
Summary
Total changed =
Total inserted =
Total queries =
Total warnings =0
Total errors = 0
```

```
Migrating Calculation Framework..
```

```
...
...
Summary
Total changed =
Total inserted =
Total queries =
Total warnings =0
Total errors = 0.
```

If you are not satisfied with the database migration for any reason, and you need to run the database migration again, you must:

1. Drop the partially migrated database schema
2. Restore the original Commerce Suite 5.1 schema from your back up device
3. Fix any problems encountered during migration
4. Begin the migration process again, starting with the pre-data migration task.

Before beginning the remigration, ensure that your instance user profile has ownership to the schema library and its tables. Also ensure that the tables are properly journaled. Dropping the database first, restoring while signed on as the owner of the database and restoring a database under its original name should ensure that authorities and journaling attributes are maintained.

Additional Checks

After checking the database migration log file, you should run the following SQL statements against your database.

- Check the FLOW table by running the following select statement:

```
SELECT * FROM FLOW
```

If the flow table is empty, then that means the Mass Loader has problems loading business flow data. You should view the `migratedb.log` and `message.txt` files for more information.

- Check the distinguished names column, DN, in the ORGENTITY and USERS tables by running the following select statements:

```
SELECT DN FROM ORGENTITY
```

```
SELECT DN FROM USERS
```

If any of the DN entries are empty, you likely did not run the `migrateDN` script for your database after migration, as described in “Updating distinguished names” on page 62.

Database consistency checker output

If the consistency checker generates the following return codes, you must take the *mandatory* actions that are listed before proceeding with the migration process for your system. Otherwise, the WebSphere Commerce 5.5 runtime will not function with the migrated data.

Return Code	Explanation/Action
401	Your migrated data contains items that do not have a parent product. You must assign a parent product for each of these items. To create a parent product for an item, add an entry to the CATENTREL table.
402	Your migrated data contains items that have more than one parent product. You must ensure that only one parent product is assigned to each item. To delete additional parent products for items that are flagged as having more than one parent product, remove entries from the CATENTREL table.
405	The migrated USER table contains organizational inconsistencies. You need to correct any items that are flagged. For a user with a record in the BUSPROF table, if the ORG_ID and the ORGUNIT_ID columns are not null, starting with ORGUNIT_ID, use the ORGENTITY table to traverse up the organization hierarchy until MEMBER_ID eventually contains null or contains the same value as ORGENTITY_ID. The value of the MEMBER_ID column in ORGENTITY is not the same as the ORG_ID column in BUSPROF. Make sure that the value of the MEMBER_ID column in ORGENTITY is the same as the ORG_ID column in BUSPROF.
500	None of your catalogs are designated as a master catalog. Ensure that one of your catalogs are designated as a master catalog if you have multiple catalogs per store.

Design a master catalog, and choose a master catalog by running the `choosemc.sql` script as documented in “Assigning a master catalog” on page 63.

- 503** Your migrated data does not contain any top-level catalog groups. You must ensure that your data contains at least one top-level catalog group for each master catalog. Add the top-level catalog group relationship by adding an entry to the `CATTOGRP` table.

If the consistency checker generates the following return codes, it is recommended but not mandatory that you take the *optional* actions that are listed before proceeding with the migration process for your system. Otherwise, you will not be able to use the WebSphere Commerce 5.5 tools such as the Product Management tool on the migrated data.

Return Code Explanation/Action

- 408** Your migrated data contains catalog groups that are under more than one trading position container. You need to ensure that your catalog groups are not under more than one trading position container. Delete any additional trading position container relationships by removing entries from the `CATGRPTPC` table.
- 501** Your migrated data contains catalog groups with more than one parent catalog group. Ensure that each of your catalog groups only designate one catalog group as its parent. Delete any additional parent catalog groups by removing entries from the `CATGRPREL` table.
- 502** Your migrated data contains catalog entries belonging to more than one catalog group. Ensure that every catalog entry belongs to only one catalog group. Delete any additional catalog groups by removing entries from the `CATGPENREL` table.
- 614** In the `ATTRVALUE` table, there needs to be a row for each attribute value defined with a `CatEntryId = 0`. For example, if you have items that have the attribute — color, and associated attribute values of red and blue, in the `ATTRVALUE` table as follows:

CATENTRY_ID	ATTRVALUE_ID	NAME	STRINGVALUE	LANGUAGE_ID	ATTRIBUTE_ID
item1_id	redval_id	redvalue	red	-1	color_id
item2_id	blueval_id1	bluevalue	blue	-1	color_id
item3_id	blueval_id2	bluevalue	blue	-10	color_id
item4_id	redval_id	redvalue	red	-1	color_id

You need to populate the `ATTRVALUE` table where `CATENTRY_ID` is set to 0 (zero).

CATENTRY_ID	ATTRBUTEVALUE_ID	NAME	STRINGVALUE	LANGUAGE_ID	ATTRIBUTE_ID
item1_id	1001	redvalue	red	-1	color_id
item2_id	1002	bluevalue	blue	-1	color_id
item3_id	1003	bluevalue	blue	-10	color_id
item4_id	1004	redvalue	red	-1	color_id
0	1005	redvalue	red	-1	color_id
0	1006	bluevalue	blue	-1	color_id
0	1007	bluevalue	blue	-10	color_id

Note that the above tables are not complete tables, and are only for the sake of the example. With this method, even if there are not any items defined, you can display valid attribute values that are defined for attributes. If multiple languages exist then each row needs to be defined again for each language.

This step is required only if you want to use the WebSphere Commerce 5.5 tools such as the Product Management tool to manage your catalogs.

Add one row to the ATTRVALUE table with CATENTRY_ID set to 0 (zero) for each case where more than one item has the same ATTRIBUTE_ID.

The catalogs for the Commerce Suite 5.1 sample stores, InFashion or WebFashion, do not have well-formed catalog trees that are suitable for a WebSphere Commerce 5.5 master catalog. You need to redesign the catalog according to the requirements of a master catalog as described in “Important structural considerations” on page 9, otherwise, the WebSphere Commerce 5.5 catalog editor tool, Product Management, may run into problems when navigating the catalog. For examples of well-formed catalog trees, see the sample stores supplied with WebSphere Commerce 5.5 in the *WC55_installdir/samples* directory.

Appendix F. Converting to ATP inventory at a later time

If you choose the `atp` option in the database migration script when you migrate to WebSphere Commerce 5.5, your inventory data is converted by default to the ATP inventory representation supported by WebSphere Commerce 5.5 (and introduced in WebSphere Commerce 5.4). If you choose not to convert your inventory data by specifying the `noatp` option of the migration framework, you can convert to ATP at a later time by running the `migrateATP` script as described in the sections below.

Regardless of whether you choose to migrate using the ATP option or not, the database migration script sets up the following tables for you, for the `PRODUCT` and `ITEM` entries:

- `BASEITEM`
- `ITEMSPC`
- `ITEMVERSN`
- `STOREITEM`
- `VERSIONSPC`

Note that this setup is only done once at migration time. If you add new `PRODUCT` and `ITEM` entries after running the database migration script (`migratedb`) you need to ensure that before you run the `migrateATP` script, the above tables are properly set up for any new `PRODUCT` and `ITEM` entries that have been added since you have run the migration script. If you use the WebSphere Commerce 5.5 tools to add new products and items, the entries will be set up properly for you.

If you use the mass loader (`massload.xml`) to populate the `ITEM` and `PRODUCT` entries, you will also need to set up those tables (`BASEITEM`, `ITEMSPC`, `ITEMVERSN`, `STOREITEM`, and `VERSIONSPC`) prior to running the script. For information on how to use the mass loader, refer to the WebSphere Commerce 5.5 online help information on using the Loader, as well as the descriptions of these tables.

The `migrateATP` script creates entries in the following tables:

- For each product, it creates `DISTARRANG`. It sets the `ENDDATE` to a large value (59 years) and the `STARTDATE` to the current date.
- For each item, it creates `RADETAIL`, `RECEIPT`, `RCTAVAIL`, and `ITEMFFMCTR`.
- For each package, it creates `DISTARRANG`, `RADETAIL`, `RECEIPT`, `RCTAVAIL` and `ITEMFFMCTR`. (Treats it as if each parent is a parent product and each item is a child item.)

It also:

- Fills in the `TRADING_ID` field of all order items to use the store's default contract.
- Generates a script to convert all orders and order items whose status is `C` to a status of `S`.

For DB2 databases

If you are using a DB2 database, use the following procedure to run the `migrateATP` script:

- ___ 1. Switch to the `bin` subdirectory where WebSphere Commerce 5.5 is installed.
For example, `WC55_installdir/bin`.
- ___ 2. Run the `migrateATP` migration script for DB2 as follows:

```
./migrateATP.db2.sh db_name db_userID password  
host_name
```

where:

- *db_name* is the Commerce Suite 5.1 database that you are migrating (for example, mall).
- *db_userID* is the instance user profile for the Commerce Suite 5.1 database that you are migrating (for example, mydbuser).
- *password* is the password for the user ID to connect to the Commerce Suite 5.1 database that you are migrating (for example, mypasswd).
- *host_name* is the fully qualified host name of your machine (for example, myhost.montreal.ca).

For example, to convert your inventory data to the new ATP representation, you can use the following command:

```
migrateATP.db2.sh mall mydbuser mypasswd myhost.montreal.ca
```

3. Check the summary section in the ATPmigrate.log which is generated in the *WC55_userdir/instances/instance_name/migration/logs* directory to see if any errors or warnings have occurred. If no errors have occurred, you can proceed, otherwise you need to address any error condition and rerun the script before proceeding.

Appendix G. Updates to sample JSPs

In order for aspects of your store to work correctly in WebSphere Commerce 5.5 after migration some of the JSPs need to be modified. For example, to be able to complete shopping flows using Store Services, you need to update `shipaddress.jsp` that was shipped in Commerce Suite 5.1. In addition, to improve logon error messaging, you need to update the `register.jsp` and `account.jsp` JSPs that were shipped in Commerce Suite 5.1.

The updated JSPs are listed here for your reference.

register.jsp

```
<%
//*****
//*_-----
//*_ Licensed Materials - Property of IBM
//*_
//*_ 5697-D24
//*_
//*_ (c) Copyright IBM Corp. 2000, 2002
//*_
//*_ US Government Users Restricted Rights - Use, duplication or
//*_ disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
//*_
//*_-----
//*_
%>

<%@ page language="java" %>
<% // All JSPs requires the first 4 packages for getResource.jsp which is used for multi language support %>
<%@ page import="java.io.*" %>
<%@ page import="java.util.*" %>
<%@ page import="com.ibm.commerce.server.*" %>
<%@ page import="com.ibm.commerce.command.*" %>

<%@ page import="javax.servlet.*" %>
<%@ page import="com.ibm.commerce.catalog.beans.*" %>
<%@ page import="com.ibm.commerce.catalog.objects.*" %>
<%@ page import="com.ibm.commerce.beans.*" %>

<%@ page import="com.ibm.commerce.user.beans.*" %>
<%@ page import="com.ibm.commerce.user.objects.*" %>
<%@ page import="com.ibm.commerce.datatype.*" %>
<%@ page import="com.ibm.commerce.usermanagement.commands.ECUserConstants" %>
<%@ page import="com.ibm.commerce.common.beans.*" %>
<%@ page import="com.ibm.commerce.security.commands.ECSecurityConstants" %>
<%@ include file="getResource.jsp"%>
<%
// JSPHelper provides you with a easy way to retrieve
//      URL parameters when they are encrypted
JSPHelper jhelper = new JSPHelper(request);

String storeId = jhelper.getParameter("storeId");
String catalogId = jhelper.getParameter("catalogId");
String languageId = jhelper.getParameter("langId");
%>

<jsp:useBean id="bnError" class="com.ibm.commerce.beans.ErrorDataBean" scope="page">
<% com.ibm.commerce.beans.DataBeanManager.activate(bnError, request); %>
</jsp:useBean>

<%
String strErrorMessage = null;
String strErrorCode = "";

String strLogonID = null;
String strPassword = null;
String strPasswordVerify = null;
String strLastName = null;
String strFirstName = null;
```



```

TypedProperty hshErrorProperties = bnError.getExceptionData();

if (hshErrorProperties != null)
{
    //We have a registration error.

    strErrorCode = hshErrorProperties.getString(ECConstants.EC_ERROR_CODE, "");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONID))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE40");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_LOGONID_EXISTS))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE41");

    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONPASSWORD))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE42");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_BAD_LOGONPASSWORDVERIFY))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE43");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_PASSWORDS_NOT_SAME))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE44");

    if (strErrorCode.equals(ECUserConstants.EC_ADDR_ERR_BAD_LASTNAME))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE45");
    if (strErrorCode.equals(ECUserConstants.EC_UREG_ERR_MISSING_LOGONPASSWORDVERIFY))
        strErrorMessage = infashiontext.getString("ERROR_MESSAGE46");

    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMLLENGTH_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE21");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMDIGITS_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE22");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MINIMUMLETTERS_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE23");
    if (strErrorCode.equals(ECSecurityConstants.ERR_USERIDMATCH_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE24");
    if (strErrorCode.equals(ECSecurityConstants.ERR_REUSEOLD_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE25");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MAXCONSECUTIVECHAR_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE26");
    if (strErrorCode.equals(ECSecurityConstants.ERR_MAXINTANCECHAR_PASSWORD))
        strErrorMessage = infashiontext.getString("PASS_ERROR_MESSAGE27");
    //Redisplay what was entered when the
    //invalid entry was submitted.
    strLogonID = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_UREG_LOGONID));
    strPassword = jhelper.getParameter(ECUserConstants.EC_UREG_LOGONPASSWORD);
    strPasswordVerify = jhelper.getParameter(ECUserConstants.EC_UREG_LOGONPASSWORDVERIFY);
    strLastName = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_ADDR_LASTNAME));
    strFirstName = jhelper.htmlTextEncoder(jhelper.getParameter(ECUserConstants.EC_ADDR_FIRSTNAME));
}
else
{
    //Form is loading under regular condition.
    //Initialize all fields to empty.

    strLogonID = "";
    strPassword = "";
    strPasswordVerify = "";
    strLastName = "";
    strFirstName = "";
}
%>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html>
<head>
<title><%=infashiontext.getString("REGISTER_TITLE")%></title>

<link rel=stylesheet href="<%=storeDir%>/fashionfair.css" type="text/css">
</head>

<body marginheight="0" marginwidth="0">

<!-- Set the user id and e-mail to the same value -->
<SCRIPT language="javascript">
function prepareSubmit(form)
{
    form.<%=ECUserConstants.EC_ADDR_EMAIL1%>.value =
        form.<%= ECUserConstants.EC_UREG_LOGONID%>.value.toLowerCase()
    form.<%=ECUserConstants.EC_UREG_LOGONID%>.value =
        form.<%= ECUserConstants.EC_UREG_LOGONID%>.value.toLowerCase()

    form.submit()
}
</SCRIPT>

<%
String infile;

infile = "/" + storeDir + "/header.jsp";
%>
<jsp:include page="<%=infile%>" flush="true" />

<%
infile = "/" + storeDir + "/sidebar.jsp";
%>

<jsp:include page="<%=infile%>" flush="true" />

<!--END SEARCH-->

<td bgcolor="#FFFFFF" width="600" rowspan="6" valign="top">

<!--MAIN CONTENT STARTS HERE-->

<table cellpadding="2" cellspacing="0" width="580" border="0" align="left">
<tr>

<td width="10" rowspan="10">&nbsp;</td>

<td align="left" valign="top" colspan="3" class="categoryspace">
<font class="category"><%=infashiontext.getString("REGISTRATION")%></font>
<hr width="580" noshade align="left">
<font class="required">*</font><font class="text"><%=infashiontext.getString("REQUIRED_FIELDS3")%></font></td>
</tr><tr>

```

```

<td align="left" valign="top" width="400" class="topspace">
<%
if (strErrorMessage != null)
{
    //We have error message.
%>
<p><font color="red"><%=strErrorMessage%></font><br><br></p>
<%
}
%>
<FORM name="Register" method=POST action="<%=UserRegistrationAdd"%>">
<INPUT TYPE="hidden" NAME="langId" Value="<%=languageId%>">
<INPUT TYPE="hidden" NAME="new" Value="Y">
<INPUT TYPE="hidden" NAME="storeId" Value="<%=storeId%>">
<INPUT TYPE="hidden" NAME="catalogId" Value="<%=catalogId%>">
<INPUT TYPE="hidden" NAME="URL" Value="LogonForm">
<INPUT TYPE="hidden" NAME="page" Value="account">
<INPUT TYPE="hidden" NAME="registerType" Value="G">
<INPUT TYPE="hidden" NAME="profileType" Value="C">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_EMAIL1%>" Value="">
<!--
Lots of mandetary fields are not displayed in this form.
We set them to "-".
-->
<INPUT TYPE="hidden" NAME="personTitle" Value="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ADDRESS1%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ADDRESS2%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_CITY%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_STATE%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_ZIPCODE%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_COUNTRY%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%=ECUserConstants.EC_ADDR_PHONE1%>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%= ECUserConstants.EC_UREG_CHALLENGEQUESTION %>" VALUE="-">
<INPUT TYPE="hidden" NAME="<%= ECUserConstants.EC_UREG_CHALLENGEANSWER %>" VALUE="-">

<table cellpadding="3" cellspacing="0" border="0" align="left">
<tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("EMAIL2")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_UREG_LOGONID%>" value="<%=strLogonID%>"></td>
</tr><tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("PASSWORD3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_UREG_LOGONPASSWORD%>"
type="password" value="<%=strPassword%>"></td>
</tr><tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("VERIFY_PASSWORD3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_UREG_LOGONPASSWORDVERIFY%>"
type="password" value="<%=strPasswordVerify%>"></td>
</tr>
<% if (locale.toString().equals("ja_JP") || locale.toString().equals("ko_KR") ||
locale.toString().equals("zh_CN") || locale.toString().equals("zh_TW")) { %>

```

```

<tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("LAST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_ADDR_LASTNAME%>" value="<%=strLastName%>" type="text"></td>
</tr><tr>
<td align="right" valign="middle">
<font class="strongtext"><%=infashiontext.getString("FIRST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_ADDR_FIRSTNAME %>" value="<%=strFirstName%>" type="text"></td>
</tr>
<% } else { %>
<font class="product">
<tr>
<td align="right" valign="middle">
<font class="strongtext"><%=infashiontext.getString("FIRST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%= ECUserConstants.EC_ADDR_FIRSTNAME %>" value="<%=strFirstName%>" type="text"></td>
</tr><tr>
<td align="right" valign="middle">
<font class="required">*</font><font class="strongtext"><%=infashiontext.getString("LAST_NAME3")%></font></td>
<td align="left" valign="middle">
<input size="25" maxlength="50" name="<%=ECUserConstants.EC_ADDR_LASTNAME%>" value="<%=strLastName%>" type="text"></td>
</tr>
<% } %>
<tr>
<td align="left" valign="top">
<font class="text">&nbsp;</font></td>
<td align="left" valign="top" class="categoryspace">
<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="javascript:prepareSubmit(document.Register)"><font class="strongtext"><%=infashiontext.getString("SUBMIT")%></font></a>
</td>
</tr></table>
</td></tr></table>
</form>
</td>

<td width="180" valign="top" class="topspace">
<table cellpadding="3" cellspacing="0" border="1" width="180" bgcolor="#FFFFCC">
<tr>
<td align="left" valign="top">
<font class="strongtext"><%=infashiontext.getString("YOUR_PRIVACY")%></font><br>
<font class="text"><%=infashiontext.getString("PRIVACY_STATEMENT")%><br>
<a href="PrivacyView?langId=<%=languageId%>&storeId=<%=storeId%>&catalogId=<%=catalogId%>">
<%=infashiontext.getString("LEARN_MORE")%></a></font>
</td></tr></table>
</td>

</tr></table>
</td>

</tr></table>
<%
incfile = "/" + storeDir + "/footer.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

</body>

</html>

```

account.jsp

```
<%
//*****
//*_____
//* Licensed Materials - Property of IBM
//*
//* 5697-D24
//*
//* (c) Copyright IBM Corp. 2000, 2002
//*
//* US Government Users Restricted Rights - Use, duplication or
//* disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
//*_____
//*
%>

<%@ page language="java" %>
<% // All JSPs requires the first 4 packages for getResource.jsp which is used for multi language support %>
<%@ page import="java.io.*" %>
<%@ page import="java.util.*" %>
<%@ page import="com.ibm.commerce.server.*" %>
<%@ page import="com.ibm.commerce.command.*" %>

<%@ page import="javax.servlet.*" %>
<%@ page import="com.ibm.commerce.catalog.beans.*" %>
<%@ page import="com.ibm.commerce.catalog.objects.*" %>
<%@ page import="com.ibm.commerce.beans.*" %>
<%@ page import="com.ibm.commerce.user.beans.*" %>
<%@ page import="com.ibm.commerce.user.objects.*" %>
<%@ page import="com.ibm.commerce.datatype.*" %>
<%@ page import="com.ibm.commerce.usermanagement.commands.ECUserConstants" %>
<%@ page import="com.ibm.commerce.security.commands.ECSecurityConstants" %>
<%@ page import="com.ibm.commerce.common.beans.*" %>
<%@ include file="getResource.jsp"%>

<%
CommandContext commandContext = (CommandContext)
    request.getAttribute(ECConstants.EC_COMMANDCONTEXT);

String catalogId = request.getParameter("catalogId");
String storeId = request.getParameter("storeId");
String languageId = request.getParameter("langId");

//Parameters may be encrypted.
if (catalogId == null)
    catalogId = ((String[]) request.getAttribute("catalogId"))[0];
if (storeId == null)
    storeId = ((String[]) request.getAttribute("storeId"))[0];
if (languageId == null)
    languageId = ((String[]) request.getAttribute("langId"))[0];
%>
```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">

<head><title><%=infashiontext.getString("ACCOUNT_TITLE")%></title>
<link rel=stylesheet href="<%=storeDir%>/fashionfair.css" type="text/css">
</head>

<body marginheight="0" marginwidth="0">

<%
String incfile;

incfile = "/" + storeDir + "/header.jsp";
%>
<jsp:include page="<%=incfile%>" flush="true"/>

<%
incfile = "/" + storeDir + "/sidebar.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

<!--END SEARCH-->

<td bgcolor="#FFFFFF" width="600" rowspan="6" valign="top">

<%
//Deal with possible errors when logging in
String strPageTitle    = "Logon";
String strErrorMessage = null;
String strErrorCode    = null;

String[] strArrayAuth = (String [])request.getAttribute(ECConstants.EC_ERROR_CODE);

if (strArrayAuth != null){
    if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_DISABLED_ACCOUNT) == true){
        strErrorMessage = infashiontext.getString("ACCOUNT_LOCKED");
    }else if( strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_MISSING_LOGONID) == true){
        strErrorMessage = infashiontext.getString("LOGIN_ID_MISSING");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_INVALID_LOGONID) == true){
        strErrorMessage = infashiontext.getString("LOGON_ID_INVALID");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_MISSING_PASSWORD) == true){
        strErrorMessage = infashiontext.getString("PASSWD_MISSING");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_INVALID_PASSWORD) == true){
        strErrorMessage = infashiontext.getString("PASSWORD_INCORRECT");
    }else if(strArrayAuth[0].equalsIgnoreCase(ECSecurityConstants.ERR_LOGON_NOT_ALLOWED) == true){
        strErrorMessage = infashiontext.getString("WAIT_TO_LOGIN");
    }
}
}

```

```

/*
 * Register link behaves differently depending on if the user is logged
 * in. If the user is logged in (hence a registered user) clicking on
 * register will log the user off then display the registration form.
 * This will allow multiple registration using the same browser.
 * For guest shoppers clicking on Register link will simply display the
 * registration form.
 */
UserRegistrationDataBean regBean = new UserRegistrationDataBean();
com.ibm.commerce.beans.DataBeanManager.activate(regBean, request);
String regURL = null;

if (! regBean.findUser())
// findUser() return false if the customer is not registered
{
    //This is a guest user. Simply display the registration form.
    regURL = "UserRegistrationForm";
}
else
{
    //This is a registered/logged in user. Log him off first.
    //Through LogoffView registration page will be displayed.
    regURL = "Logoff";
}
%>
<!--MAIN CONTENT STARTS HERE-->

<table cellpadding="2" cellspacing="0" width="580" border="0">
<tr>

<td width="10" rowspan="10">&nbsp;</td>

<td align="left" valign="top" colspan="3" class="categoryspace">
<font class="category"><%=infashiontext.getString("MY_ACCOUNT3")%></font>
<hr width="580" color="#336666" noshade align="left">
</td>

</tr><tr>

<td align="left" valign="top" width="280" bgcolor="#CC6600">
<font class="subheader"><%=infashiontext.getString("PERSONAL_INFO")%></font></td>

<td width="20" rowspan="5">&nbsp;&nbsp;&nbsp;</td>

<td align="left" valign="top" width="280" bgcolor="#CC6600">
<font class="subheader"><%=infashiontext.getString("ADDRESS_BOOK")%></font>
</td>

</tr><tr>

<td align="left" valign="top" width="280" class="topspace">
<font class="text"><%=infashiontext.getString("UPDATE_NAME")%></font><p>

<table cellpadding="0" cellspacing="0" border="0" align="left">
<tr>
<td align="left" valign="top">

```

```

<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="UserRegistrationForm?storeId=<%=storeId%>&langId=<%=languageId%>&catalogId=<%=catalogId%>">
<font class="strongtext"><%=infashiontext.getString("CHANGE_INFO")%></font></a></td>
</tr></table>

</td>
</tr></table>

</td>

<td align="left" valign="top" width="280" class="topspace">

<%
if (strErrorMessage != null)
{
    //We have an error message.
    %>
<p><font color="red" ><%=strErrorMessage%></font><br>
<%
}
%>
<table cellpadding="0" cellspacing="0" border="0">
<tr>
<td align="left" valign="top">
<font class="text"><%=infashiontext.getString("UPDATE_ADDRESS1")%></font><br><br></td>
</tr><tr>

<td align="left">
<table cellpadding="4" cellspacing="0" border="0">
<tr>
<td align="left" valign="middle" bgcolor="#FFCC99">
<A href="AddressBookForm?storeId=<%=storeId%>&langId=<%=languageId%>&catalogId=<%=catalogId%>">
<font class="strongtext"><%=infashiontext.getString("EDIT_ADD")%></font></a></td>
</tr></table>
</td></tr></table>

</tr></table>
</td>

</td>
</tr></table>
<%
incfile = "/" + storeDir + "/footer.jsp";
%>

<jsp:include page="<%=incfile%>" flush="true"/>

</body>

</html>

```

infashiontext_en_US.properties

LOGON_ID_INVALID = The e-mail address entered is invalid. Type another e-mail address in the E-mail address field and try again.
PASSWORD_INCORRECT = The password entered is incorrect. Type your password in the Password field and try again.
LOGIN_ID_MISSING = Type an e-mail address in the E-mail address field.
PASSWORD_MISSING = Type a password in the Password field.
ACCOUNT_LOCKED = Due to 3 unsuccessful password attempts, you will be unable to logon. Please contact a store representative to unlock your account.
WAIT_TO_LOGIN = Please wait a few seconds before attempting to log in again.

PASS_ERROR_MESSAGE21 = You entered a password with less than 6 characters. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE22 = Your password does not contain a digit. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE23 = Your password does not contain a letter. Passwords must be at least 6 characters in length, and include one digit and one letter. Please re-enter your password.
PASS_ERROR_MESSAGE24 = Your password is the same as your user-id. Please assure that your user-id and password are different.
PASS_ERROR_MESSAGE25 = Your new password is the same as the previous one. Please enter a new password, or choose 'My Account' on the menu bar to return to your account page.
PASS_ERROR_MESSAGE26 = A character in your password occurs more consecutively than the allowed limit of 3. Please re-enter your password.
PASS_ERROR_MESSAGE27 = A character in your password occurs more than the allowed limit of 4. Please re-enter your password.

infashion store translation text

ENCODINGSTATEMENT = text/html; charset=ISO_8859-1

header.jsp

SHOPPING_CART = SHOPPING CART
MY_ACCOUNT = MY ACCOUNT
CONTACT_US = CONTACT US
HELP = HELP
SEARCH = SEARCH
HOME = Home

#footer.jsp

SHOPPING_CART2 = Shopping cart
MY_ACCOUNT2 = My account
CONTACT_US2 = Contact us
HELP2 = Help
PRIVACY_POLICY = Privacy policy

#sidebar.jsp

CHOOSE_COUNTRY = CHOOSE A LANGUAGE
CANADA = Canada
UNITED_STATES = United States
CHINA = China
FRANCE = France
GERMANY = Germany
ITALY = Italy
JAPAN = Japan
SPAIN = Spain
TURKEY = Turkey
SERVICES = SERVICES
REGISTER = Register
NOW_BUY = now and get 10% off your first purchase!
NEED_HELP = Need help?
JUST_ASK = Just ask!
GO = GO

.
.
.

Appendix H. Troubleshooting

This section lists potential problems you may encounter during migration, and actions to resolve these problems.

1. **Problem:** You have store that uses the CatalogDataBean access bean, or a store that is based on the WebFashion sample store, and you can no longer display catalog groups such as the product in the home page.

Solution: In Commerce Suite 5.1, CatalogDataBean displays catalog groups that do not belong to the current store. That is, it displays catalog groups that should not be displayed. In WebSphere Commerce 5.5, this behavior has been modified so as not to display the catalog groups that do not belong to the current store.

If you want to display the catalog groups in the home page after migration you can perform the following steps:

- a. Execute the following SQL:
`select * from cattogrp where CATALOG_ID= your_store_id`
 - b. From the result , you can find the value of catgroup_id that catalog missed in the home page, for example, 11111.
 - c. Execute the following SQL:
`insert into storecgrp (catgroup_id,storeent_id) values (11111,your_store_id)`
 - d. Launch your store. You should now be able to find the product in the home page.
2. **Problem:** When migrating campaign data, you receive an error in the database migration log similar to the following:

```
*** Migrating INITIATIVE ***
Query: select * from keys where tablename='initiative'

...
Query: select * from campaign where name='End of Season Sale.' and storeent_id=10051
Query: select * from bzrpent where path=
    /UserTemplate/Store10051/Campaign/Campaign10101/Initiative/Initiative10001_working'
Query: select * from bzrpentstg where bzrpent_id=10009 order by sequence
Query: select * from intvmpe where initiative_id=10001
Query: select * from mpe where mpe_id=10051
Query: select * from emspot where name='StoreHomePage' and storeent_id = 10051
MigrateCampaigns [Fatal Error] :240:2019: Attribute name "instan" must be followed by the
' = ' character.
Attribute name "instan" must be followed by the ' = ' character.
Campaign 10101: failed to migrate initiative 10001 ...
    possible error in the rule xml parsing, please check the log for details.
Execute update keys set counter=10050 where tablename='initiative'
Execute update keys set counter=10000 where tablename='intvsched'
```

```
*** INITIATIVE migrated and committed ***
```

Solution: Analyze the database migration log and campaign data in the database. For example, if you find that there is corrupt data in the BZRPENTSTG table, you can issue a select similar to the following:

```
select * from bzrpentstg where bzrpent_id=10009
```

If this select yields results similar to:

```
.....
10006 <value>com.ibm.commerce.marketingcenter.object.rule.SelectAttributeWithValue</value>
      .....
      </value>
    </item>
  </instance-list>
  <instan //this element key word has been cut off,
//so migration for campaign initiative has failed.
```

```

// The following is next record
10007 ce ref='Class'>
  <value>com.ibm.commerce.marketingcenter.object.rule.
    CatalogEntryDistinctiveRecommendationActionEndMark</value>
  </instance>

```

.....

you would need to manually modify the corrupt record in the table BZRPESTSTG in the Commerce Suite 5.1 database to ensure that the <instance ref='Class'> string is in the same record and rerun the database migration.

3. **Problem:** You receive a "Response already committed" error, for a catalogReq.jsp page similar to the following:

```

452957dc WebGroup I SRVE0180I: [Stores] [/webapp/wcs/stores] [Servlet.LOG]: /xxxxx/catalogReq.jsp: init
ERROR 2004-06-01 15:34:31,528 - Unrecoverable error: JSP Exception occurred on the catalogReq.jsp page.
Thread ID: <Servlet.Engine.Transports : 0_452957dc#14>
Client ID: <X%2bh99Sz3R7L0jKCiXLIqT6J6DWA%3d%0d%0a>

```

```

com.ibm.ws.webcontainer.servlet.exception.UncaughtServletException: Server caught unhandled exception from servlet
[SimpleFileServlet]: Response already committed.
at com.ibm.ws.webcontainer.webapp.WebAppRequestDispatcher.handleWebAppDispatch(WebAppRequestDispatcher.java:994)
at com.ibm.ws.webcontainer.webapp.WebAppRequestDispatcher.dispatch(WebAppRequestDispatcher.java:530)

```

Solution: WebSphere Application Server 5.0 looks for the file to be included based on the Application Server root and not the Web Server root. For instance, assume that you have set up a Web server project and pointed its "extendedDocumentRoot" to "C:\htdocs". And under C:\htdocs, you have your \stores\catalog\YY.html file. With this setting you are able to view this file in your browser using the following Web address: http://myhostname/stores/catalog/YY.html. If your JSP file contains the following include page statement:

```
<jsp:include page="/stores/catalogs/YY.html" />
```

it looks for the YY.html file under ../workspace/Stores/Web Content/stores/catalogs. As it cannot find it there, it redirects to the 404 page, and the server throws an exception. If you add "C:\htdocs" to the extendedDocumentRoot attribute of the Stores project, it resolves properly.

In this case, the above exception is due to the presence of an HTML file inside the JSP file, instead of another JSP file. The solution is to change the HTML file to a JSP file.

Enabling logging and tracing

In WebSphere Commerce 5.5, the logging subsystem makes use of the WebSphere Application Server logging infrastructure. This simplifies administration by making the logs consistent throughout the system and takes advantage of the tooling provided with WebSphere Application Server to improve problem determination capabilities. As log files are shared between the WebSphere Commerce application infrastructure the WebSphere Application Server correlation of log records between the two is automatic.

If you have created tools which use the Commerce Suite 5.1 logs directly, you will need to modify the tools use the WebSphere Application Server logging infrastructure. For the WebSphere Commerce standalone applications (such as Configuration Manager and the dbclean utility), the logs and traces use the previous logging format, rather than the new WebSphere Application Server JRas format.

For more information, see the sections on diagnosing and fixing problems using the available logging infrastructure, in the WebSphere Application Server Version

Logging

The following sections describe logging:

- Logging severities

In Commerce Suite 5.1, there are five severity types: ERROR, INFORMATION, DEBUG, STATUS, WARNING. However, in the WebSphere Application Server logging facility, there are three types of logging - ERROR, INFORMATION and WARNING. The following table shows the mapping to the WebSphere Application Server log types.

Table 10. Mapping of log severity types

Severity types in Commerce Suite 5.1 logging	Severity types in WebSphere Commerce 5.5 logging (using WebSphere Application Server JRes extensions)
ERROR or ERR	TYPE_ERROR or TYPE_ERR
INFORMATION or INFO	TYPE_INFORMATION or TYPE_INFO
DEBUG	TYPE_INFORMATION or TYPE_INFO
WARNING or WARN	TYPE_WARNING or TYPE_WARN
STATUS	TYPE_INFORMATION or TYPE_INFO

Note that the ERROR, INFORMATION, and WARNING types do not change and are mapped to the corresponding types in WebSphere Application Server JRes extensions. The STATUS and DEBUG are logged as INFORMATION type.

- Output file location

By default, the output log file is called `activity.log` which is located in the following directory: `/QIBM/UserData/WebAS5/Base/instance_name/logs`. Since the file `activity.log` is in binary format, you need to use the WebSphere Application Server Log Analyzer to read this file. Instructions for using Log Analyzer:

If any of the trace components are enabled, WebSphere Application Server JRes will write the log entries in plain text format into the trace output file along with the trace entries.

- Configuring logging

You can configure logging, for example, enabling and disabling the logging severities and changing the file name. To configure logging from the WebSphere Application Server Administration Console:

1. Click **Troubleshooting** —> **Logs and Trace** —> **WC_instance_name** —> **IBM Service Logs**
2. Click on **IBM Service Logs**. In the Configuration panel, click **Message Filtering** to enable the logging severities. You can change the default logging output file in the File Name field before the server is started.

Note: To dynamically configure the `WC_instance_name` server from the WebSphere Application Server Administration Console, you need to have WebSphere Application Server Network Deployment installed. Otherwise, you need to restart the server `WC_instance_name` server after configuring it.

Using Log Analyzer

Log Analyzer is installed by default when you install WebSphere Application Server 5.0. To launch and use Log Analyzer:

- Click **Start** → **Programs** → **IBM WebSphere** → **Application Server v5.0** → **Log Analyzer**.
- Use Log analyzer for trouble shooting runtime logs:
 1. Open the log file you wish to analyze:

From the Log Analyzer window, select **File** → **Open** to open the log file to be analyzed. The WebSphere Commerce and WebSphere Application Server logs are located in `/QIBM/UserData/WebAS5/Base/instance_name/logs`
 2. (Optional) To select the sequence of the messages in left panel:

From the Log Analyzer window, select **File** → **Preferences**, and highlight **Logs** from the left panel, and in the Sorting Sequence table in the right panel, select the sequence you would like the messages to be displayed in the left panel.
 3. To analyze a message:
 - a. Expand time stamps to display messages to be analyzed
 - b. Highlight and right-click a message and choose Analyze. The result will be shown in the tables in the right panel.

Tracing

The following sections describe tracing:

- Configuring tracing

Trace entries are stored in an in-memory circular buffer called a ring buffer. A ring buffer must be dumped to a file in order for the trace to be viewed. In the WebSphere Application Server Administrative Console GUI, you can set the size of the ring buffer, dump the ring buffer and direct the trace entries to a file by specifying the output file name. By default, the trace output file, `trace.log`, is located at: `/QIBM/UserData/WebAS5/Base/instance_name/logs`

To configure tracing from the WebSphere Application Server Administration Console:

1. Click **Troubleshooting** → **Logs and Trace** → `WC_instance_name` → **Diagnostic Trace**
2. Click on **IBM Service Logs**. Use the Configuration panel before the server is started. You can enter the trace specification directly. For example,

```
com.ibm.websphere.commerce.WC_SERVER=all=  
enabled:com.ibm.websphere.commerce.WC_RAS=all=enabled
```

(Note that the above example is split across two lines for display purposes; it should actually be on one line.)

Also you can specify the trace output format and output file name.

3. If WebSphere Application Server Network Deployment is not installed, please manually enter the trace string as shown in the above step. If WebSphere Application Server Network Deployment is installed, you can use the WebSphere Application Server GUI to configure the tracing in the Configuration and Runtime tab as following:
 - a. Click the **Modify** button to enable the tracing components using the graphical trace interface.
 - b. Expand until you see `com.ibm.websphere.commerce`
 - c. Right-click on the tracing component you would like to enable and select **All**.

Note that the tracing components can be dynamically enabled by clicking the **Apply** button without restarting the server.

- Trace components

Table 11. Mapping of trace components

Commerce Suite 5.1 tracing components	WebSphere Application Server JRes extensions trace logger
SERVER	com.ibm.websphere.commerce.WC_SERVER
CATALOG	com.ibm.websphere.commerce.WC_CATALOG
DATASOURCE	com.ibm.websphere.commerce.WC_DATASOURCE
ORDER	com.ibm.websphere.commerce.WC_ORDER
USER	com.ibm.websphere.commerce.WC_USER
COMMAND	com.ibm.websphere.commerce.WC_COMMAND
CF	com.ibm.websphere.commerce.WC_CF
NEGOTIATION	com.ibm.websphere.commerce.WC_NEGOTIATION
RAS	com.ibm.websphere.commerce.WC_RAS
DB	com.ibm.websphere.commerce.WC_DB
METAPHOR	com.ibm.websphere.commerce.WC_METAPHOR
SCHEDULER	com.ibm.websphere.commerce.WC_SCHEDULER
DEVTOOLS	com.ibm.websphere.commerce.WC_DEVTOOLS
TOOLSFRAMEWORK	com.ibm.websphere.commerce.WC_TOOLSFRAMEWORK
RULESYSTEM	com.ibm.websphere.commerce.WC_RULESYSTEM
MERCHANDISING	com.ibm.websphere.commerce.WC_MERCHANDISING
MARKETING	com.ibm.websphere.commerce.WC_MARKETING
REPORTING	com.ibm.websphere.commerce.WC_REPORTING
TRANSPORT_ADAPTER	com.ibm.websphere.commerce.WC_TRANSPORT_ADAPTER
SVT	com.ibm.websphere.commerce.WC_SVT
PERFMONITOR	com.ibm.websphere.commerce.WC_PERFMONITOR
MESSAGING	com.ibm.websphere.commerce.WC_MESSAGING
STOREOPERATIONS	com.ibm.websphere.commerce.WC_STOREOPERATIONS
CACHE	com.ibm.websphere.commerce.WC_CACHE
EVENT	com.ibm.websphere.commerce.WC_EVENT
EJB	com.ibm.websphere.commerce.WC_EJB
CURRENCY	com.ibm.websphere.commerce.WC_CURRENCY
CATALOGTOOL	com.ibm.websphere.commerce.WC_CATALOGTOOL
CONTRACT	com.ibm.websphere.commerce.WC_CONTRACT
UBF	com.ibm.websphere.commerce.WC_UBF
BI	com.ibm.websphere.commerce.WC_BI
INVENTORY	com.ibm.websphere.commerce.WC_INVENTORY
UTF	com.ibm.websphere.commerce.WC_UTF
RFQ	om.ibm.websphere.commerce.WC_BI
EXCHANGE	com.ibm.websphere.commerce.WC_INVENTORY
ACCESSCONTROL	com.ibm.websphere.commerce.WC_ACCESSCONTROL
AC_UNITTEST	com.ibm.websphere.commerce.WC_AC_UNITTEST
APPROVAL	com.ibm.websphere.commerce.WC_APPROVAL
COLLABORATION	com.ibm.websphere.commerce.WC_COLLABORATION
THREAD	com.ibm.websphere.commerce.WC_THREAD

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 user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

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, 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.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

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 which 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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the

names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Credit card images, trademarks and trade names provided in this product should be used only by merchants authorized by the credit card mark's owner to accept payment via that credit card.

Trademarks

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

- AIX
- DB2
- @server
- IBM
- iSeries
- Lotus
- Net.Commerce
- OS/400
- QuickPlace
- Sametime
- SecureWay
- WebSphere

Microsoft, Windows, Windows NT, and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the United States, other countries, or both.

SET and the SET Logo are registered trademarks owned by SET Secure Electronic Transaction LLC in the United States, other countries, or both.

Java, JavaBeans and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc.

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



Printed in USA