

IBM WebSphere Commerce - Express



Migration Guide for OS/400

from WebSphere Commerce 54 Professional Entry Edition

Version 5.5

IBM WebSphere Commerce - Express



Migration Guide for OS/400

from WebSphere Commerce 54 Professional Entry Edition

Version 5.5

Note:

Before using this information and the product it supports, be sure to read the general information under “Notices” on page 115.

First Edition, Fourth Revision (December, 2004).

This edition applies to IBM WebSphere Commerce - Express Version 5.5 for @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	v
Summary of changes	v
Supported migration paths	vi
Migration paths by operating systems	vii
Other considerations	vii
Conventions used in this book	vii
Path variables	viii
Running database scripts	ix
References to Oracle	ix

Chapter 1. Before you migrate from WebSphere Commerce 5.4. **1**

Premigration actions	2
Ensuring no new messages are stored in MSGSTORE table	2
Preparing the staging server for migration	3
Transitioning to WebSphere Application Server 5.0	3

Chapter 2. Backing up WebSphere Commerce 5.4 **5**

Backing up your WebSphere Commerce 5.4 system	5
Backing up directories and files	5
Backing up your databases	6

Chapter 3. Upgrading your software . . . **7**

Mapping the WebSphere Commerce 5.4 to WebSphere Commerce 5.5 IBM software	7
Approach to upgrading the software	7
Installing WebSphere Commerce 5.5 on a separate machine (remote migration)	8
Additional machine requirements for transition	9
Creating a WebSphere Commerce test instance	9
Exporting the WebSphere Commerce 5.4 enterprise application	10
Migrating your previous WebSphere Application Server configuration	10
Next Step	12
Upgrading your existing system to WebSphere Commerce 5.5 (in-place migration)	12
Upgrading your hardware	12
Upgrading your operating system	12
Upgrading additional software components	12
Exporting the WebSphere Commerce 5.4 enterprise application	12
Upgrading to WebSphere Application Server 5.0	13
Upgrading to WebSphere Commerce 5.5	14
Next Step	15

Chapter 4. Minimizing downtime during coexistence migration **17**

Assigning a host name and IP address	17
Creating a new WebSphere Commerce 5.4 instance	17
Dropping the new WebSphere Commerce 5.4 instance database	18

Backing up and restoring the database for the instance to be migrated	18
Exporting the EAR file of the instance to be migrated	18
Removing the EAR of the new WebSphere Commerce 5.4 instance	18
Installing the EAR of the instance to be migrated	19
Testing the new WebSphere Commerce 5.4 instance	21
Proceeding with instance and data migration	21

Chapter 5. Migrating your Commerce instance configuration **23**

Checking the WebSphere Commerce 5.4 product.dtd file	23
Checking the WebSphere Commerce 5.4 product.xml file	23
Updating system values in the WebSphere Commerce 5.5 product.xml file	23
Checking the WebSphere Commerce 5.5 product.xml file (in-place migration only)	24
Steps before migrating your instance configuration	25
Migrating your instance configuration using WCIM	26
Prerequisites for running WCIM	27
Migrating a WebSphere Commerce 5.4 instance	28
After running the WCIM script	38
Adjusting JVM properties in WebSphere Application Server	38
Running the wcimWasConfig.jacl script	39
Checking the JDBC Provider	41
Deploying the migrated .ear file	41
Selecting the WebSphere Application Server pass by reference setting	43
Creating JNDI names for JCA connectors	43
Updating file and directory permissions	44
Transitioning and deploying your custom code Customizations in httpd.conf	44
Reconfiguring Web servers	44
Adjusting the <i>WC_instance_name.ear</i> path	45
Enabling WebSphere Application Server EJB Security	45
Copying files with static content to the documentation root directory	45

Chapter 6. Before migrating your database **47**

Ensuring database journal receivers are large enough	47
Updating database system views to the V5R2 level	48
Considering the order of columns	48
Checking the unique indexes in the MSGTYPES table	49
Running the database preparation script	49
DB2 databases	50
Dropping custom constraints	51

Preconditions for running the database migration script for WebSphere Commerce 5.4 databases . . .	51
Basic data migration (no custom data) . . .	52
Custom data migration . . .	52

Chapter 7. Migrating your WebSphere Commerce 5.4 database 53

Migrating your database	53
DB2 databases	54
Restoring custom constraints	55
DB2 Databases	55
Remigrating your database	55

Chapter 8. Post instance and database migration. 57

iSeries Apache HTTP Server	57
Migrating store assets	57
+ Updating the WebSphere Commerce 5.4 JSP files	58
Tools XML Files Migration	58
Migrating cache policies	59
Enabling WebSphere Application Server dynamic cache	59
Enabling the dynamic cache service and servlet caching	60
Web server plug-in consideration	60
Web address change for Organization Administration Console	61
Restarting the Web server and WebSphere Application Server	61
Changing your host name	61
Verifying successful instance and database migration	62

Chapter 9. Migrating to Commerce Payments 63

Payments migration overview	63
Components migrated	63
Transition methodology	63
Third-party cassette considerations (custom code)	64
Product rename in version 3.1.3	64
Payment Manager 3.1.1	65
Premigration considerations	65
Migrating your Payments instance and database	67
Migrating your Payments instance using WCIM	67
Updating the Web server ports	71
Migrating your Payments database	72
Post instance and database migration steps	73
Changing to WCSRealm (recommended)	73
Continuing to use PSOS400Realm (optional)	74
Starting the Payments instance	74
Payments migration usage scenarios	75
Migrating the previous Payment Manager in-place	75
Migrating the previous Payment Manager on a remote machine	76
Multiple WebSphere Commerce instances pointing to one instance of Payments	77
Payments subsystem migration considerations	78
What the data migration does	78

Chapter 10. Migrating other WebSphere Commerce components 81

Configuring user roles	81
Migrating the staging server	81
Reconfiguring the database cleanup utility	82
Business accounts and contracts	82
Access control policy subscriptions	83
Migrating WebSphere Commerce 5.4 overridden policies in the ACORGPOL table	84
Roles removed in WebSphere Commerce 5.5	84
WebSphere Commerce 5.4 setup (premigration)	84
WebSphere Commerce 5.5 conversion (postmigration)	85

Appendix A. Where to find more information 87

WebSphere Commerce information	87
WebSphere Commerce online help	87
WebSphere Commerce technical library	87
WebSphere Commerce Payments information	87
IBM HTTP Server information	89
WebSphere Application Server information	89
DB2 Universal Database information	89
Other IBM publications	89

Appendix B. Renaming your Payments instance 91

Appendix C. Restoring a Payments library under a different name when the Payments instance has been moved to a new system 93

Appendix D. Extending the data migration script 95

Appendix E. Overview of the WCIM tool and the data migration script 97

Instance migration using WCIM	97
Data migration script	97
Supplemental information for data migration	98
Ensuring column lengths are not exceeded	99
Checking the database migration log files	99

Appendix F. Reducing the size of ACRELGRP.CONDITIONS and ACRESGRP.CONDITIONS 103

Appendix G. Troubleshooting 109

Enabling logging and tracing	110
Logging	110
Tracing	111

Notices 115

Trademarks	117
----------------------	-----

About this book

This document describes the steps to migrate IBM® WebSphere® Commerce 5.4 Professional Entry Edition to IBM WebSphere Commerce 5.5 - Express on IBM @server® iSeries™. If you wish to migrate your previous version of WebSphere Commerce Studio to WebSphere Commerce - Express Developer Edition, see the *WebSphere Commerce - Express Developer Edition Migration Guide*.

This document collectively refers to WebSphere Commerce 5.4 Professional Entry Edition plus any associated fix packs that have been applied to that release, as “WebSphere Commerce 5.4”. This document refers to the current release, WebSphere Commerce 5.5 - Express, 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:

- Express Edition
(<http://www.ibm.com/software/genservers/commerce/express/>)
- 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 - Express 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 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 43
Added in information on creating JNDI names for JCA connectors.	“Creating JNDI names for JCA connectors” on page 43

Change	Chapter, section, or page affected
Added setenv.sh file requirement for WCIM script when migrating Payments.	"Migrating your Payments instance using WCIM" on page 67
Added <i>WC_instance.ear</i> path consideration when changing the WebSphere Application Server instance from default to non-default or from non-default to default.	"Adjusting the <i>WC_instance_name.ear</i> path" on page 45
Corrected steps to check the JDBC provider in the WebSphere Application Server Administration Console.	"Checking the JDBC Provider" on page 41
Added several steps to the procedure on changing your host name to the final host name after migration.	"Changing your host name" on page 61
Removed <i>instance_name_was_Start.xml</i> file from list of files in the WCIM backup zip file.	"Running WCIM for remote migration" on page 32
Added steps to enable the "Enable servlet caching" flag.	"Enabling the dynamic cache service and servlet caching" on page 60
Clarify steps for deploying ear file.	"Deploying the migrated .ear file" on page 41

Supported migration paths

Note:

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

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

- WebSphere Commerce 5.4 Professional Entry Edition to WebSphere Commerce 5.5 - Express Edition

WebSphere Commerce 5.5 - Express supports migration from the following WebSphere Commerce 5.4 fix pack levels:

- WebSphere Commerce 5.4.0.1
- WebSphere Commerce 5.4.0.2
- WebSphere Commerce 5.4.0.3
- WebSphere Commerce 5.4.0.4
- WebSphere Commerce 5.4.0.5

WebSphere Commerce 5.5 also supports migration from the following Commerce Enhancement Packs:

- WebSphere Commerce 5.4 Commerce Enhancement Pack 1
- WebSphere Commerce 5.4 Commerce Enhancement Pack 2
- WebSphere Commerce 5.4 Commerce Enhancement Pack 3

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 WebSphere Commerce 5.4.

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>)

Migration paths by operating systems

WebSphere Commerce 5.5 supports the migration to the Express Edition from WebSphere Commerce 5.4 Professional Entry Edition on the same operating system. It specifically supports migration from the following paths:

- iSeries to iSeries
- Windows[®] 2000 to Windows 2000

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

WebSphere Commerce 5.5 does not support migration between dissimilar operating systems, for example, WebSphere Commerce 5.4 on Windows NT[®] to WebSphere Commerce 5.5 on iSeries.

Other considerations

- When you install a WebSphere Commerce 5.5 system on a machine that has WebSphere Commerce 5.4 installed, the old version will not be upgraded to WebSphere Commerce 5.5 and both versions can coexist.
- If you have customized code or commands written in Java[™] or Enterprise JavaBeans[™] objects in WebSphere Commerce 5.4, 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 - Express Developer Edition Migration Guide*. This transition is required due to the move to WebSphere Application Server 5.0 from WebSphere Application Server 4.0.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:

WAS40_installdir

This variable represents the actual installation directory for WebSphere Application Server Version 4.0.x provided with WebSphere Commerce 5.4: /QIBM/ProdData/WebASAdv4

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*

WC54_installdir

This variable represents the actual installation directory for WebSphere Commerce 5.4: /QIBM/ProdData/WebCommerce

WC54_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/WebCommerce

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

WPM312_path

This variable represents the actual installation directory for the previous version of WebSphere Commerce Payments (WebSphere Payment Manager 3.1.2) provided with WebSphere Commerce 5.4. 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.

Chapter 1. Before you migrate from WebSphere Commerce 5.4

The following sections describe specific steps you must complete while you still have an operational WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4, 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 - Express Developer Edition Migration Guide*. This transition is required due to the upgrade to WebSphere Application Server 5.0 from WebSphere Application Server 4.0.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 WebSphere Commerce 5.4 level.

Ensuring no new messages are stored in MSGSTORE table

In WebSphere Commerce 5.4, 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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.)

Preparing the staging server for migration

You must synchronize your staging server database with your production WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 (CPYWCSSTG) to synchronize your databases. (This option will normally take much longer.) See the topic "Stage Copy utility command " in the WebSphere Commerce 5.4 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 (CPYWCSSTG) 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 WebSphere Commerce 5.4 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 7, "Migrating your WebSphere Commerce 5.4 database," on page 53. The process of migrating this database will be similar to your production database.

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 WebSphere Commerce 5.4

Before you proceed with the migration process documented in this Guide, you should perform a complete system backup of your working WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 after you complete the migration to WebSphere Commerce 5.5.

Backing up your WebSphere Commerce 5.4 system

To back up your WebSphere Commerce 5.4 system:

- Perform a complete system backup of your WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 database as described in “Backing up your databases” on page 6.
- In particular, backup the main WebSphere Commerce 5.4 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 *WCS54_userdir* (/QIBM/UserData/WebCommerce).

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 WebSphere Commerce 5.4 user installation directory (*WCS54_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 *WC54_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(('WC54_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('/WC54_userdir/instances/your_instance')
```

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

- The following important WebSphere Commerce 5.4 files and directories under the *WC54_installdir* directory. 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 26, however it is still recommended that you back these files up to a location that you can access:
 - `properties/*`
 - `xml/product.*`
 - `PaymentManager/profile.dtd`
 - `bin/cfg.passwd`
- The following important WebSphere Commerce 5.4 files and directories under the *WCS54_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`
- 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 - Express Developer Edition 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 WebSphere Commerce 5.4 system, including the database. See Chapter 2, “Backing up WebSphere Commerce 5.4,” on page 5 on how to perform a system backup.

If you need to stop any of your WebSphere Commerce 5.4 software components, see the appendix of the appropriate Installation Guide for your product and platform for the detailed steps.

Mapping the WebSphere Commerce 5.4 to WebSphere Commerce 5.5 IBM software

The following table maps the version levels and the installation paths between WebSphere Commerce 5.4 and WebSphere Commerce 5.5 for most of the software packaged with those versions.

Table 1. Supplied software levels and default installation paths for WebSphere Commerce for iSeries

Software	WebSphere Commerce 5.4	WebSphere Commerce 5.5
WebSphere Commerce	Version 5.4 /QIBM/ProdData/WebCommerce	Version 5.5 /QIBM/ProdData/CommerceServer55
IBM WebSphere Application Server — Advanced Edition	Version 4.0.2 (Version 4.0.1 plus PTF and eFixes) /QIBM/ProdData/WebASAdv4	Version 5.0 /QIBM/ProdData/WebAS5/base
IBM SDK for Java	Version 1.3.1 /QIBM/ProdData/Java400/jdk13	Version 1.3.1
WebSphere Commerce Payments (Formerly IBM WebSphere Payment Manager)	Version 3.1.2 /QIBM/ProdData/PymSvr	Version 5.5 /QIBM/ProdData/CommercePayments/V55
IBM WebSphere Commerce Analyzer	Version 5.2	Version 5.5
IBM SecureWay® Directory Server	Version 3.2.1	Version 3.2.2
Blaze Advisor™ Rule Server	Version 3.1.3 /QIBM/ProdData/WebCommerce/blaze	Version 4.5.5 /QIBM/ProdData/CommerceServer55/blaze

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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 machine. However, the approach does require additional machine resources. With this approach, you can keep your WebSphere Commerce 5.4 system running until your WebSphere Commerce 5.5 system is fully operational, and all required assets are transferred, migrated, and fully tested. At this point, you can shut down your WebSphere Commerce 5.4 system and start up your WebSphere Commerce 5.5 system, thereby switching over the machines. You can redeploy the WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 12.



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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4. 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 - Express Installation Guide*. For information on installing the WebSphere Commerce 5.5 system, see the chapters on installation in the *WebSphere Commerce - Express Installation Guide*.

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 - Express 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 database.
- Space to hold a backup copy of the WebSphere Commerce 5.5 database at various stage of the migration.
- 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 - Express 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 - Express 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

WebSphere Commerce 5.4, 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.

Exporting the WebSphere Commerce 5.4 enterprise application

Before you begin the transition to WebSphere Application Server 5.0, you need to export your existing WebSphere Commerce 5.4 enterprise applications (`WC_Enterprise_App_instance_name.ear`, for example, `WC_Enterprise_App_demo.ear`) while you are at the previous WebSphere Application Server 4.0.x level. You should also export the previous Payment Manager ear file. The WebSphere Commerce 5.5 WCIM instance migration tool will need to access the exported application in order to migrate it to the WebSphere Commerce 5.5 level.

You can export your application to a location of your choice:

1. Select the Enterprise Applications folder in the tree view of the WebSphere Application Server 4.0.x administrative console. The details view of the console will display the list of the installed applications (including those for WebSphere Commerce 5.4 and Payment Manager).
2. Right-click your application in the details view and select **Export Application** from the menu.
3. In the Export Application dialog that opens:
 - a. From the **Node** drop-down list, select a node to which the application should be exported.
 - b. In the **Export directory** field, specify the target drive and directory for the application. (for example, `/WAS40_backup`)
 - c. Click **OK**.

The application will be exported to the specified node and directory, and will have the name *enterpriseApplicationName.ear*.

4. Copy the exported `WC_Enterprise_App_instance_name.ear` file into the migration working directory on WebSphere Commerce 5.5. This is specified by the `WORK_DIR` variable in the `wcimenv.sh` file. (`WORK_DIR` is typically set to `WC55_userdir/temp`.) See Table 2 on page 28 for the definition of `WORK_DIR`.

Note: On iSeries, you need to remove the enterprise application for Commerce and Payments from the WebSphere Application Server 4.0 Administration Console after you have exported them. Otherwise, when you run the `WASPostUpgrade` tool, you will receive exceptions.

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 WebSphere Commerce 5.4 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/iseries/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>

In addition, you will need to export your WebSphere Commerce 5.4 and Payment Manager enterprise application (EAR file) as documented in “Exporting the WebSphere Commerce 5.4 enterprise application” on page 10. The WCIM tool requires access to the exported EAR file in “Migrating your instance configuration using WCIM” on page 26, in order to migrate your instance.



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 `<enterprise-application action="create" name="instance_name - WebSphere Commerce Enterprise Application">` entry and remove the entire node:

```
<enterprise-application action="create" name="instance_name -  
    WebSphere Commerce Enterprise Application">  
    ...  
</enterprise-application>
```

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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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/iseries/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.

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 4.0.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 4.0.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 5, “Migrating your Commerce instance configuration,” on page 23.

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

This section describes how to upgrade the software for your WebSphere Commerce 5.4 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 13
- “Upgrading to WebSphere Commerce 5.5” on page 14

Upgrading your hardware

For information on upgrading your hardware, see the pre-installation section of the *WebSphere Commerce - Express 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 WebSphere Commerce 5.4, 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:

Exporting the WebSphere Commerce 5.4 enterprise application

Before you begin the transition to WebSphere Application Server 5.0, you need to export your existing WebSphere Commerce 5.4 enterprise applications (`WC_Enterprise_App_instance_name.ear`, for example, `WC_Enterprise_App_demo.ear`) while you are at the previous WebSphere Application Server 4.0.x level. You should also export the previous Payment Manager .ear file. The WebSphere Commerce 5.5 WCIM instance migration tool will need to access the exported application in order to migrate it application to the WebSphere Commerce 5.5 level.

You can export your application to a location of your choice:

1. Select the Enterprise Applications folder in the tree view of the WebSphere Application Server 4.0.x administrative console. The details view of the console will display the list of the installed applications (including those for WebSphere Commerce 5.4 and Payment Manager).
2. Right-click your application in the details view and select **Export Application** from the menu.
3. In the Export Application dialog that opens:
 - a. From the **Node** drop-down list, select a node to which the application should be exported.
 - b. In the **Export directory** field, specify the target drive and directory for the application. (for example, /WAS40_backup)
 - c. Click **OK**.

The application will be exported to the specified node and directory, and will have the name *enterpriseApplicationName.ear*.

4. After you install WebSphere Commerce 5.5, you will need to copy the exported *WC_Enterprise_App_instance_name.ear* file into the WebSphere Commerce 5.5 migration working directory. This is specified by the `WORK_DIR` variable in the `wcimenv.sh` file. (`WORK_DIR` is typically set to *WC55_userdir/temp*.) See Table 2 on page 28 for the definition of `WORK_DIR`.

Note: On iSeries, you need to remove the enterprise application for Commerce and Payments from the WebSphere Application Server 4.0 Administration Console after you have exported them. Otherwise, when you run the `WASPostUpgrade` tool, you will receive exceptions.

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/iseries/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/iseries/v1r1m0/websphere/ic2924/index.htm?info/rzaiz/50/migration/mig.htm>

In addition, you will need to export your WebSphere Commerce 5.4 and Payment Manager enterprise application (EAR file) as documented in “Exporting the WebSphere Commerce 5.4 enterprise application” on page 12. The `WCIM` tool requires access to the exported EAR file in “Migrating your instance configuration using `WCIM`” on page 26, in order to migrate your instance.



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 `<enterprise-application action="create" name="instance_name - WebSphere Commerce Enterprise Application">` entry and remove the entire node:


```
<enterprise-application action="create" name="instance_name -
WebSphere Commerce Enterprise Application">
...
</enterprise-application>
```
 - 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/iseries/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 4.0.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 4.0.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 WebSphere Commerce 5.4 installation. Coexistence of the two installations are fully supported. Before beginning a migration, you must stop all operations against that instance. For information on these pre-requirements, see the *WebSphere Commerce - Express Installation Guide*.

For information on installing the WebSphere Commerce 5.5 system, see the chapters on installation in the *WebSphere Commerce - Express 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 2.
2. Do not delete your previous WebSphere Commerce 5.4 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 5, “Migrating your Commerce instance configuration,” on page 23.

Chapter 4. Minimizing downtime during coexistence migration

If you wish to implement coexistence of WebSphere Commerce 5.4 and WebSphere Commerce 5.5 on the same machine during migration, there are some additional steps and considerations that you need to perform as described in this chapter. By coexistence, it is meant that both an existing WebSphere Commerce 5.4 and a WebSphere Commerce 5.5 instance will run concurrently on the same machine. You must ensure that your system has enough memory and storage for both installations to run simultaneously.

Coexistence migration means that you can migrate one instance at a time without stopping other instances you may have running on the same machine. If you only have one machine, this means that you must do an *in-place* migration for each instance. Performing an in-place migration with coexistence enables you to keep your original WebSphere Commerce 5.4 instance running, while you migrate and test a *copy* of that instance along with the migration process. Note that testing the migration process can take several days or even in the order of weeks, depending on the amount of customization you have made to an instance. For many installations, this amount of downtime may be excessive.

This section is for those customers who wish to make a duplicate copy of the instance to be migrated on the same machine in order to minimize downtime for that instance. The migrated instance can coexist with the original instance, until such time that you are ready to migrate your production server after sufficient testing of the migrated instance. Alternatively, you can switchover your migrated instance as the WebSphere Commerce 5.5 production instance. Migrating the original instance (and thereby preserving the original name) may be an advantage to some customers, whereas using the new instance means you would only have to remigrate the database part.

Assigning a host name and IP address

Assign a new host name and IP address to the current node. This will be the host name you will use to configure a new WebSphere Commerce 5.4 instance.

When you create a new WebSphere Commerce 5.4 in the next step, you need to assign the instance a different host name and IP address from the existing WebSphere Commerce 5.4 instance. Configuration Manager does not allow you to create a new instance by using same host name.

The idea here is to create a new WebSphere Commerce 5.4 instance with a new name, but ensure that it contains all the data that the original WebSphere Commerce 5.4 instance contained.

Creating a new WebSphere Commerce 5.4 instance

Use the WebSphere Commerce 5.4 Configuration Manager to create a new WebSphere Commerce 5.4 instance. See the chapter on using the Configuration Manager in the *WebSphere Commerce Installation Guide* for Version 5.4, for the steps to create a new WebSphere Commerce 5.4 instance.

Note: You must use the same instance password and merchant key as the existing WebSphere Commerce 5.4 instance.

Dropping the new WebSphere Commerce 5.4 instance database

Drop the database that has been created for the new WebSphere Commerce 5.4 instance, using the following SQL:

```
DROP SCHEMA new_instance_name
```

Backing up and restoring the database for the instance to be migrated

To restore the database under a new name, complete the following steps:

1. Modify your new instance's user profile to have *SECOFR authority, using the following command:

```
CHGUSRPRF USRPRF(new_instance_name) USRCLS(*SECOFR) SPCAUT(*USRCLS)  
CURLIB(*CRTDFT)
```

2. Signon with a profile that has *SECOFR authority.
3. Enter the QSH command window (STRQSH), 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
```

where *host_name* is the name of the machine where the database resides, *database_name* is the name of the database, as displayed in the relational database directory, *new_instance_name* is the name of the newly created instance, *new_instance_passwd* is the password for the newly created instance, *old_instance_name* is the name of the schema to be copied, and *empty_native_lib* is an existing empty native library.

This step will create a saveFile of the original instance in the specified saveFile library (*empty_native_lib*) and use this savefile to create a database schema with the name *new_instance_name*.

4. Update the stored procedures using the following WebSphere Commerce 5.4 command from the /QIBM/ProdData/WebCommerce/bin directory:

```
update_storedprocedures database_name new_SCHEMA new_instance_passwd
```

5. Change your new instance user profile back to its original settings using the following command:

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

Exporting the EAR file of the instance to be migrated

Export the .ear file from the existing WebSphere Commerce 5.4 instance. It will be used to replace the .ear file in the newly created WebSphere Commerce 5.4 instance. For steps on how to do this, see "Exporting the WebSphere Commerce 5.4 enterprise application" on page 12

Note: Save the .ear file to the following directory:

```
/QIBM/UserData/WebCommerce/instances/old_instance_name/temp/earfilename.ear
```

Removing the EAR of the new WebSphere Commerce 5.4 instance

Delete the .ear folder for the newly created WebSphere Commerce 5.4 instance by deleting the enterprise application for that instance in the WebSphere Application Server Administration Console.

Once you have removed the enterprise application, ensure that the instance's .ear folder is no longer present. By default, this folder is in the following location:

```
/QIBM/UserData/WebASAdv4/WAS_instance_name/installedApps/  
WC_Enterprise_App_instance_name.ear
```

If the WC_Enterprise_App_instance_name.ear folder is still present, you need to either delete it, or rename it.

Installing the EAR of the instance to be migrated

The following are the steps to clone the EJB portion of WebSphere Commerce and to ensure that the WebSphere Commerce tools will function after the instance has been duplicated.

1. Export the configuration information for enterprise beans from the existing WebSphere Commerce 5.4 instance:
 - a. Copy the was.export.app.xml file located in the /QIBM/ProdData/WebCommerce/xml/config folder, to a temporary folder such as /home/users_home_dir, and open it in a text editor. In this file replace all occurrences of \$Enterprise_Application_Name\$ with instance_name - WebSphere Commerce Enterprise Application where instance_name is the name of your previous WebSphere Commerce instance (for example, demo). Save this file.
 - b. Run the following commands:
 - 1) Launch the QShell environment: STRQSH
 - 2) Switch to the following directory:

```
cd /QIBM/ProdData/WebASAdv4/bin
```
 - 3) Export the configuration content of the was.export.app.xml file using the following command:

```
xmlConfig -export /home/users_home_dir/OutputFile.xml  
-partial /home/users_home_dir/was.export.app.xml  
-adminNodeName WAS_host_name -nameServiceHost WAS_host_name  
-nameServicePort WAS_admin_port -instance WAS_instance_name
```

Note: The WAS_host_name is case sensitive. Ensure that it matches the host name in CFGTP option 12.

2. Open the /home/users_home_dir/OutputFile.xml in a text editor, and globally replace the current instance_name with new_instance_name, except in the XML section <ear_file_name>...</ear_file_name>. The information between the <ear_file_name>...</ear_file_name> tags should be the location and name of the current (that is, previous) instance exported .ear file. In the rest of the file, ensure that the .ear file location is set correctly, for example:

```
/QIBM/UserData/WebCommerce/instances/old_instance_name/temp/earfilename.ear
```

Notes:

- a. This assumes you use the same password for both the new and old instances.
 - b. The instance name is case sensitive in some parts of the OutputFile.xml file. So when you do the global replace, use the exact case of the new_instance_name.
3. Run the following command to import EJBs:

```
XMLConfig -import /home/users_home_dir/OutputFile.xml  
-adminNodeName WAS_host_name -nameServiceHost WAS_host_name  
-nameServicePort WAS_admin_port -instance WAS_instance_name
```
 4. After the import is complete, modify the web.xml files in the following folders:

```

/QIBM/UserData/WebASAdv4/default/installedApps/
  WC_Enterprise_App_new_54_instance_name.ear/wctools.war/WEB-INF
/QIBM/UserData/WebASAdv4/default/installedApps/
  WC_Enterprise_App_new_54_instance_name.ear/wcstores.war/WEB-INF

```

Open the web.xml file in a text editor, and globally replace the previous *instance_name* with the new *instance_name*

5. Copy all the *.sar files from the /QIBM/UserData/WebCommerce/instances/*old_WC54_instance_name*/sar directory to the /QIBM/UserData/WebCommerce/instances/*new_WC54_instance_name*/sar directory
6. Update the SCHCONFIG table for each .sar (store archive) file so that any publish information remaining in this table will be reflected properly in the WebSphere Commerce Administration Console of the copied instance.
 - a. Using the iSeries Navigator that comes with iSeries Access for Windows, expand **My Connections** -> *HostName* -> **Database** -> **Libraries**.
 - b. Select **Libraries**, and right-click to select the "Select Libraries to Display" menu.
 - c. In the "Select Libraries to Display" window, enter the new instance name in the "Enter libraries" field. Click **Add** and then **OK**.
 - d. Select the library you just added (*new_instance_name*); Navigator will refresh the right panel to give you a list of the tables in the WebSphere Commerce schema.
 - e. Double click the SCHCONFIG table from the right panel.
 - f. Correct the path in the SCCQUERY field where the SCCPATHINFO field contains the "ScheduledPublishArchive" entry, by updating the path as follows. If the SCCPATHINFO field does not contain the string "ScheduledPublishArchive", then you do not need to make the following changes:

From:

```

-DBTYPE=DB2&-SAR=/QIBM/UserData/WebCommerce/instances/old_WC54_instance_name
/sar/WebFashion.sar&-LOGONID=wcsadmin&-XML=ALL&-DBNAME=1p12ut4&-MODE=INSERT&-DBPWD
=xxxxxxx&-ASSET=/QIBM/UserData/WebASAdv4/default/installedApps/
WC_Enterprise_App_old_WC54_instance_name.ear/wcstores.war=webapp.zip,
/QIBM/UserData/WebASAdv4/default/installedApps/
WC_Enterprise_App_old_WC54_instance_name.ear/
wcstores.war/WEB-INF/classes=properties.zip,&-DBUSER=&DBHOST;=&-LOCALE=en_US

```

To:

```

-DBTYPE=DB2&-SAR=/QIBM/UserData/WebCommerce/instances/new_WC54_instance_name
/sar/WebFashion.sar&-LOGONID=wcsadmin&-XML=ALL&-DBNAME=1p12ut4&-MODE=INSERT&-DBPWD
=xxxxxxx&-ASSET=/QIBM/UserData/WebASAdv4/default/installedApps/
WC_Enterprise_App_old_WC54_instance_name.ear/wcstores.war=webapp.zip,
/QIBM/UserData/WebASAdv4/default/installedApps/
WC_Enterprise_App_old_WC54_instance_name.ear/
wcstores.war/WEB-INF/classes=properties.zip,&-DBUSER=&DBHOST;=&-LOCALE=en_US

```

Note that the above examples have been split to display on the page. They are supposed to be on one contiguous line.

The above changes are needed for all .sar files associated with the instance.

Alternatively, you may use SQL to update each record.

7. For any other customized settings in the WebSphere Application Server portion of the existing instance, you may have to retrofit these settings to the new instance.

Testing the new WebSphere Commerce 5.4 instance

If you have any customized code that refers to the instance name, IP address, or other information that has changed during this process, be sure to update your code with the new instance's information. Start your WebSphere Commerce server, and test that your new WebSphere Commerce 5.4 instance functions properly without errors..

Proceeding with instance and data migration

After you have completed the above steps, proceed with instance and data migration for the instance that you wish to migrate as documented in Chapter 5, "Migrating your Commerce instance configuration," on page 23, Chapter 6, "Before migrating your database," on page 47, and Chapter 7, "Migrating your WebSphere Commerce 5.4 database," on page 53, along with the remaining chapters in this Guide if applicable to your system.

Once you are satisfied with the migration process, and you are ready to migrate your production server, you have two options:

1. Migrate your original WebSphere Commerce 5.4 production instance and go live with it. This option may be useful if you do not wish to change your instance name. With this option, you would:
 - a. Stop your production server
 - b. Follow the migration process with your original instance.
2. Migrate the copy of your WebSphere Commerce 5.4 instance and go live with it. This option may be useful if you have modified a large portion of your code and do not wish to remigrate the WebSphere Commerce instance. With this option, you would:
 - a. Stop your production server.
 - b. Drop the database for the new instance as described in "Dropping the new WebSphere Commerce 5.4 instance database" on page 18.
 - c. Make a copy of your database by following the steps in "Backing up and restoring the database for the instance to be migrated" on page 18.
 - d. Migrate the database for this instance as described in Chapter 7, "Migrating your WebSphere Commerce 5.4 database," on page 53 before going live.
 - e. Modify your host name back to that of your original instance before going live. See "Changing your host name" on page 61 for guidance on changing your host name.

Chapter 5. Migrating your Commerce instance configuration

This section describes how to migrate your WebSphere Commerce 5.4 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 - Express Installation Guide* for your database and platform.

The WebSphere Commerce 5.4 instance migration is performed by starting from a base WebSphere Commerce 5.5 instance (EAR) and adding in the required components from the WebSphere Commerce 5.4 instance. As there are some changes in the instance packaging between WebSphere Commerce 5.4 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.

Checking the WebSphere Commerce 5.4 product.dtd file

In the WebSphere Commerce 5.4 `product.dtd` file, located in the `WC54_installdir/xml` directory, search for the string "configInstaneList" and modify it to "configInstanceList".

Checking the WebSphere Commerce 5.4 product.xml file

Ensure the following line (in bold) exists at the top of the WebSphere Commerce 5.4 `product.xml` file (located in the `WC54_userdir/xml` directory) as the second line. If it is missing, manually add it in:

```
<?xml version="1.0"?>  
<!DOCTYPE websphere SYSTEM "product.dtd">  
<!-- This file contains the wcsuite component information at the installation time -->
```

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.)

```
<migrationFrom>
  <edition>
    <name>name</name>
  </edition>
  <version>5</version>
  <release>rel</release>
  <modification>mod</modification>
  <fixpak>fixpak</fixpak>
  <path>path</path>
  <altpath>alt_path</altpath>
</migrationFrom>
```

In the above:

- *name* is:
 - Pro** WebSphere Commerce Professional Entry Edition
- **5** is the version of WebSphere Commerce. The only possible value for this migration is 5.
- *rel* is:
 - 4** For the WebSphere Commerce 5.4 release.
- *mod* is one of:
 - 0** WebSphere Commerce 5.4.0.1
 - 0** WebSphere Commerce 5.4.0.2
 - 0** WebSphere Commerce 5.4.0.3
 - 0** WebSphere Commerce 5.4.0.4
 - 0** WebSphere Commerce 5.4.0.5
- *fixpak* is one of:
 - 0** WebSphere Commerce 5.4.0.0 or WebSphere Commerce 5.4.0.1 (See the **Note** at the end of this section on page 25.)
 - 2** WebSphere Commerce 5.4.0.2
 - 3** WebSphere Commerce 5.4.0.3
 - 4** WebSphere Commerce 5.4.0.4 or WebSphere Commerce 5.4.0.5 (See the **Note** at the end of this section on page 25.)

- *path* is the installation path for the previous WebSphere Commerce 5.4 system (*WC54_installdir*).
- *alt_path* is the *WC54_installdir* directory. This can either be the same value as *path* or be left blank.



To check your current release, use the following SQL query against the site table:

```
select * from SITE
```

View the results for the PRODUCTVERSION column to determine your release level.

The following is an example of what your *product.xml* file needs to contain if you are migrating from WebSphere Commerce 5.4.0.0 Professional Entry Edition:

```
<migrationFrom>
  <edition>
    <name>Pro</name>
  </edition>
  <version>5</version>
  <release>4</release>
  <modification>0</modification>
  <fixpak>0</fixpak>
  <path>/QIBM/ProdData/WebCommerce</path>
  <altpath>/QIBM/ProdData/WebCommerce</altpath>
</migrationFrom>
```

Note: If you are migrating from WebSphere Commerce 5.4.0.1 (WebSphere Commerce 5.4 plus fix pack 1), you need to use the same parameters that you would use if you are migrating from WebSphere Commerce 5.4.0.0. That is, the `<fixpak></fixpak>` value should be set to `0` and not `1`. So your `<migrationFrom>` section should appear consistent with the above example.

Similarly, if you are migrating from WebSphere Commerce 5.4.0.5 (WebSphere Commerce 5.4 plus fix pack 5), you need to use the same parameters that you would use if you are migrating from WebSphere Commerce 5.4.0.4. That is, the `<fixpak></fixpak>` value should be set to `4` and not `5`.

Steps before migrating your instance configuration

Before migrating your WebSphere Commerce 5.4 instance to WebSphere Commerce 5.5:

- The WCIM tool will back up the entire directory structure for your instance. If you have WebSphere Commerce 5.4 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 - Express Developer Edition 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 WebSphere Commerce 5.4 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 97

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 WebSphere Commerce 5.4 system. For an in-place migration, this is done on your WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 user'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 WebSphere Commerce 5.4 machines. In a remote migration, WebSphere Application Server 5.0 needs to be uninstalled from the WebSphere Commerce 5.4 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
- 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.
 - `WC54_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, `wcbkup54.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 2 where this variable is defined).

2. If your WebSphere Commerce 5.4 machine is remote from your WebSphere Commerce 5.5 machine, you have to manually copy the ZIP file, `wcbbackup54.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 WebSphere Commerce 5.4 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 2. 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 WebSphere Commerce 5.4 is on the same machine as WebSphere Commerce 5.5 (in-place).
 - `WCIM_INPLACE = false` when WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 is on the same machine as WebSphere Commerce 5.5 (in-place).
 - `WCIM_INPLACE = true` when WebSphere Commerce 5.4 is not on the same machine as WebSphere Commerce 5.5 (remote).

Table 2. Variables set in *wcimenv.sh* (continued)

Variable	Value	Description
WCIM_MIGRATE_FROM	54	Backup or migrate from WebSphere Commerce 5.4
	WPM221	Backup or migrate from Payment Manager 2.2.1. See “Migrating your Payments instance and database” on page 67 for details.
	WPM312	Backup or migrate from Payment Manager 3.1.2. See “Migrating your Payments instance and database” on page 67 for details.
	WCP313	Backup or migrate from WebSphere Commerce Payments 3.1.3. See “Migrating your Payments instance and database” on page 67 for details.
DB_TYPE	db2	The database type for DB2 databases.
WAS_PATH	<i>WAS40_installdir</i> or <i>WAS50_installdir</i>	The WebSphere Application Server installation path: <ul style="list-style-type: none"> For backups (where <code>WCIM_BACKUP</code> is true), specify the previous WebSphere Application Server installation path. For migration (where <code>WCIM_BACKUP</code> is false), specify the WebSphere Application Server 5.0 installation path.
JAVA_EXE	java	The SDK for Java executable.
WC_PATH	<i>WC54_installdir</i> or <i>WC55_installdir</i>	The WebSphere Commerce installation path: <ul style="list-style-type: none"> For backups (where <code>WCIM_BACKUP</code> is true), specify the WebSphere Commerce 5.4 installation path. For migration (where <code>WCIM_BACKUP</code> is false), specify the WebSphere Commerce 5.5 installation path.
WEBSERVER_PATH	<ul style="list-style-type: none"> For IBM HTTP Server: <i>HTTPServer1319_installdir</i> or <i>HTTPServer1326_installdir</i> 	<p>The Web server installation path:</p> <ul style="list-style-type: none"> For backups (where <code>WCIM_BACKUP</code> is true), specify the previous Web server path. For migration (where <code>WCIM_BACKUP</code> is false), specify the current Web server installation path. <p>Note: <code>WEBSERVER_PATH</code> referenced here is the production Web server path (QIBM/ProdData/HTTPPA) as opposed to the user Web server path (QIBM/UserData/HTTPPA). It is the same value for both WCIM backup and WCIM migration.</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 WCIM tool. (For example, <i>WC55_userdir/temp</i>)
MIG_FROM_WAS_INSTANCE		The previous WebSphere Application Server instance with which the Commerce instance was originally associated.

Table 2. Variables set in `wcimenv.sh` (continued)

Variable	Value	Description
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 67.
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	WebSphere Commerce 5.4 <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 WebSphere Commerce 5.4.

Pre-requirements for remote instance migration: If you are installing your WebSphere Commerce 5.5 software components on a machine that is remote from WebSphere Commerce 5.4, you should have completed the steps described in “Exporting the WebSphere Commerce 5.4 enterprise application” on page 10 and “Migrating your previous WebSphere Application Server configuration” on page 10.

Ensure WebSphere Application Server 5.0 is installed on both the WebSphere Commerce 5.5 and the WebSphere Commerce 5.4 machines. WebSphere Application Server 5.0 needs to be uninstalled from the WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 machine, for example `WC54_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 WebSphere Commerce 5.4 machine (`WC54_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 *instbackup54.xml* file to the *xml/migration* subdirectory under the working directory on your WebSphere Commerce 5.4 machine (*WC54_userdir/temp/xml/migration*).

Note: If migrating Payments, instead of copying *instbackup54.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 WebSphere Commerce 5.4 machine (*WC54_userdir/temp/bin*), copy the following files:
 - *wcim.sh*
 - *wcimenv.sh*
 - From the *WC54_installdir/xml* directory in the WebSphere Commerce 5.4 machine, copy the *product.xml* and *product.dtd* files to the *xml* subdirectory under the migration working directory (*WC54_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 WebSphere Commerce 5.4 machine (*WC54_userdir/temp/lib*).
3. On your WebSphere Commerce 5.4 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="54"
DB_TYPE="db2"

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...

WAS_PATH="WAS40_installdir"
WC_PATH="WC54_installdir"
WEBSERVER_PATH="HTTPServer1319_installdir"
ANT_PATH="WC54_installdir/temp/lib"
WORK_DIR="WC54_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 WebSphere Commerce 5.4 machine, 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]
```

where:

- *wcmenv_path* is the path where the wcmenv.sh is located on your machine. If this file is located in the same directory as wcm.sh, this parameter is not required.
5. Copy the wcbakup54.zip file from the working directory on the remote WebSphere Commerce 5.4 machine (for example, *WC54_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 WebSphere Commerce 5.4 machine as documented in step 2 on page 30.
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 WebSphere Commerce 5.4 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 wcm.sh script on your WebSphere Commerce 5.5 machine, search all the files in the wcbakup54.zip package for all occurrences of the WebSphere Commerce 5.4 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._EJB.xml*
- *instance_name_was_VH.xml*
- *cache.properties*

You should check all files in the *wbackup54.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="54"
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="wcmigration.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

...

```

Ensure the exported *WC_Enterprise_App_instance_name.ear* (referenced in step 4 on page 10) is available in the migration working directory specified by *WORK_DIR* in the *wcimenv.sh* file (for example, *WC55_userdir/temp*).

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 67.

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 WebSphere Commerce 5.4, you should have completed the steps described in “Exporting the WebSphere Commerce 5.4 enterprise application” on page 12 and “Manually migrating your previous WebSphere Application Server configuration” on page 13.

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="54"  
DB_TYPE="db2"
```

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

...

```
WAS_PATH="WAS50_installdir"  
WC_PATH="WC54_installdir"  
WEBSERVER_PATH="HTTPServer1319_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="wcimbackup.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 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.

Running WCIM for in-place migration:

Note: The Web server member in QUSRSYS/QATMHINSTC for the WebSphere Commerce 5.4 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 wcim.sh script for migration, update the environment variables in the wcimenv.sh file.

The wcimenv.sh file contains the following entries:

...

```

WCIM_BACKUP="false"
WCIM_INPLACE="true"
WCIM_MIGRATE_FROM="54"
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

```

...

Ensure the exported `WC_Enterprise_App_instance_name.ear` (referenced in step 4 on page 13) is available in the migration working directory specified by `WORK_DIR` in the `wcimenv.sh` file (for example, `WC55_userdir/temp`).

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 67.
- `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 7, “Migrating your WebSphere Commerce 5.4 database,” on page 53 and then you can verify that your instance is migrated in “Verifying successful instance and database migration” on page 62.

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 G, “Troubleshooting,” on page 109 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 ... migratejps`
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 WebSphere Commerce 5.4 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)`

For details on migrating to WebSphere Commerce Payments, see Chapter 9, “Migrating to Commerce Payments,” on page 63.

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.

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 (`WC_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, `WC_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`

- com.ibm.CORBA.iiop.noLocalCopies
 - com.ibm.wca.logging.configFile
 - com.ibm.websphere.ejbcontainer.fbpkAlwaysReadOnly
 - com.ibm.ws.classloader.ejbDelegationMode
 - com.ibm.wca.MassLoader.MassLoadOutputFilesDir
5. Add the following new properties (if they are not present on your system). Click **New** to add them:
- | Name | Value |
|----------------------------------|--|
| com.ibm.servlet.file.esi.timeOut | 0 |
| os400.define.class.cache.file | /QIBM/ProdData/CommerceServer55/lib/classcache.jar |
| os400.define.class.cache.hours | 9999 |
| os400.define.class.cache.maxpgms | 40000 |
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 WebSphere Commerce 5.4 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_IMPL_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 WebSphere Commerce 5.4 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 JAACL 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, jdbc/WC_instance_name WebSphere Commerce 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/WC_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:** jdbc/WC_instance_name WebSphere Commerce 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 "jdbc/" 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*: jdbc/WC_*instance_name* WebSphere Commerce 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_ <i>instance_name</i> _tools
OrganizationAdministration	VH_ <i>instance_name</i> _Orgadmin
SiteAdministration	VH_ <i>instance_name</i> _Admin
Stores	VH_ <i>instance_name</i>

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 39, 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

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 on page 43 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 wcmWasConfig.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_iserie_ear.sh cell_name wc_instance was_instance
```

For example, from the `/QIBM/ProdData/CommerceServer55/bin` directory run:


```
chgaut_iserie_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 - Express Developer Edition 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.

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 - Express 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 - Express Installation Guide*.

Adjusting the `WC_instance_name.ear` path

If you choose to switch from the default WebSphere Application Server instance to a non-default WebSphere Application Server instance (or switch from a non-default WebSphere Application Server instance to the default WebSphere Application Server instance) during migration, the `WC_instance_name.ear` path in the `WCInstanceName.xml` and `httpd.conf` files will not be migrated correctly.

For example, if you switch from the default WebSphere Application Server instance to a non-default instance, change:

```
/QIBM/UserData/WebAS5/Base/default/installedApps/MYHOST/WC_instance_name.ear
```

to:

```
/QIBM/UserData/WebAS5/Base/mywas/installedApps/MYHOST_mywas/WC_instance_name.ear
```

If you switch from a non-default WebSphere Application Server instance to the default WebSphere Application Server instance, change:

```
/QIBM/UserData/WebAS5/Base/mywas1/installedApps/MYHOST_mywas1/WC_instance_name.ear
```

to:

```
/QIBM/UserData/WebAS5/Base/default/installedApps/MYHOST/WC_instance_name.ear
```

Use Configuration Manager to make the above adjustments to the `WC_instance_name.ear` path:

1. Launch the WebSphere Commerce 5.5 Configuration Manager
2. Expand **WebSphere Commerce** -> `host_name` -> **Commerce** -> **Instance List** -> `instance_name` -> **Instance Properties**.
3. Select **WebServer** in the left panel, then click the **Advanced** tab in the right panel and adjust all paths (for the `wcstore` and `wcs` aliases) as required for both the secure and non-secure ports.
4. After making your updates, click **Apply**.

Enabling WebSphere Application Server EJB Security

If you had WebSphere Application Server EJB Security enabled in WebSphere Commerce 5.4, 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 *WC54_userdir/instances/instance_name/web* directory to the *WC55_userdir/web* directory.

Chapter 6. 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 7, “Migrating your WebSphere Commerce 5.4 database,” on page 53.



- Ensure that you have performed the steps in Chapter 1, “Before you migrate from WebSphere Commerce 5.4,” on page 1 including the steps in “Ensuring no new messages are stored in MSGSTORE table” on page 2 as well as the other sections in “Premigration actions” on page 2, while you had an operational WebSphere Commerce 5.4 system.
- For details on the database schema differences between WebSphere Commerce 5.4 and WebSphere Commerce 5.5 refer to the WebSphere Commerce - Express 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/QSQRN) JRNRCV(*GEN) RCVSIZOPT(*MAXOPT2)
```

where *instance* is the name of your database schema, and QSQRN 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 could only happen if your schema was originally created on Commerce Suite 5.1. 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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 WebSphere Commerce 5.4, 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 WebSphere Commerce 5.4.

The following table lists the newly added message types for WebSphere Commerce 5.5, compared with WebSphere Commerce 5.4 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 3. MSGTYPE_ID and NAME additions to MSGTYPES table in WebSphere Commerce 5.5

MSGTYPE_ID	NAME
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

Note that not all the above entries will show up after the data migration. CollabEmail is only available under Business Edition.

Running the database preparation script

The database preparation script also attempts to remove database triggers and reapplies all WebSphere Commerce 5.4 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.

When you run the premigration script, you will need to take the actions suggested that are output to the screen, or that are contained in the premigratedb54.log file.

Once you have fixed all necessary items, back up your updated WebSphere Commerce 5.4 databases once again as described in Chapter 2, "Backing up WebSphere Commerce 5.4," on page 5, 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:
`premigratedb54.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 WebSphere Commerce 5.4 database that you are migrating, as displayed in the relational database directory.
- *db_userID* is the name of the instance user profile of the WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 database that you are migrating.

For example, to run the database preparation script for WebSphere Commerce 5.4 databases, you can use the following command:

```
premigratedb54.sh db2 dbname instance instance_password
```

- ___ 4. Check the `premigratedb54.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 `premigratedb54.log` file generated in the directory for additional messages or errors. You need to resolve any errors that are flagged before proceeding.

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 6.

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 99.

Sample `premigratedb.log` file

The following is a partial example of the `premigratedb.log` output when it is run against a WebSphere Commerce 5.4 database. Note in this example, that no rows have exceeded length limits and the premigration ran successfully:

```
Info: Logging started in VERBOSE mode.
Event: Executing command: preChecking
Info: 0 rows of the table account exceeds length limits comments.
Info: 0 rows of the table acrelgrp exceeds length limits conditions.
Info: 0 rows of the table acresgrp exceeds length limits conditions.
Info: 0 rows of the table baseitmdsc exceeds length limits longdescription.
Info: 0 rows of the table bzrpentstg exceeds length limits value.
Info: 0 rows of the table calcodedsc exceeds length limits longdescription.
:
:
```

Event: Pre migration ends.
Event: Migration has terminated successfully.

Dropping custom constraints

If you have customized tables that contain foreign key links to the shipped WebSphere Commerce 5.4 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 55.

1. Identify all new tables that you have added, as well as all the WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 tables, or from WebSphere Commerce 5.4 tables to the new tables, or all new constraints you have added to the WebSphere Commerce 5.4 tables.

To determine the constraints that you have added to the WebSphere Commerce 5.4 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 ix. 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.

Preconditions for running the database migration script for WebSphere Commerce 5.4 databases

Before running the database migration tool, you must meet the following pre-requirements:

- You have backed up the database in a location that is accessible, should it be required.
- The database type you are using is supported by WebSphere Commerce.
- The database is at the correct release level, and all necessary fix packs have been applied.
- WebSphere Commerce 5.5 has been successfully installed on your machine.

In addition, before running the script, ensure that you have met the following preconditions, depending on your scenario:

Basic data migration (no custom data)

In this scenario, the WebSphere Commerce 5.4 database schema contains minor changes from the previous release. No changes or customizations have been made to the standard WebSphere Commerce 5.4 schema.

The preconditions for running the tool are:

- Your database schema remains uncustomized, that is, it is the same as the WebSphere Commerce 5.4 schema provided *out-of-the-box*.
- WebSphere Commerce 5.5 (in which migration package resides) is installed on your machine.
- The WebSphere Commerce 5.4 database is accessible (either locally or remotely).

Custom data migration

In this scenario, the WebSphere Commerce 5.4 database schema contains customizations relative to the WebSphere Commerce 5.4 schema provided *out-of-the-box*. For example, you have added a table or column to the standard schema, that needs to be handled by the data migration to WebSphere Commerce 5.5.

The preconditions for running the tool are:

- Your database schema is at the WebSphere Commerce 5.4 level and contains customized tables or columns.
- WebSphere Commerce 5.5 (in which migration package resides) is installed on your machine.
- The WebSphere Commerce 5.4 database is accessible (either locally or remotely).

In this scenario, prior to running the data migration tool you need to drop any custom triggers or constraints that you have added. You should note the triggers or constraints that you are dropping from the database since you will need to add them back in after you have migrated your database to WebSphere Commerce 5.5.

Chapter 7. Migrating your WebSphere Commerce 5.4 database

This chapter describes the steps to migrate your WebSphere Commerce 5.4 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 WebSphere Commerce 5.4,” on page 1, Chapter 2, “Backing up WebSphere Commerce 5.4,” on page 5, Chapter 3, “Upgrading your software,” on page 7, and Chapter 6, “Before migrating your database,” on page 47. In addition, it is mandatory that you run the database preparation script as documented in “Running the database preparation script” on page 49 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 WebSphere Commerce 5.4 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 95.
- 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 6.

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 WebSphere Commerce 5.4 schema to the WebSphere Commerce 5.5 level. For an overview of what the migration script does when migrating WebSphere Commerce 5.4 databases, see Appendix E, “Overview of the WCIM tool and the data migration script,” on page 97. For a summary of the database schema changes between WebSphere Commerce 5.4 and WebSphere Commerce 5.5, see the topic “Database schema changes in this release” in the WebSphere Commerce - Express 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:

```
migratedb54.sh db2 db_name db_userID password host_name backuplib
```

where:

- *db2* specifies that you are migrating a DB2 database.
- *db_name* is the WebSphere Commerce 5.4 database that you are migrating, as displayed in the relational database directory.
- *db_userID* is the instance user profile of the WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 database that you are migrating.
- *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, you can use the following command:

```
./migratedb54.sh db2 MY_DB MY_INSTANCE PASS myhost.montreal.ca MYBACKDIR
```

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

After you run the database migration script, you should check the log files as documented in “Checking the database migration log files” on page 99. 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 NEWBUFFERPOOL SIZE 20000 PAGESIZE 4K
ALTER TABLESPACE SYSCATSPACE BUFFERPOOL NEWBUFFERPOOL
```

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 WebSphere Commerce 5.4 database using the migration script, you need to restore any referential constraints that you dropped in “Dropping custom constraints” on page 51.

To run SQL statements, see “Running database scripts” on page ix. 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.

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. 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 WebSphere Commerce 5.4 version. Be sure that your instance user profile has ownership of the restored database library and tables.

Never restore a WebSphere Commerce schema with a different name than which 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 49.

Chapter 8. Post instance and database migration

The following sections describe migration actions you need to take to migrate your WebSphere Commerce 5.4 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 store assets”

iSeries Apache HTTP Server

WebSphere Commerce 5.4 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 store assets

This section describes how to migrate a published WebSphere Commerce 5.4 store to WebSphere Commerce.

Updating the WebSphere Commerce 5.4 JSP files

The following sections describe updates required to various JSP files from the WebSphere Commerce 5.4 sample stores, such as Infashion. If you reused the JSP files or your published WebSphere Commerce 5.4 stores, you need to make the updates as documented and redeploy the updated JSPs in your migrated stores.

Updating Coupon JSPs

The WebSphere Commerce 5.4 coupon function was introduced in B2C stores in the WebSphere Commerce 5.4 Commerce Enhancement Pack. In WebSphere Commerce 5.5 you are no longer allowed to add a coupon more than once. If you have a `catch_all` clause in your error checking code, you will receive this error as a generic error message. If you wish to handle the error message more completely, you should update the coupon error checking code (for example, in the `CouponError.jsp` file) in the following directory:

```
WAS_userdir/installedApps/cell_name/WC_instance_name.ear/Stores.war  
/store_name/web/ShoppingArea/CatalogSection/CategorySubsection
```

Add the following error checking to the coupons error checking code:

```
else if( error.equals(CouponConstant.PROMO_ACCEPTED_ERROR) )  
{  
%>  
    <%=infashiontext.getString("COUPON_ALREADY_ADDED")%>  
<%  
}
```

You also need to add the following to the properties file:

```
COUPON_ALREADY_ADDED = You have already added this coupon,  
or you have used this coupon before. You cannot add the same coupon twice
```

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 - Express 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"?>
```

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 WebSphere Commerce 5.4 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 WCS54_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 WebSphere Commerce 5.4, 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
 - CACHEIVL-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 `plugin-cfg.xml` file located in the `WAS50_installdir/plugins` directory:

```
<Property name="CacheLibrary" value="QWBCOMM/QYNGCACHE"/>
```

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

Web address change for Organization Administration Console

The launch Web address port and module for the Organization Administration Console Web address has changed in this release. The port has changed from 8000 to 8004, and the module has changed from `wctools` to `wcorgadmin`. Consider the following link to the Organization Administration Console in WebSphere Commerce 5.4:

```
https://host_name/webapp/wcs/wctools/servlet/...
```

In WebSphere Commerce 5.5, you need to use the following (changes are in bold):

```
https://host_name:8004/webapp/wcs/orgadmin/servlet/...
```

The port and web module are set in the `instance_name.xml` file. If you wish to reconfigure them, you can update the values in `instance_name.xml`. In WebSphere Commerce 5.5 the links to the tooling is dynamically generated based on the `instance_name.xml`. So launching the tools within your stores is not a migration issue.

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

If you are performing a remote migration or a coexistence 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 WebSphere Commerce 5.4 system prior to migration. Note that you will need to either take down your WebSphere Commerce 5.4 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 WebSphere Commerce 5.4 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_install_dir/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 5, “Migrating your Commerce instance configuration,” on page 23 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 9. 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 3.1.2
- Commerce Payments 3.1.3

The migration process will utilize the WebSphere Commerce 5.5 WCIM migration tool described in Chapter 5, “Migrating your Commerce instance configuration,” on page 23 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 3.1.2 or WebSphere Commerce Payments 3.1.3 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 3.1.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:

- With an EAR file (Payments 3.1.2 and above, as those versions supported WebSphere Application Server 4.0.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 3.1.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 3.1.x cassettes are recognized in a migrated Payments instance. However, if a new instance of Payments is created, the previous 3.1.x cassettes are not recognized as Payments has no knowledge as to the table layout for these cassettes.

Product rename in version 3.1.3

Payments Version 3.x included a number of different Payments releases:

- WebSphere Commerce Payments version 3.1.2 was a multi-platform release of Payment Manager on WebSphere Application Server 4.0.x
- WebSphere Commerce Payments 3.1.3 (or Payments 3.1.3) was released for the mainly for the purpose of renaming the product.

Given this information, any migration of Payments from Version 3.x has to be very specific about what version of Payment Manager it is migrating from. Even though Commerce Suite 5.1 installed version 2.2.x and WebSphere Commerce 5.4 installed version 3.1.2, all aspects of a Version 3.x migration are considered since upgrades to your existing Payments are possible. Thus, the migration handles version 3.1.2 or greater.

Payment Manager 3.1.1

Migration from Payment Manager 3.1.1 is not supported by WebSphere Commerce 5.5. If you are using Payment Manager 3.1.1, you need to apply the latest Payments PTF (version 3.1.3 at this time) before migrating to the WebSphere Commerce 5.5 level of Payments. If the migration detects version 3.1.1, it will flag an error in order to prevent migration from occurring.

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 4.0.x resides.
- WebSphere Application Server 4.0.x has been updated to WebSphere Application Server 5.0, with installed products migrated by WebSphere Application Server, with the exception of Payments.
- The Payments 3.1.x enterprise application has been exported and saved in the *previous_Payments_userdir/instance_name/Service* directory

To export the enterprise application, do one of the following:

- From the WebSphere Administrative Console, right-click on your Payments enterprise application and select **Export Application**. When prompted for Export directory, enter */QIBM/UserData/PymSvr/instance_name/Service*.

Attention: Before exporting the EAR file, user QEJB must be given *RWX authority to the following directories:

- */QIBM/UserData/PymSvr/instance_name*
- */QIBM/UserData/PymSvr/instance_name/Service*
- From the QShell, run the following commands
 - For version 3.1.2:


```
cd /QIBM/ProdData/WebASAdv4/bin

EARExpander -expandDir
/QIBM/UserData/WebASAdv4/WAS_instance_name/installedApps/
payments_instance_IBM_PaymentManager.ear
-ear /QIBM/UserData/PymSvr/payments_instance/Service/
payments_instance_IBM_PaymentManager.ear
-operation collapse
```
 - For version 3.1.3:

```

cd /QIBM/ProdData/WebASAdv4/bin

EARExpander -expandDir
/QIBM/UserData/WebASAdv4/WAS_instance_name/installedApps/
payments_instance_IBM_Payments.ear
-ear /QIBM/UserData/PymSvr/payments_instance/Service/
payments_instance_IBM_Payments.ear
-operation collapse

```

Note that the path to the *payments_instance_IBM_Payments.ear* has been split across two lines for display purposes.

- 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 websphere-sa-config SYSTEM "$XMLConfigDTDLocation$dsepxmlconfig.dtd" >
<websphere-sa-config>

<node name="NodeName" action="locate">

<application-server name="Payments_application_server_name" action="export">

</application-server>

</node>
</websphere-sa-config>

```

where *Payments_application_server name* is one of the following:

- For version 3.1.2 — "WPM INSTANCE WebSphere Payment Manager"
- For version 3.1.3: — "WPM INSTANCE WebSphere Commerce Payments"

For Payments 3.1.2 or 3.1.3, make a backup copy of */QIBM/UserData/PymSvr/PaymentsInstance/Service/instCreate.xml*, and then run the following command from the QShell:

```

/QIBM/ProdData/WebASAdv4/bin/XMLConfig -adminNodeName NodeName
-export /QIBM/UserData/PymSvr/PaymentsInstance/Service/instCreate.xml
-partial /tmp/new.xml -instance WebSphereInstanceName
-nameServiceHost WASServiceHost -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 4.0 Information Center at <http://publib.boulder.ibm.com/was400/40/AE/english/docs/>.

- You can access the previous Payment Manager and Commerce databases. If the previous Payments 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 91 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.”
- Removing virtual hosts if required as described in “Updating the Web server ports” on page 71.
- The Payments data migration script as described in “Migrating your Payments database” on page 72.

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 26, 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 26 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="/QIBM/ProdData/PymSvr"
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 3.1.2 — WPM312
Use WPM312 if your exported `.ear` file is of the form `PM_instance_name_IBM_PaymentManager.ear`
 - WebSphere Commerce Payments 3.1.3 — WCP313
Use WCP313 if your exported `.ear` file is of the form `PM_instance_name_IBM_Payments.ear`

For example:

```
...
WCIM_BACKUP="true"
WCIM_INPLACE="false"
WCIM_MIGRATE_FROM="WPM312 | WCP313"
```

```

DB_TYPE="db2"

...

export WCIM_BACKUP
export WCIM_INPLACE
export WCIM_MIGRATE_FROM
export DB_TYPE

...

WAS_PATH="WAS40_installdir"
WAS_INSTANCE="default"
WC_PATH="WC54_installdir"
WEBSERVER_PATH="HTTPServer1319_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/PymSvr"
export WPM_PATH

...

LOG_FILE="wcim_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 WebSphere Commerce 5.4 machine. For example

```

WORK_DIR="WC54_installdir/temp"
ANT_PATH="WORK_DIR/lib"
PATH=WAS40_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 `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 the `wcimenv.sh` path is the same as the `wcimenv.sh` path, use `."` to specify the current directory. For example:

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

4. The script backs up the version 3.1.x Payments instance files, including custom code, into the `/backupInst` directory on the WebSphere Commerce 5.4 (payments components) machine.

WCIM packages the files as a .zip file in the .../zip subdirectory of the working directory supplied in wcimenv.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 instbackupwpm312.xml or the instbackupwcp313.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="WPM312 | WCP313"  
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 `wcimenv.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 `wcimenv.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 `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 the following are *required* parameters for Payments migration:

- *wcimenv_path* is the path where the `wcimenv.sh` is located on your machine. If the `wcimenv.sh` path is the same as the `wcimenv.sh` path, use "." to specify the current directory. For example:
`./wcim.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 Payments database user ID and password. The parameters must be valid on both the system with the previous Payments 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 6.

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 93.

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: 3.1.2.0, 3.1.3.0, or 3.1.4.0.

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 - Express 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 3.1.x to WebSphere Commerce Payments, with an in-place migration.

Performed by

Site administrator

Preconditions

- You have installed WebSphere Application Server 5.0.
- You have exported the enterprise application and if required, the application server.
- 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 34 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 35 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 72 for the syntax to invoke the script.
8. Complete the post migration tasks in “Post instance and database migration steps” on page 73.

Post condition

The previous Payment Manager 3.1.x is successfully migrated to WebSphere Commerce Payments.

Migrating the previous Payment Manager on a remote machine

Description

To migrate Payment Manager 3.1.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.
- You have exported the enterprise application and if required, the application server.
- 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 WebSphere Commerce 5.4 machine.
2. For a remote backup, you need to copy the `instbackupwpm312.xml` or the `instbackupwcp313.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 30 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 32 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 wcimenv_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 72 for the syntax to invoke the script.
7. Complete the post migration tasks in “Post instance and database migration steps” on page 73.

Post condition

The previous Payment Manager 3.1.x is successfully migrated to WebSphere Commerce Payments.

Multiple WebSphere Commerce instances pointing to one instance of Payments

Description

To migrate Payment Manager 3.1.2 or Commerce Payments 3.1.3 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 exported the application server and enterprise application.
- 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 3.1.x, or Commerce Payments 3.1.3 is successfully migrated to WebSphere Commerce Payments.

Payments subsystem migration considerations

WebSphere Commerce 5.5 has introduced the following changes relative to WebSphere Commerce 5.4:

- Policy IDs for the CustomOffline_COD and CustomOffline_BillMe profile names are now supplied in WebSphere Commerce 5.5. In WebSphere Commerce 5.4, they were not supplied and you were required to create your own.

Table 4. CustomOffline profile names

Policy ID	Policy name	Profile name
-9991	CustomOffline_COD	WC51_CustomOffline_COD
-9990	CustomOffline_BillMe	WC51_CustomOffline_BillMe

- 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, Paymentech, has been introduced:

Table 5. Paymentech profile

Policy ID	Policy name	Profile name
-9980	Paymentech	WC_Paymentech

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 WebSphere Commerce 5.4, 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 WebSphere Commerce 5.4 `.../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 WebSphere Commerce 5.4 `.../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 WebSphere Commerce 5.4 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%' )
```
 5. 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

Chapter 10. 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”
- “Reconfiguring the database cleanup utility” on page 82
- “Access control policy subscriptions” on page 83

Configuring user roles

WebSphere Commerce 5.4 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 - Express 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 WebSphere Commerce 5.4 level, you should have run either the Stage Propagate or Stage Copy utilities as documented in “Preparing the staging server for migration” on page 3 to synchronize your staging server and your production server.

2. You should have successfully migrated your WebSphere Commerce 5.4 production database as documented in Chapter 7, "Migrating your WebSphere Commerce 5.4 database," on page 53.
3. You should have successfully migrated your WebSphere Commerce 5.4 staging server database, also following the steps documented in Chapter 7, "Migrating your WebSphere Commerce 5.4 database," on page 53.

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 WebSphere Commerce 5.4 configuration.

Note that in the migration process, the WebSphere Commerce 5.4 staging server tables, are renamed by appending `_54` to the original name. Thus, the WebSphere Commerce 5.4 staging server tables are preserved as:

- `STGSITETAB_54`
- `STGMERTAB_54`
- `STGMRSTTAB_54`
- `STAGLOG_54`

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 WebSphere Commerce 5.4 database cleanup utility table, by appending `_54` to the original name. Thus, the WebSphere Commerce 5.4 database cleanup utility table is preserved as `CLEANCONF_54`. 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*.

Business accounts and contracts

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>
```

Access control policy subscriptions

In WebSphere Commerce 5.4, with the out-of-the-box access control set up, guest users could shop at B2B stores. WebSphere Commerce 5.5 has introduced policy groups that allow you to prevent guest shoppers from shopping at B2B stores. In WebSphere Commerce 5.5, to allow a guest user to shop at a B2B store, it is just a matter of the B2B organization subscribing to a policy group with guest shopping policies. During migration, your existing policies are kept intact. Thus, if an existing customer had a policy that allowed guest shoppers to shop B2B stores, this is still the case after migration.

When WebSphere Commerce 5.4 is migrated, the migration process maintains the existing organization structure and policy behavior. The migration process converts policies of type 0 (standard) or null to type 2 (standard subscription). It also converts policies of type 1 (template) to type 3 (template subscription). In addition, the migration process associates these converted policies with the appropriate policy groups and subscribes these groups to the appropriate organizations to maintain your store's previous behavior.

Since stores in WebSphere Commerce 5.4 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 to, refer to the Access Control portion of the *WebSphere Commerce Security Guide*.

The migration process will also automatically update the access control access groups to make use of the new policy group feature in WebSphere Commerce 5.5. To preserve the old behavior, WebSphere Commerce 5.5 has introduced a new role-for-organization qualifier called "OrgAndAncestorOrgs". With this new qualifier, the access group evaluation will start from the organization that owns the resource and continue on with all its ancestor organizations. This is the same behavior of the previous template policies. The migration process will automatically replace the previous role-for-organization qualifier with this new qualifier.

In an effort to remove language dependencies in the access groups' role evaluation, WebSphere Commerce 5.5 uses the internal names of the roles (in the ROLE table) in its access group definition instead of the translated names in the ROLEDESC table (which is introduced in this release). The migration process also performs this update automatically for you.

Beginning from the organization that owns the resource, WebSphere Commerce 5.5 searches for policies of both the old and new types. WebSphere Commerce 5.5 stops searching policies of the new type once it reaches an organization that subscribes to one or more policy groups. However, it will continue searching for policies of the old type until the highest organization in the hierarchy has been reached. This is not affected by the policy subscription model and will behave exactly the same as WebSphere Commerce 5.4.

Note: If you have a custom policies XML files that you composed in WebSphere Commerce 5.4, it is recommended that you migrate these policies according to the process described above and assign them to the appropriate policy groups after the conversion. One item that is not mentioned in the process above is that in WebSphere Commerce 5.5, the DTD for the access group definition XML file has been changed from "ACUserGroups_en_US.dtd" to "ACUserGroups.dtd". Update your customized policies accordingly. Consult the Access Control section of the *WebSphere Commerce Security Guide* for more information on the new XML format supported in WebSphere Commerce 5.5.

Migrating WebSphere Commerce 5.4 overridden policies in the ACORGPOL table

With the introduction of policy groups in WebSphere Commerce 5.5, overridden policies are no longer supported. To override policies, WebSphere Commerce 5.5 creates new policy groups with the overridden policies removed, and subscribes the organization to these new policy groups instead.

The following steps briefly outline on how to override policies:

1. Identify the policy groups that contain the policies in the ACORGPOL table.
2. Create a copy of each of these policy groups.
3. From each copy of the policy groups, remove the unwanted policies.
4. Have the organization subscribe to the new policy groups as required.

For the detail instruction on how to create policy groups and assign policies to policy groups, see the Access Control section of the *WebSphere Commerce Security Guide*.

Roles removed in WebSphere Commerce 5.5

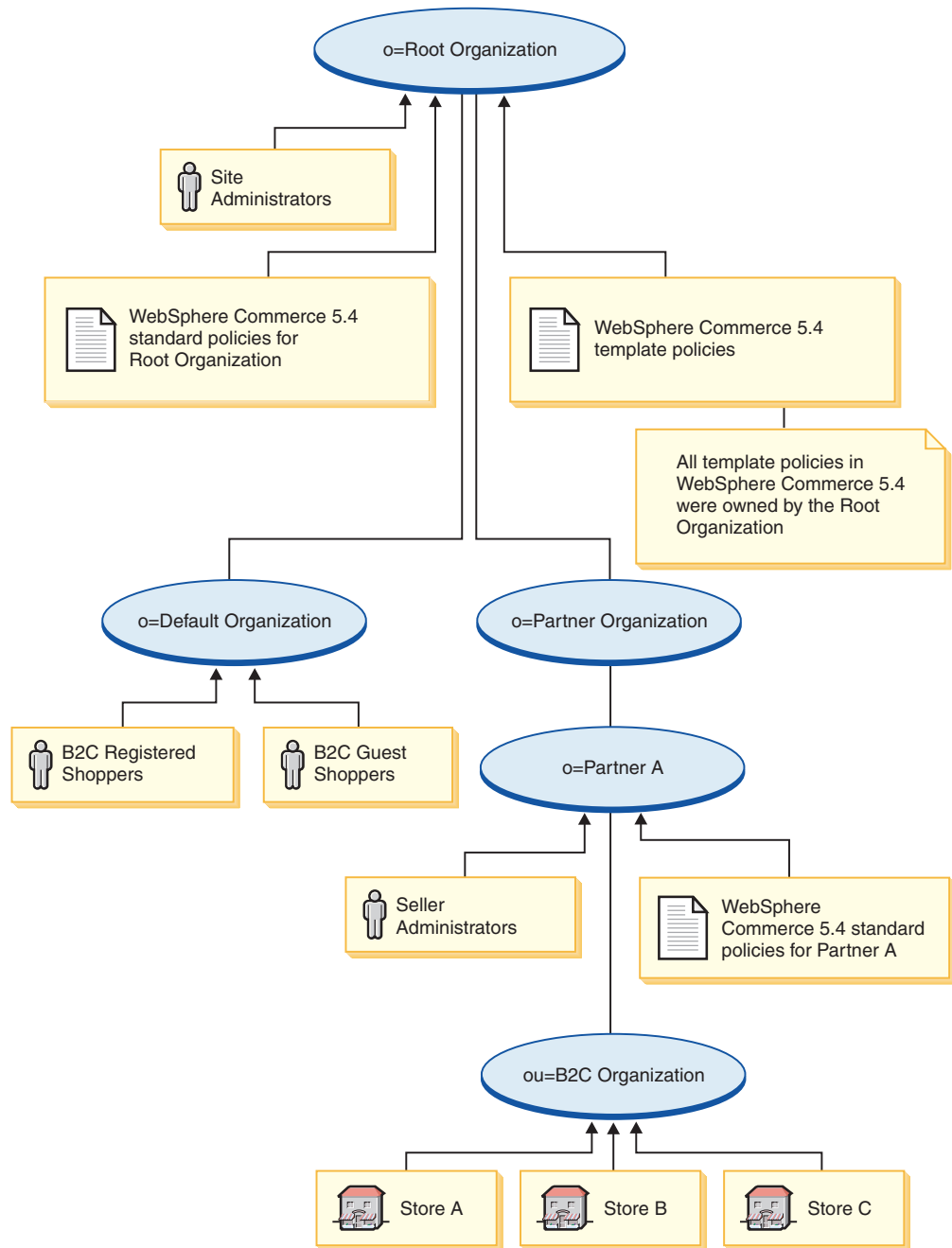
The Store Developer and Procurement Manager roles are not used in WebSphere Commerce 5.5. As such, WebSphere Commerce 5.5 bootstrap access control policies do not make reference to these roles. However, these roles and any access control policies referencing these roles are preserved in the system. They will be treated as customized policies and will be placed in policy groups other than the bootstrap policy groups.

WebSphere Commerce 5.4 setup (premigration)

The diagram below depicts a hypothetical setup of an existing WebSphere Commerce 5.4 site. Note the following items with respect to access control policy setup:

1. In WebSphere Commerce 5.4, all the customized template policies are owned and applied at the Root Organization.
2. Standard policies can be owned by any organization. The owning organization is where the policy was applied.

3. If a resource is accessed, all the policies starting at the resource owning organization and walking up the tree branch to the root organization can grant access to the resource.



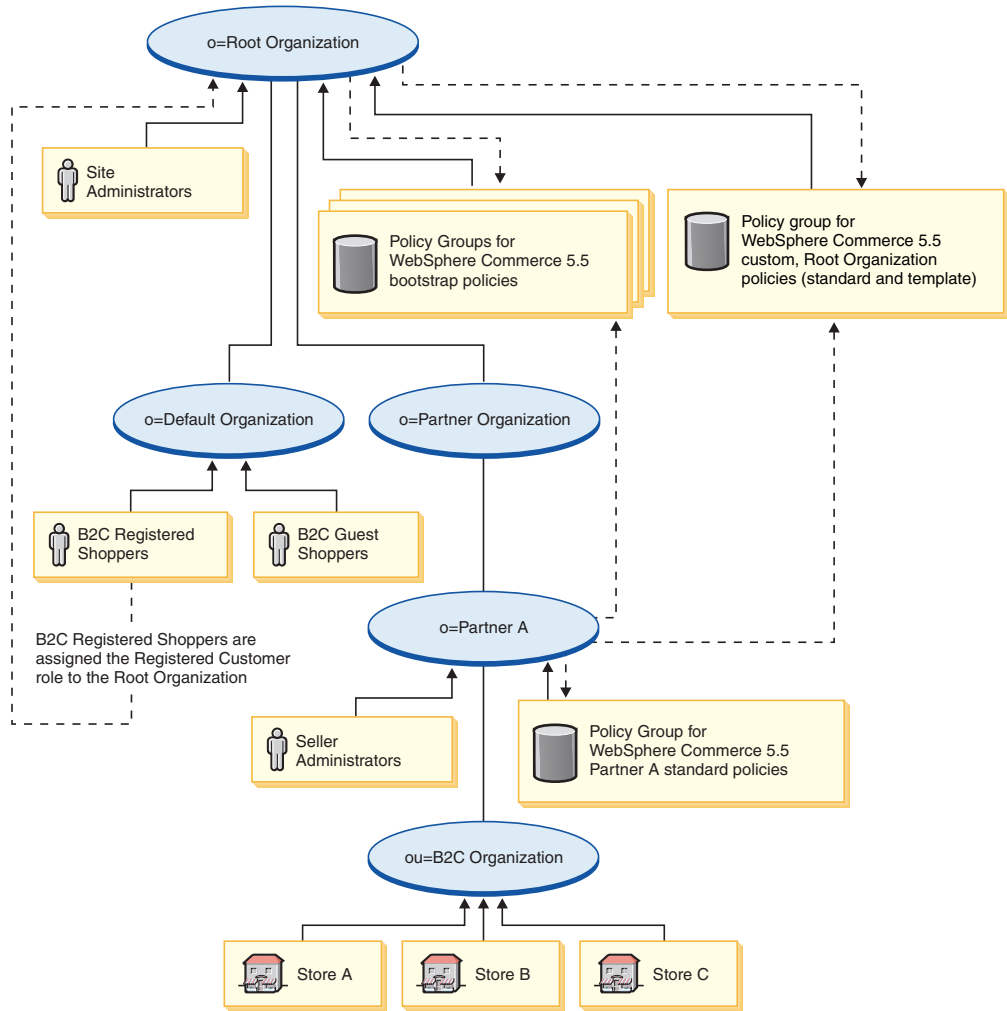
WebSphere Commerce 5.5 conversion (postmigration)

The diagram below depicts the migrated WebSphere Commerce 5.5 site. Note the following items with respect to access control policy setup:

1. All WebSphere Commerce 5.5 bootstrap policies are loaded and placed under the Root Organization.
2. All the WebSphere Commerce 5.4 customized template policies (which were all owned by the Root Organization) are placed in a new policy group. The Root Organization now owns the new policy group and the Root Organization

subscribes to this group. The standard policies are converted from type 0 to type 2 while the template policies are converted from type 1 to type 3.

3. Standard policies on an organization other than the Root Organization are placed in a new policy group.
4. After migration, B2C Registered Shoppers are assigned a "Registered Customer" role in the Root Organization.



For more information on policy groups and subscriptions, refer to the Access Control portion of the *WebSphere Commerce Security Guide*.

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/iSeries/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 5, “Migrating your Commerce instance configuration,” on page 23, Chapter 6, “Before migrating your database,” on page 47 and Chapter 7, “Migrating your WebSphere Commerce 5.4 database,” on page 53.

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 67 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 69 for details on running WCIM for migration.
4. Restore the WebSphere Commerce 5.4 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 72 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 9, “Migrating to Commerce Payments,” on page 63

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
```

Note: The copyDB.sh script is only available if you apply the IBM WebSphere Commerce 5.5.0.2 fix pack. This fix pack is available from the WebSphere Commerce Support site (<http://www.ibm.com/software/commerce/support/>). Click on the link for the edition you are using and see the section **Software downloads**. Download this fix pack, and follow the instructions that are provided to install it.

If you do have not applied this fix pack, you can call the class in QSH using the following command:

```
java -Djava.version=1.3 -classpath  
/QIBM/ProdData/CommerceServer55/properties:/QIBM/ProdData/WebAS5/Base/lib/j2ee.jar:  
/QIBM/ProdData/CommerceServer55/lib/Utilities.jar:  
/QIBM/ProdData/CommerceServer55/lib/jtopen.jar:  
/QIBM/ProdData/WebAS5/Base/lib/xerces.jar
```

```
com.ibm.commerce.migration.tool.CopyDB machine_host database  
new_instance new_instance_pass_word old_instance backuplib  
>> /qibm/copydb_instance_name.log
```

Note that in the above command syntax, the value for classpath is contiguous but is split on separate lines in order to fit the example on a printed page.

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) OWNNOBJOPT(*CHGOWN QPYMSVR) PGPOPT(*NOCHG)
```

6. Migrate the Payments database as described in “Migrating your Payments database” on page 72.

Appendix D. Extending the data migration script

If you have extended the standard WebSphere Commerce 5.4 database schema, you will need to update the supplied data migration script. For example, consider the case if you have added a new column called FRIEND to the USERREG table, as INTEGER DEFAULT 0 and FRIEND has a foreign key to the MBRGRP table.

Before starting the data migration using the supplied migration scripts:

1. Check whether the table USERREG is recreated by the supplied migration scripts. You can go to the directory *WC55_installdir/schema/migration/swim* and search for the string "create table userreg" in the scripts.
2. If you find that one of the supplied scripts tries to recreate the table USERREG, you will need to modify the script to include your modification. If your modified table is not referenced in the supplied scripts, that is, the scripts do not change or update the table, you can skip the rest of this section.

3.

- a. Proceed to the *WC55_installdir/schema/your_platform* directory. Make a copy of the file *wcs.referential.sql*, in the same directory and rename it to *wcs.referential_instance.sql*, then modify it to add a referential constraint:

```
ALTER TABLE USERREG ADD CONSTRAINT F_FRIEND FOREIGN KEY (FRIEND)
REFERENCES MBRGRP ON DELETE CASCADE;
```

Save the file.

- b. Make a copy of the file *WC55_installdir/schema/migration/config/DataMigrationPlan_iSeries.xml* in the same directory and rename it to *DataMigrationPlan_iSeries_instance.xml*. Open this new file and replace the string

```
<command>$MigrationRoot/../../os400/wcs.referential.sql</command>
```

with

```
<command>$MigrationRoot/../../os400/wcs.referential_instance.sql</command>
```

Save the file.

- c. Make a copy of the file *WC_installdir/bin/migratedb54.sh*, and rename it to *migratedb54_instance.sh*. Open this new file, and replace the string *DataMigrationPlan_iSeries.xml*

with

```
DataMigrationPlan_iSeries_instance.xml
```

Save the file.

4. Run the migration script as described in Chapter 7, "Migrating your WebSphere Commerce 5.4 database," on page 53 using *migratedb54_instance.sh*.

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 WebSphere Commerce 5.4 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
```

Data migration script

To migrate the WebSphere Commerce 5.4 database schema to the WebSphere Commerce 5.5 level, you run the database migration tool supplied with WebSphere Commerce 5.5. The database migration tool is a native platform command that invokes a Java program of the same name; it takes the database name, database administrator ID and password, data base type, and an optional log file name as input parameters. It can also use a configuration property file or a data migration plan as input. If no parameters are passed into the tool when it is invoked, the tool will prompt the user to provide necessary information at run time.

It is recommended that before you run the database migration tool, you back up your previous database as described in “Backing up your databases” on page 6.



For details on the database schema differences between WebSphere Commerce 5.4 and WebSphere Commerce 5.5 refer to the WebSphere Commerce - Express 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.

The data migration tool performs the following steps:

1. Collects information required to begin the migration:
 - Migration options that you specify (for example, the database type, or how you wish to handle items such as the master catalog, the default contract, cache triggers, store organization, and so on).
 - Information from your WebSphere Commerce 5.4 system (for example, the backup directory, the WebSphere Commerce instance directory, the migration log directory, and so on.)

The tool shows the premigration conditions and status for the database. It will prompt you to confirm the premigration status.

2. For each WebSphere Commerce component, the tool runs the premigration commands for that component, based on the database version, release level, and database type. For example, it checks the uniqueness of the CPPMNAME column in the CPPMN table because a unique index CPPMNAME+STOREENT_ID is added in WebSphere Commerce 5.5. The tool will fix any items that does not satisfy premigration conditions or it will prompt you to provide more information if required.
3. The tool drops all WebSphere Commerce 5.4 constraints and triggers.
4. The tool updates the WebSphere Commerce 5.4 schema to the WebSphere Commerce 5.5 level.
5. The tool loads new and changed bootstrap data for WebSphere Commerce 5.5 into the database (DELTA only).
6. For each component, the tool runs the data migration command for that component based on the database version and release.
7. For each component, the tool runs a postmigration check of the migrated database and runs the database clean up commands for each WebSphere Commerce component.
8. The tool establishes new WebSphere Commerce 5.5 constraints and triggers.
9. The tool adds new or updated stored procedures required for WebSphere Commerce 5.5.

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 83.

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.

Ensuring column lengths are not exceeded

The lengths of the following columns have changed from WebSphere Commerce 5.4 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	WebSphere Commerce 5.4	WebSphere Commerce 5.5
MBRGRP.DESCRPTION	VARGRAPHIC(4000)	VARGRAPHIC(512)
ORGENITY.DESCRPTION	VARGRAPHIC(4000)	VARGRAPHIC(512)
CONTRACT.NAME	VARGRAPHIC(254)	VARGRAPHIC(200)
CATCONFINF.URL	VARGRAPHIC(4000)	VARGRAPHIC(254)
USRTRAFFIC.PREVURL	VARGRAPHIC(4000)	VARGRAPHIC(254)
USRTRAFFIC.REFURL	VARGRAPHIC(4000)	VARGRAPHIC(254)

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 if they appear in the `migratedb54.log`:

```
[2003.02.04 17:30:06] Info: Executing SQL: DROP INDEX I172145;  
[2003.02.04 17:30:06] Error: I172145 in INSTANCE1 type *FILE not found.  
[2003.02.04 17:28:31] Info: Executing SQL: DROP VIEW wcPRODCatp;  
[2003.02.04 17:28:31] Error: WCPRODCATP in INSTANCE1 type *FILE not found.
```

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 location of the `.txt` files listed below is determined by the content of the `WCALoggerConfig.xml` file whose location is listed on the green screen. If this file exists in the `WC55_userdir/instances/instance_name/xml/loader/ (UserData)` directory, the migration will use the contents of this file to customize the location and contents of these log files. Otherwise, it will use the default file in the `WC55_installdir/xml/loader (ProdData)` directory, which results in all instances sharing the same logs for database migration. If you wish these logs to be instance specific, you must make sure that a customized version of the `WCALoggerConfig.xml` file exists in the `UserData` directory listed above. You will also need the corresponding `wcaLogger.dtd` file in the same directory. If you have previously completed your instance migration, these files will already exist in the `INSTANCE_ROOT/xml/loader` directory.

The log files that are generated for WebSphere Commerce 5.4 databases are:

premigrate_dbname.log

This log file contains all messages generated when the premigration script was run against your WebSphere Commerce 5.4 database. You should ensure there are no errors or warnings flagged in this log.

migratedb_dbname.log

The log file for the overall data migration including populating bootstrap data. 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. This file name does not have a timestamp attached.

messages.txt

Contains the mass loader messages.

trace.txt

Contains the trace file for the Mass Loader and ID Resolver. 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 WebSphere Commerce 5.4 (creation of bootstrap data, create new schema tables, constraints, manipulate business logic related data)

Search for the string, Migration has terminated successfully. Your migratedb.log file should appear similar to the following:

```
[2003.02.04 17:27:01] Info: Logging started in VERBOSE mode.
[2003.02.04 17:27:01] Event: Executing command: preChecking
[2003.02.04 17:27:02] Info: 0 rows of the table account exceeds length limits comments.
...
[2003.02.04 17:27:03] Event: Pre migration ends.
[2003.02.04 17:27:03] Event: Data migration begins...
[2003.02.04 17:27:03] Event: Executing command:
...
[2003.02.04 17:31:52] Event: Data migration ends
[2003.02.04 17:31:52] Event: Post migration begins...
...
[2003.02.04 17:31:52] Info: Executing SQL:
[2003.02.04 17:33:57] Event: Post migration ends.
[2003.02.04 17:33:57] Event: Migration has terminated successfully.
```

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 WebSphere Commerce 5.4 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.

Appendix F. Reducing the size of ACRELGRP.CONDITIONS and ACRESGRP.CONDITIONS

If the ACRELGRP.CONDITIONS and ACRESGRP.CONDITIONS columns exceed 2500 characters, the premigration checker will flag these columns as requiring attention. Since the length of these columns has been reduced in WebSphere Commerce 5.5, you should adjust these lengths before running the data migration script. Detailed examples are provided in the sections below.

If a record in ACRESGRP table exceeds the length, do the following:

1. Extract the contents of the access control tables into XML files by running the `acpextract.sh` script as described in the Access Control portion of the *WebSphere Commerce Security Guide*. For example, from the `WC55_installdir/bin` directory, run:

```
acpextract.sh database_name database_user
             database_user_password ACPoliciesfilter.xml [schema_name]
```

One of the files produced after running the script is:

`WC55_userdir/xml/policies/xml/AccessControlPolicies.xml`. This file contains the language-independent access control policy information; this includes the data stored in the ACRESGRP.CONDITIONS column.

2. Open the `AccessControlPolicies.xml` file and isolate the “<ResourceGroup>” element representing the record that has been flagged by the premigration script. For example, assume you have a file, `acresgrp_conditions_too_big.xml`, that contains only one resource definition:

`RFQResponseDataBeansViewabletoRFQOwnerResourceGroup` as shown below.

Assume that this definition is too large. Note that the ACRESGRP.CONDITIONS data is the information within the “<ResourceCondition>” element.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no" ?>

<!DOCTYPE Policies SYSTEM "../dtd/accesscontrolpolicies.dtd">
<Policies>

  <ResourceGroup Name="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
    OwnerID="RootOrganization">
    <ResourceCondition>
      <![CDATA[
        <profile>
          <andListCondition>
            <orListCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="2"/>
              </simpleCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="4"/>
              </simpleCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="8"/>
              </simpleCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="9"/>
              </simpleCondition>
            </orListCondition>
          </andListCondition>
        </profile>
      ]>
    </ResourceCondition>
  </ResourceGroup>
</Policies>
```

```

        <simpleCondition>
          <variable name="state"/>
          <operator name="="/>
          <value data="10"/>
        </simpleCondition>
        <simpleCondition>
          <variable name="state"/>
          <operator name="="/>
          <value data="11"/>
        </simpleCondition>
      </orListCondition>
      <simpleCondition>
        <variable name="classname"/>
        <operator name="="/>
        <value data="com.ibm.commerce.rfq.beans.RFQResponseDataBean"/>
      </simpleCondition>
    </andListCondition>
  </profile>
]]>
</ResourceCondition>
</ResourceGroup>

<Policy Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
  OwnerID="RootOrganization"
  UserGroup="RFQBuyers"
  ActionGroupName="DisplayDataBeanActionGroup"
  ResourceGroupName="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
  RelationName="owner of RFQ"
  PolicyType="groupableStandard">
</Policy>

<PolicyGroup Name="B2BPolicyGroup" OwnerID="RootOrganization">
  <PolicyGroupPolicy
    Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
    PolicyOwnerId="RootOrganization" />
</PolicyGroup>
</Policies>

```

3. The easiest option to reduce the size is to delete any extraneous data that is defined in the “<ResourceCondition>” element.

However, if everything in the “<ResourceCondition>” element is needed, you will need to split this into two or more separate conditions that would yield the same boolean result with a boolean OR condition. In the following example file, `acresgrp_conditions_fixed.xml`, the resource group from the above step is split into two resource groups:

`RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_1` and `RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2`. Note that the WebSphere Commerce Access Control model is a granting model; that is, if WebSphere Commerce finds at least one policy that grants access, then it provides authorization to the resource.

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no" ?>

<!DOCTYPE Policies SYSTEM "../dtd/accesscontrolpolicies.dtd">
<Policies>
  <ResourceGroup Name="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_1"
    OwnerID="RootOrganization">
    <ResourceCondition>
      <![CDATA[
        <profile>
          <andListCondition>
            <orListCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="2"/>
              </simpleCondition>
              <simpleCondition>
                <variable name="state"/>
                <operator name="="/>
                <value data="4"/>
              </simpleCondition>
            </orListCondition>
          </andListCondition>
        </profile>
      ]]>
    </ResourceCondition>
  </ResourceGroup>

```

```

        <simpleCondition>
            <variable name="state"/>
            <operator name="="/>
            <value data="8"/>
        </simpleCondition>
        <simpleCondition>
            <variable name="classname"/>
            <operator name="="/>
            <value data="com.ibm.commerce.rfq.beans.RFQResponseDataBean"/>
        </simpleCondition>
    </andListCondition>
</profile>
]]>
</ResourceCondition>
</ResourceGroup>

<ResourceGroup Name="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2"
OwnerID="RootOrganization">
<ResourceCondition>
    <![CDATA[
        <profile>
            <andListCondition>
                <orListCondition>
                    <simpleCondition>
                        <variable name="state"/>
                        <operator name="="/>
                        <value data="9"/>
                    </simpleCondition>
                    <simpleCondition>
                        <variable name="state"/>
                        <operator name="="/>
                        <value data="10"/>
                    </simpleCondition>
                    <simpleCondition>
                        <variable name="state"/>
                        <operator name="="/>
                        <value data="11"/>
                    </simpleCondition>
                </orListCondition>
                <simpleCondition>
                    <variable name="classname"/>
                    <operator name="="/>
                    <value data="com.ibm.commerce.rfq.beans.RFQResponseDataBean"/>
                </simpleCondition>
            </andListCondition>
        </profile>
    ]]>
</ResourceCondition>
</ResourceGroup>

<Policy Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
OwnerID="RootOrganization"
UserGroup="RFQBuyers"
ActionGroupName="DisplayDataBeanActionGroup"
ResourceGroupName="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_1"
RelationName="owner of RFQ"
PolicyType="groupableStandard">
</Policy>

<Policy Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2"
OwnerID="RootOrganization"
UserGroup="RFQBuyers"
ActionGroupName="DisplayDataBeanActionGroup"
ResourceGroupName="RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2"
RelationName="owner of RFQ"
PolicyType="groupableStandard">
</Policy>

<PolicyGroup Name="B2BPolicyGroup" OwnerID="RootOrganization">
    <PolicyGroupPolicy
        Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup"
        PolicyOwnerId="RootOrganization" />
    <PolicyGroupPolicy

```

```

        Name="RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2"
        PolicyOwnerId="RootOrganization" />
    </PolicyGroup>
</Policies>

```

4. You must now split the policy that referred to the original resource group into similar policies, such that each one refers to one of the newly created resource groups. The original policy can keep its name, but it must now refer to one of the new split resource groups. You should create a new policy that is similar to the first one, however, it must have a different name and it should refer to another one of the newly created resource groups. You must then assign this new policy to the same policy groups as the original policy in order for the new policy to take effect. You repeat this process until there is a policy for each of the newly created resource groups. In the example, the original policy, RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup maintains the same name, but refers to a new resource group: RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_1. A new policy is created, named RFQBuyersDisplayRFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2, and it refers to the other new resource group: RFQResponseDataBeansViewabletoRFQOwnerResourceGroup_2.
5. Once you split the resource group, associate the groups with different policies and assign the new policies into appropriate policy groups, this data must be loaded back to the database. This can be done by loading the new access control data definition file (acresgrp_conditions_fixed.xml in this example) using the acpload.sh script. For example, from the WC55_installdir/bin directory, run:

```

acpload.sh database_name database_user database_user_password
acresgrp_conditions_fixed.xml [schema_name]

```
6. Steps 2 through 5 should be repeated for other records whose ACRESGRP.CONDITIONS value is flagged as too large.

You can repeat steps 2 through 6 for ACRELGRP.CONDITIONS if it is flagged as being too large by the premigration script. This time, you work with the “<RelationGroup>” element. In the sample file, relationship_group_too_big.xml below, CustomerOrderManagers->RegisteredOrganizationalEntity gets separated into CustomerOrderManagers->RegisteredOrganizationalEntity_1 and CustomerOrderManagers->RegisteredOrganizationalEntity_2 in relationship_group_fixed.xml. The original CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource policy is modified to refer to relationship group CustomerOrderManagers->RegisteredOrganizationalEntity_1, and a new policy, CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource, to refer to the other relationship group: CustomerOrderManagers->RegisteredOrganizationalEntity_2. After the new policy is created, assign this new policy to the same policy groups as the original policy.

Content of relationship_group_too_big.xml:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no" ?>
<!DOCTYPE Policies SYSTEM "../dtd/accesscontrolpolicies.dtd">
<Policies>
    <RelationGroup Name="CustomerOrderManagers->RegisteredOrganizationalEntity"
        OwnerID="RootOrganization">
        <RelationCondition><![CDATA[
            <profile>
                <orListCondition>
                    <openCondition name="RELATIONSHIP_CHAIN">
                        <parameter name="ROLE" value="Customer Service Representative"/>
                        <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>

```

```

        </openCondition>
        <openCondition name="RELATIONSHIP_CHAIN">
          <parameter name="ROLE" value="Seller"/>
          <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
        </openCondition>
        <openCondition name="RELATIONSHIP_CHAIN">
          <parameter name="ROLE" value="Operations Manager"/>
          <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
        </openCondition>
        <openCondition name="RELATIONSHIP_CHAIN">
          <parameter name="ROLE" value="Customer Service Supervisor"/>
          <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
        </openCondition>
        <openCondition name="RELATIONSHIP_CHAIN">
          <parameter name="ROLE" value="Sales Manager"/>
          <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
        </openCondition>
      </orListCondition>
    </profile>
  ]]>
</RelationCondition>
</RelationGroup>

<Policy Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource"
  OwnerID="RootOrganization"
  UserGroup="CustomerOrderManagers"
  ActionGroupName="CustomerServiceCustomerWrite"
  ResourceGroupName="UserDataResourceGroup"
  RelationGroupName="CustomerOrderManagers->RegisteredOrganizationalEntity"
  PolicyType="groupableStandard">
</Policy>

<PolicyGroup Name="ManagementAndAdministrationPolicyGroup" OwnerID="RootOrganization">
  <PolicyGroupPolicy
    Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource"
    PolicyOwnerId="RootOrganization" />
</PolicyGroup>
</Policies>

```

Content of relationship_group_fixed.xml:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no" ?>

<!DOCTYPE Policies SYSTEM "../dtd/accesscontrolpolicies.dtd">
<Policies>

  <RelationGroup Name="CustomerOrderManagers->RegisteredOrganizationalEntity_1"
    OwnerID="RootOrganization">
    <RelationCondition><![CDATA[
      <profile>
        <orListCondition>
          <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Customer Service Representative"/>
            <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
          </openCondition>
          <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Seller"/>
            <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
          </openCondition>
          <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Operations Manager"/>
            <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
          </openCondition>
        </orListCondition>
      </profile>
    ]]>
  </RelationCondition>
</RelationGroup>

  <RelationGroup Name="CustomerOrderManagers->RegisteredOrganizationalEntity_2"
    OwnerID="RootOrganization">
    <RelationCondition><![CDATA[
      <profile>
        <orListCondition>
          <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Customer Service Supervisor"/>
            <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
          </openCondition>
          <openCondition name="RELATIONSHIP_CHAIN">
            <parameter name="ROLE" value="Sales Manager"/>

```

```

        <parameter name="RELATIONSHIP" value="RegisteredOrganizationalEntity"/>
    </openCondition>
</orListCondition>
</profile>
]]>
</RelationCondition>
</RelationGroup>

<Policy Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource"
  OwnerID="RootOrganization"
  UserGroup="CustomerOrderManagers"
  ActionGroupName="CustomerServiceCustomerWrite"
  ResourceGroupName="UserDataResourceGroup"
  RelationGroupName="CustomerOrderManagers->RegisteredOrganizationalEntity_1"
  PolicyType="groupableStandard">
</Policy>

<Policy Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource_2"
  OwnerID="RootOrganization"
  UserGroup="CustomerOrderManagers"
  ActionGroupName="CustomerServiceCustomerWrite"
  ResourceGroupName="UserDataResourceGroup"
  RelationGroupName="CustomerOrderManagers->RegisteredOrganizationalEntity_2"
  PolicyType="groupableStandard">
</Policy>

<PolicyGroup Name="ManagementAndAdministrationPolicyGroup" OwnerID="RootOrganization">
  <PolicyGroupPolicy
    Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource"
    PolicyOwnerId="RootOrganization" />
  <PolicyGroupPolicy
    Name="CustomerOrderManagersExecuteCustomerWriteCommandsOnUserResource_2"
    PolicyOwnerId="RootOrganization" />
</PolicyGroup>
</Policies>

```

For more information on manually loading access control policies, see the topic “Managing access control policies” in the WebSphere Commerce - Express Production and Development online help.

Appendix G. 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 WebSphere Commerce 5.4, 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:** 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%2bh99Sz3R7L0jKC1XLIqT6J6DWA%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 WebSphere Commerce 5.4 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 5.0 Information Center (<http://www.ibm.com/software/webservers/appserv/infocenter.html>).

Logging

The following sections describe logging:

- Logging severities

In WebSphere Commerce 5.4, 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 6. Mapping of log severity types

Severity types in WebSphere Commerce 5.4 logging	Severity types in WebSphere Commerce 5.5 logging (using WebSphere Application Server JRas 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 JRas 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 JRas 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 7. Mapping of trace components

WebSphere Commerce 5.4 tracing components	WebSphere Application Server J2EE 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

Table 7. Mapping of trace components (continued)

WebSphere Commerce 5.4 tracing components	WebSphere Application Server JRes extensions trace logger
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:

- DB2
- @server
- IBM
- iSeries
- OS/400
- 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