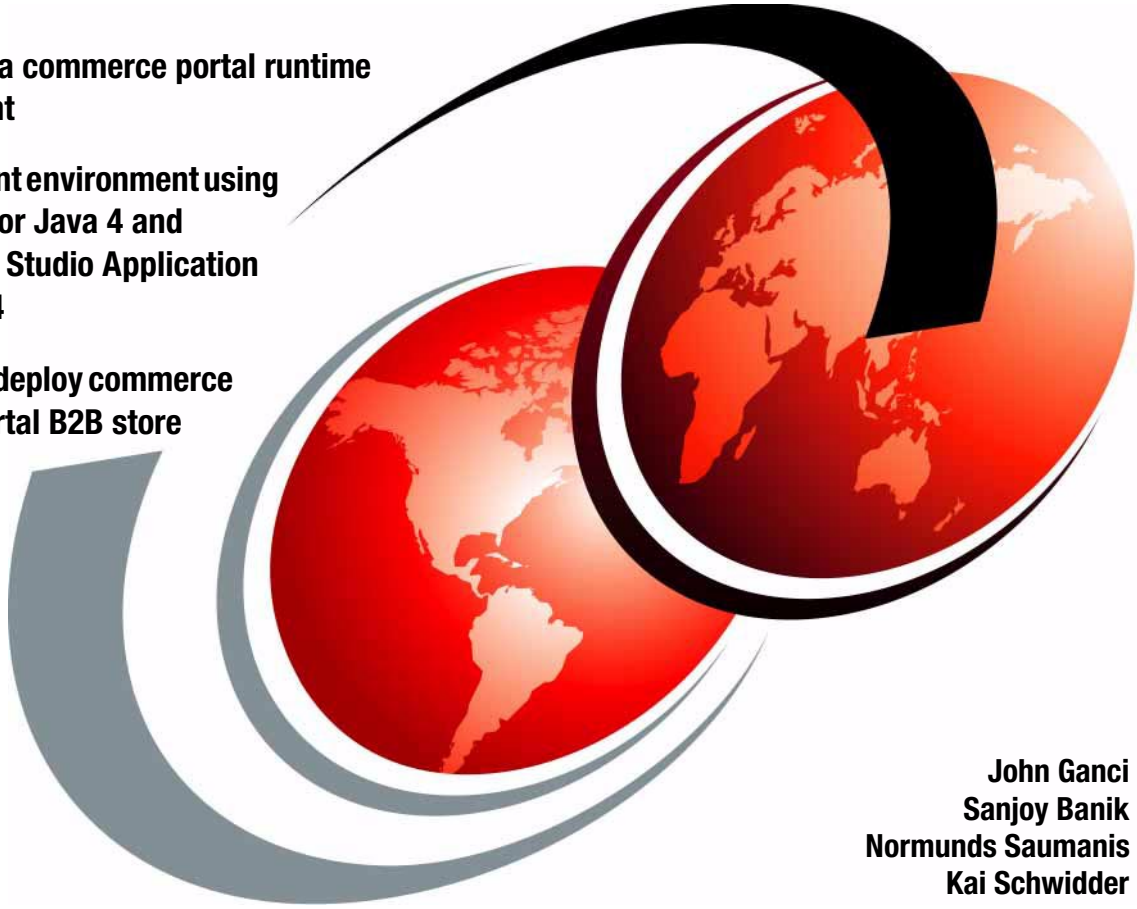
WebSphere software

IBM

Integrating WebSphere Commerce V5.4 and WebSphere Portal V4.1.4

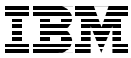
- Implement a commerce portal runtime environment
- Development environment using VisualAge for Java 4 and WebSphere Studio Application Developer 4
- Create and deploy commerce enabled portal B2B store



John Ganci
Sanjoy Banik
Normunds Saumanis
Kai Schwidder

Redpaper

ibm.com/redbooks



International Technical Support Organization

Integrating WebSphere Commerce V5.4 and WebSphere Portal V4.1.4

May 2003

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

First Edition (May 2003)

This edition applies to IBM WebSphere Commerce V5.4 for Windows 2000 and IBM WebSphere Portal Enable V4.1.4 for Windows 2000.

Contents

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.

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.

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.

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.

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.

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

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	IBM®	Redbooks (logo) ™
DB2 Universal Database™	ibm.com®	SecureWay®
DB2®	iSeries™	VisualAge®
@server™	NetVista™	WebSphere®
eServer™	Redbooks™	xSeries®

The following terms are trademarks of International Business Machines Corporation and Rational Software Corporation, in the United States, other countries or both.

ClearCase®	Rational®

The following terms are trademarks of other companies:

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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

Preface

This IBM® Redpaper describes, by example, how to integrate WebSphere® Commerce V5.4 and WebSphere Portal V4.1.4 in a runtime and development environment.

The working example includes multi-tiered runtime implementation procedures, a development environment with source level debug of commerce portlets and JSPs, creation of a commerce enabled portal store using WebSphere Studio Application Developer V4.0.3 and VisualAge® for Java V4, and deployment of the commerce enabled portal store in a runtime or development environment.

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

John Ganci is a Senior Software Engineer, WebSphere Specialist at the IBM ITSO, Raleigh Center. He writes extensively and teaches classes on WebSphere and related topics. John has 14 years of experience in product and application design, development, system testing, and consulting. His areas of expertise include e-commerce, pervasive computing, and Java programming. Before joining the ITSO, he developed e-commerce sites for IBM.

Sanjoy Banik is a Principal Consultant with Gen-i Limited in New Zealand. He has five years of experience in electrical and electronics engineering, including the electro medical sector, and eleven years of experience in I/T fields, including teaching computer science. He holds a master's degree in Computer Science from University of Auckland and a bachelor’s degree in Electrical and Electronics Engineering from Bangladesh Institute of Technology. His areas of expertise include solutions architecture and development, training, mentoring, technical team leadership, e-business, and portal solutions with Java and Microsoft architecture. He has written three redbooks about e-commerce and portals.

Normunds Saumanis is an I/T Architect in IBM Global Services, Latvia. He has over 10 years experience in systems support, systems integration, application development and I/T consulting. He holds a degree in Computer Science from Michigan State University. His areas of expertise include AIX®/UNIX systems support, I/T infrastructure design and operations, systems integration, Java, pervasive and Web applications, and IBM WebSphere.

Kai Schwidder is an I/T Architect in IBM Software Group, Switzerland. He has 13 years of experience in the field of consulting, application development, and systems integration for e-business and e-commerce solutions. He holds a degree in Computer Science from the Technical University Berlin, Germany. His areas of expertise include systems integration, application architecture and development, business to technology consulting, technical team leadership, WebSphere Commerce, WebSphere Portal and WebSphere MQ.

Thanks to the following people for their contributions to this project:

Special thanks to **Joanna Ng** of IBM Canada, Architect and Manager of WebSphere Commerce Portal Development.

Vincent Watt, IBM Canada
John Wenzel, IBM US
Chris Mann, IBM US
Terry Hudson, IBM UK
Norm Smith, IBM US
Gail Christensen, IBM US Raleigh ITSO Center
Linda Robinson, IBM US Raleigh ITSO Center

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this Redpaper or other Redbooks™ in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HZ8 Building 662
  P.O. Box 12195
  Research Triangle Park, NC 27709-2195

1

Introduction

This Redpaper describes, by example, how to integrate WebSphere Commerce V5.4 and WebSphere Portal V4.1.4 in a runtime and development environment.

In January 2003, the ITSO released a draft of the redbook *WebSphere Commerce Portal V5.4 Solutions*, SG24-6890. That redbook was based on WebSphere Commerce V5.4.0.3, IBM Commerce Enhancement Pack - October 2002 Edition, and WebSphere Portal V4.1.4 Enable. Prior to the publishing of that redbook, a decision was made to refresh the content of the redbook for WebSphere Portal V4.2 Enable and add support for WebSphere Commerce and WebSphere Studio Application Developer V5 support.

The primary reason for producing this Redpaper is to capture the valuable content created for integrating WebSphere Commerce V5.4.0.3 and WebSphere Portal V4.1.4 runtime and development environment, which includes the VisualAge for Java V4 for WebSphere Commerce development and test environment and WebSphere Studio Application Developer V4.0.3 for portlet development.

Note: For information on the WebSphere Commerce Portal architecture or integration with WebSphere Portal V4.2, refer to the redbook *WebSphere Commerce Portal V5.4 Solutions*, SG24-6890.

1.1 Structure of this Redpaper

This Redpaper is organized as follows:

- Chapter 2, “Implement the runtime environment” on page 7

  This chapter provides step-by-step procedures and best practices for installing and configuring a multi-tiered commerce enabled portal runtime environment. You will find detailed information about implementing WebSphere Commerce V5.4, WebSphere Commerce FixPak V5.4.0.3, IBM Commerce Enhancement Pack - October 2002 Edition, IBM SecureWay Directory V3.2.2, and WebSphere Portal V4.1.4.

- Chapter 3, “Implement the development environment” on page 93

  This chapter provides detailed instructions for implementing a source level debug environment for WebSphere Commerce commands, EJBs, JSPs (commerce portlet JSPs) within IBM VisualAge for Java V4, and commerce portlet development with WebSphere Studio Application Developer V4.0.3 and WebSphere Application Server V4.0.3, Single Server Edition.

- Chapter 4, “Create a commerce enabled portal store” on page 183

  This chapter describes how to create a commerce enable portal store from the ITSO provided sample code.

- Chapter 5, “Deploy a commerce enabled portal store” on page 207

  This chapter provides procedures for deploying the commerce enabled store assets, including commerce portlet JSPs to the commerce runtime and commerce portlets to the portal runtime in both a production runtime environment and development unit test environment.

- Appendix A, “WebSphere Commerce and WebSphere Portal sample LDIF files” on page 241

  This appendix includes sample LDAP LDIF files for WebSphere Commerce and WebSphere Portal used to implement the runtime environment.

- Appendix B, “Debug logon for commerce portlets without single sign-on” on page 245

  This appendix describes how the ITSO provided code can be used to provide a simulated single sign-on between the WebSphere Portal (commerce portlet) and VisualAge for Java WebSphere Test Environment where the commerce server and store (commerce portlet JSPs) are running. This functionality is needed for source level debug between commerce portlets and commerce portlet JSPs.

- Appendix C, “Tips and troubleshooting for commerce enabled portals” on page 253

  This appendix describes tips and troubleshooting commerce enabled portals.

- Appendix D, “Additional material” on page 261

  This appendix describes where and how to download the ITSO developed sample code zip file for this Redpaper.

1.2 Where to find information

This section describes where to find information (product guides, redbooks, Redpapers, links) about WebSphere Commerce, WebSphere Commerce Portal, and WebSphere Portal.

The information found in this Redpaper is intended for IT architects, specialists and developers who need to integrate the WebSphere Commerce and WebSphere Portal. The WebSphere Commerce Portal architecture is built upon the technologies of WebSphere Commerce and WebSphere Portal. For this reason, it is important that the reader have a firm understanding of the WebSphere Commerce and WebSphere Portal architecture. We have listed key sources of information for WebSphere Commerce, WebSphere Commerce Portal, and WebSphere Portal.

WebSphere Commerce information

There are several sources information that can be found for IBM WebSphere Commerce V5.4 relevant to commerce enabled portal architecture, design and implementation:

- The following IBM WebSphere Commerce V5.4 product guides and online information are included with the product or can be downloaded at:

  Business Edition:

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

  Professional Edition:

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

  – *Fundamentals Guide, IBM WebSphere Commerce V5.4*

    This product guide provides and overview of the WebSphere Commerce architecture and features.

  – *Programmer’s Guide, IBM WebSphere Commerce V5.4*

    This product guides provides programming reference information for developing WebSphere Commerce store-front and back-end assets.

Included in this guide is a description of the WebSphere Commerce programming model and architecture.

- *Store Developer's Guide, IBM WebSphere Commerce V5.4*

  This product guide describes the architecture of the store data, database schema and objects. The guide provides detailed information on how to manage such data assets as products, categories, taxes and shipping.

- *WebSphere Commerce V5.4 online documentation*

- Depending on the desired operating system platform, select the appropriate WebSphere Commerce installation guide. For example, on Windows:

  - *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
  - *Additional Software Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
  - *Installation Guide, IBM WebSphere Commerce FixPak V5.4.0.3*

- *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*

  This guide provides some basic procedures for installing and configuring the WebSphere Commerce development environment.

- IBM Redbooks

  - *WebSphere Commerce V5.4 Handbook*, SG24-6567

    This redbook includes architecture guidelines for security, scalability, performance tuning and testing. In addition, the redbook includes many advanced working example implementation and integration scenarios.

  - *Mobile Commerce Solutions Guide, using WebSphere Commerce Suite V5.1*, SG24-6171

    This redbook provides detailed architecture information and working examples for the WebSphere Commerce mobile architecture, which is used in the integration of WebSphere Commerce and WebSphere Portal.

  - *WebSphere Commerce V5.4 Catalog Design and Content Management*, SG24-6585

    This redbook describes the store and data architecture, as well as provides a working example for managing data using Catalog Manager.

  - *B2B e-commerce With WebSphere Commerce Business Edition V5.4, Patterns for e-business Series*, SG24-6194

This redbook includes Business patterns, Composite Patterns, Application patterns, Runtime patterns and Product mappings based on the IBM Patterns for e-business.

WebSphere Commerce Portal information

To find more information on WebSphere Commerce Portal, refer to the following:

- *Getting Started, IBM Commerce Enhancement Pack*

  This guide provides procedures for installing and configuring the IBM Commerce Enhancement Pack - October 2002 Edition, which WebSphere Commerce Portal is a part of.

- *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*

  We used this guide for reference information in Chapter 2, “Implement the runtime environment” on page 7, which provides a procedure for installing and configuring an advanced commerce enabled portal runtime environment.

WebSphere Portal information

For WebSphere Portal development and the portlet API information, refer the following:

- *Portlet Developers Guide, WebSphere Portal V4.1*
- *WebSphere Portal V4.1 Developers Handbook*, SG24-6897
- Redbook *WebSphere Portal V4.1 Handbook Volume 2*, SG24-6920
- *Mobile Applications with IBM WebSphere Everyplace Access Design and Development*, SG24-6259

For WebSphere Portal runtime environment information, refer to the following:

- Redbook *WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883
- Redbook *WebSphere Portal V4.1 Handbook Volume 3*, SG24-6921
- *WebSphere Portal V4.1, Windows 2000 Installation,* REDP3593
- *Access Integration Pattern using IBM WebSphere Portal Server*, SG24-6267
- *A Portal composite pattern Using WebSphere Portal V4.1.2*, SG24-6869

2

Implement the runtime environment

This chapter describes how to implement a multi-tiered WebSphere Commerce and WebSphere Portal runtime environment used to deploy the ITSO B2B CEP store working example. The procedures documented include the WebSphere Commerce FixPak V5.4.0.3, IBM Commerce Enhancement Pack - October 2002 Edition, and many other FixPak levels — beyond the WebSphere Commerce generally available release — work-arounds and best practices.

This chapter is organized into the following sections:

- Planning for implementation and deployment
- WebSphere Commerce node implementation
- Directory Server (LDAP) node implementation
- WebSphere Portal node implementation
- Enable single sign-on between WebSphere Portal and WebSphere Commerce
- Deploy the ITSO B2B CEP store
- DB2 Server node implementation
- Remote Web server node implementation

2.1 Planning for implementation and deployment

Prior to implementing the runtime environment, it is important to prepare in advance of the implementation. For example, you will need to obtain static TCP/IP addresses and hostnames in advance, because these are not easily changed once everything is configured.

This section describes the key considerations for planning and implementation such as the network configuration, hardware and software used in the runtime environment, and the high level process we used to implement the runtime environment. The runtime environment implementation is part of the deployment process.

This section includes the following topics:

- Network environment
- Hardware used within the ITSO test environment
- Software used within the ITSO test environment
- VMWare or Ghost
- High level implementation procedure

2.1.1 Network environment

The ITSO runtime environment is depicted in Figure 2-1 on page 9 with a focus on the numbered nodes. The nodes displayed in Figure 2-1 and listed in Table 2-1 are installed on the Windows 2000 Server platform. The focus of this Redpaper is about the integration of WebSphere Commerce and WebSphere Portal.

Table 2-1 Nodes and hostnames for the ITSO working example runtime environment

Diagram #	Node name	Host name
1	WebSphere Commerce node	wcserv1
2	Directory Server node	wcldap1
3	WebSphere Portal node	wcportal1
4	DB2® Server node	wcdb2
5	Web Server node	wcweb1

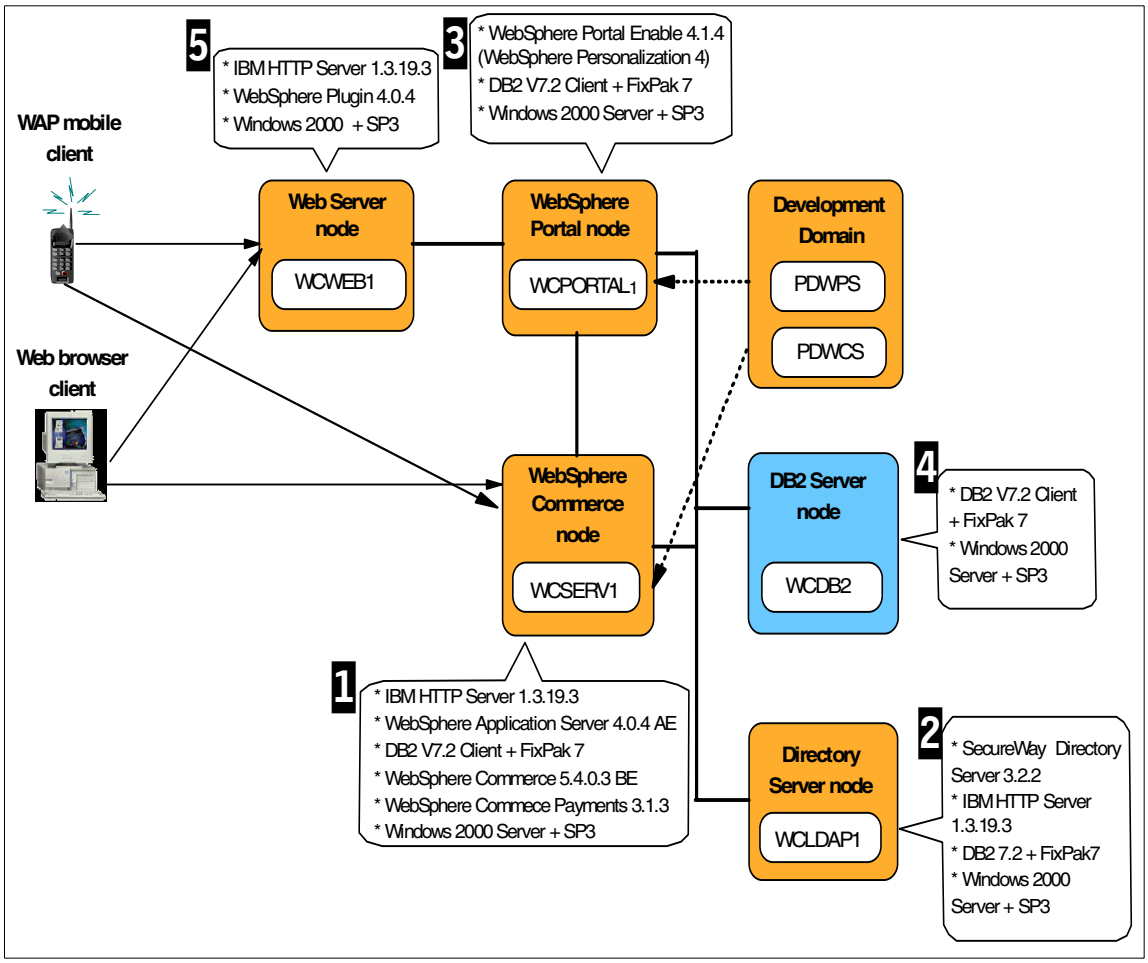


Figure 2-1 ITSO runtime environment and nodes implemented for the working example

2.1.2 Hardware used within the ITSO test environment

We recommend that you refer to the product documentation for the official guidelines and prerequisite hardware requirements. Within the ITSO runtime environment, we used the following hardware for the stated nodes.

1 WebSphere Commerce node

We used the following hardware for the WebSphere Commerce node:

- IBM eServer™ xSeries® 230 (8658-61Y)
  - 1 CPU, 1 GHz Intel PIII
  - 1 GB RAM

- 18 GB Hard Disk
- 1 IBM Ethernet (NetFinity Fault Tolerance PCI Adapter)

2 Directory Server node

We used the following hardware for the Directory Server node:

- IBM NetFinity 3000 (8476-41U)
  - 1 CPU, 450 MHz Intel PII
  - 512 MB RAM
  - 4 GB Hard Disk
  - 1 IBM Ethernet Adapter

3 WebSphere Portal node

We used the following hardware for the WebSphere Portal node:

- IBM eServer xSeries 230 (8658-61Y)
  - 1 CPU, 1 GHz Intel PIII
  - 1 GB RAM
  - 18 GB Hard Disk
  - 1 IBM Ethernet (NetFinity Fault Tolerance PCI Adapter)

4 DB2 Server node

We used the following hardware for the DB2 Server node:

- IBM eServer xSeries 230 (8658-61Y)
  - 1 CPU, 1 GHz Intel PIII
  - 1 GB RAM
  - 18 GB Hard Disk
  - 1 IBM Ethernet (NetFinity Fault Tolerance PCI Adapter)

5 Web Server node

We used the following hardware for the Web Server node for the WebSphere Portal:

- IBM NetFinity 3000 (8476-41U)
  - 1 CPU, 450 MHz Intel PII
  - 512 MB RAM
  - 4 GB Hard Disk
  - 1 IBM Ethernet Adapter

2.1.3 Software used within the ITSO test environment

Within the ITSO runtime environment, we used FixPak levels beyond the generally available release level of the required software components. We used

the FixPak levels to address software issues and used the latest service levels available at the time of writing this Redpaper.

1 WebSphere Commerce node

Table 2-2 lists the software levels used on the WebSphere Commerce node within the ITSO runtime environment.

Table 2-2 Software levels used for the WebSphere Commerce node

Software	Original Level	FixPak level
Microsoft Windows 2000 Server	na	Service Pack 3
IBM DB2 UDB V7.2, Enterprise Edition	7.1.0.55 FixPak 5	7.1.0.68 FixPak 7
IBM HTTP Server	1.3.19	1.3.19.3 (WAS FP4)
IBM WebSphere Application Server V4.0.4, Advanced Edition	4.0.1	4.0.4 FixPak 4 + e-Fixes
IBM WebSphere Commerce V5.4, Business Edition	5.4	5.4.0.3 (FixPak 5.4.0.3)
IBM WebSphere Payment Manager (renamed to IBM WebSphere Commerce Payments)	3.1.2	3.1.3
WebSphere Commerce V5.4 Enhancement Pack	October 2002 Edition	na

2 Directory Server node

Table 2-3 lists the software levels used on the Directory Server node within the ITSO runtime environment.

Table 2-3 Software levels used for the Directory Server node

Software	Original Level	FixPak level
Microsoft Windows 2000 Server	na	Service Pack 3
IBM DB2 UDB V7.2, Enterprise Edition	7.1.0.55 FixPak 5	7.1.0.68 FixPak 7
IBM HTTP Server	1.3.19	1.3.19.3 FixPak 1.3.19.3
IBM SecureWay® Directory	3.2.2	3.2.2 + e-Fix 2

3 WebSphere Portal node

Table 2-4 lists the software levels used on the WebSphere Portal node within the ITSO runtime environment.

Table 2-4 Software levels used for this Redpaper on the WebSphere Portal node

Software	Original Level	FixPak level
Microsoft Windows 2000 Server	na	Service Pack 3
IBM DB2 UDB V7.2, Client only	7.1.0.68 FixPak 7	na
IBM HTTP Server **Note**: This Web server will be installed on a separate node. We installed this on this node for ease of configuration of the httpd.conf. There server will then be disabled.	1.3.19.3	na
IBM WebSphere Application Server V4.0.4, Advanced Edition	na	4.0.4 + e-Fixes PQ66355
IBM WebSphere Personalization V4.0.1	4.0.1	na
IBM WebSphere Portal Enable V4.1.4	4.1.4	na
WebSphere Commerce V5.4 Enhancement Pack (commerce portlets)	October 2002 Edition	na

4 DB2 Server node

Table 2-5 lists the software levels used on the DB2 Server node within the ITSO runtime environment.

Table 2-5 Software levels used on the DB2 Server node

Software	Original Level	FixPak level
Microsoft Windows 2000 Server	na	Service Pack 3
IBM DB2 UDB V7.2, Enterprise Edition	7.1.0.55 FixPak 5	7.1.0.68 FixPak 7

5 Web Server node

Table 2-6 lists the software levels used on the remote Web Server node for the WebSphere Portal within the ITSO runtime environment.

Table 2-6 Software levels used for the Web Server node

Software	Original Level	FixPak level
Microsoft Windows 2000 Server	na	Service Pack 3
IBM HTTP Server	1.3.19	1.3.19.3 WAS FixPak 4
IBM WebSphere Application Server V4, Advanced Edition, plugin	4.0.1	4.0.4 (FixPak 4)

2.1.4 VMWare or Ghost

While developing the implementation procedure, we found it very useful to use VMWare 3.2 and Ghost 6.5 to take a snapshot of the installation by creating an image of the system. There are other utilities like this on the market. Each utility has advantages.

A VMWare system image is very nice in that it is portable to different systems. You can store multiple versions of virtual machines on the same system and easily start them (limited by disk space). When using VMWare you do sacrifice a bit on performance. We used VMWare 3.2 and the Microsoft Sysprep utility to change the Windows SID. We found VMWare to be especially useful and amazingly compatible and reliable (truly excellent software). For more information on VMWare, refer to the following URL:

`http://www.vmware.com/`

During the installation of the nodes in the ITSO runtime environment, we created zip files (backups) for key stages of the implementation.

Ghost allows for the imaging of systems, but is much more limited in moving images to a system of different hardware specs (device drivers). The big advantage of ghost is that when done, you are running the native operating system on the hardware and have better performance. We used the Ghost multicast feature to load systems of the network and created Microsoft DOS based network client boot diskettes to connect to a Windows 2000 share where the Ghost image was stored. This can be used to capture (dump) or load from image. For more information on Symantec Ghost, refer to the following URL:

`http://www.ghost.com/`

Using these utilities allowed us to go back to a previous state during the installation (provided an image was captured of the system). This can save a tremendous amount of time and allow you to verify your knowledge of the environment before a proceeding to deployment in a production environment.

2.1.5 High level implementation procedure

The high level runtime environment implementation procedure is as follows:

1. WebSphere Commerce node implementation
2. Directory Server (LDAP) node implementation
3. WebSphere Portal node implementation
4. Enable single sign-on between WebSphere Portal and WebSphere Commerce
5. Deploy the ITSO B2B CEP store
6. DB2 Server node implementation
7. Remote Web server node implementation

We chose the order listed to avoid problems with software dependencies. Also, we wanted the procedure to be flexible for users that want to install the components on the same node. In some cases, the installation procedures more smoothly when following the order listed. For example, the IBM Commerce Enhancement Pack - October 2002 Edition installation is designed for an environment where the database and application server are on the same node.

After the WebSphere Commerce and WebSphere Portal nodes are working, we will backup and restore databases (WebSphere Application Server, WebSphere Commerce, WebSphere Commerce Payments, and WebSphere Portal) to the database server, and configure the DB2 client/server connection.

2.2 WebSphere Commerce node implementation

This section describes the high level steps to install the WebSphere Commerce node within the ITSO test environment. Table 2-2 on page 11 lists the software levels used on the WebSphere Commerce node within the ITSO runtime environment.

The procedure outlined includes best practices and FixPaks beyond the original release of WebSphere Commerce V5.4.

Note: More detailed information can be found in the redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567 or the product installation guides.

The high level installation and configuration procedures for the WebSphere Commerce node are as follows:

1. Windows 2000 Server installation
2. DB2 Server installation
3. IBM HTTP Server installation
4. WebSphere Application Server installation
5. WebSphere Commerce installation
6. WebSphere Commerce Payments installation
7. Commerce Enhancement Pack installation
8. WebSphere Commerce instance creation
9. Commerce Enhancement Pack post-install configuration
10. WebSphere Commerce administration tools verification
11. WebSphere Commerce Payments configuration
12. Database backup
13. Enable WebSphere Commerce portal adapter
14. Disable the WebSphere Commerce cache

2.2.1 Windows 2000 Server installation

In preparation for the installation of WebSphere Commerce and supporting components, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.
2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Create a token object
   - Increase quotas
   - Log on as a service
   - Replace a process level token
3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

2.2.2 DB2 Server installation

This section highlights the key steps for installing and configuring the DB2 Server for use with WebSphere Application Server, WebSphere Commerce, WebSphere Commerce Payments. In our example, we will configure the Commerce Application Server node first with a local database server.

The DB2 Server installation is organized as follows:

- Install DB2 UDB V7.2 Enterprise Edition
- Install DB2 V7 FixPak 7 (7.1.0.68)
- Update JDBC level to JDBC2

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- Product Guide *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*

Install DB2 UDB V7.2 Enterprise Edition

To install the DB2 UDB V7.2, Enterprise Edition server on the WebSphere Commerce node refer to, “Install DB2 UDB V7.2 Enterprise Edition” on page 87.

Install DB2 V7 FixPak 7 (7.1.0.68)

To install DB2 V7.2 FixPak 7 refer to, “Install DB2 V7 FixPak 7 (7.1.0.68)” on page 87.

Update JDBC level to JDBC2

To update the JDBC level to JDBC 2 as required by WebSphere Commerce refer to, “Update JDBC level to JDBC2” on page 88.

2.2.3 IBM HTTP Server installation

This section highlights the key steps for installing and configuring the IBM HTTP Server for use with WebSphere Application Server and WebSphere Commerce.

The IBM HTTP Server installation is organized as follows:

- Install the IBM HTTP Server
- Configure the IBM HTTP Server
- Verify the IBM HTTP Server

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*

Install the IBM HTTP Server

To install the IBM HTTP Server, do the following:

1. Insert the WebSphere Application Server CD, change to the httpd directory and run Setup.
2. We accepted the default options unless noted:
   - We installed the IBM HTTP Server to the `c:\ibm\http` directory.

Configure the IBM HTTP Server

After the IBM HTTP Server installation is complete, we need to configure the IBM HTTP Server by completing the following steps:

1. Create admin user.

   ```
   <HTTP_HOME>\htpasswd -c conf/admin.passwd <http_admin_user>
   ```

2. Enable httpd.conf for SSL.
3. Create IBM HTTP Server key database.
4. Create certificate for the IBM HTTP Server.

Verify the IBM HTTP Server

After the IBM HTTP Server has been configured, verify that the Web server is accessible as follows:

1. Restart the IBM HTTP Server from Windows services.
2. Enter the following URLs in a Web browsers:

   ```
   http://<hostname>
   https://<hostname>
   ```

Note: As part of the procedure for installing the WebSphere Application Server, we will install WebSphere Application Server V4 FixPak 4. In our example, the IBM HTTP Server and WebSphere Application Server are installed on the same node. FixPak 4 includes services fixes for the IBM HTTP Server. After installing FixPak 4, the IBM HTTP Server version is 1.3.19.3.

2.2.4 WebSphere Application Server installation

This section highlights the key steps for installing and configuring the WebSphere Application Server for use with WebSphere Commerce.

The section is organized as follows:

- Create the WebSphere Application Server repository database
- Install WebSphere Application Server V4.0.1
- Install WebSphere Application Server V4 FixPak 4 (V4.0.4)

- Install WebSphere Application Server V4.0.4 e-Fixes
- Configure the WebSphere Application Server
- Verify the WebSphere Application Server

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
- WebSphere Application Server V4 FixPak 4 (V4.0.4) Readme file
- WebSphere Application Server V4.0.4 E-fix Readme files

Create the WebSphere Application Server repository database

Prior to starting the WebSphere Application Server Administrative server, the WebSphere repository database needs to be created. To create the WebSphere Application Server repository database, complete the following steps:

1. Start a DB2 Command window.
2. Create the WebSphere repository database as follows:

   Syntax:

   ```
   db2 create db <db_name>
   ```

   Example:

   ```
   db2 create db was4
   ```

3. Update the database applheapsz window as follows:

   ```
   db2 update db cfg for was4 using applheapsz 1024
   ```

   Note: The value suggested for the applheapsz in our example is 1024. This is needed by the WebSphere Portal and WebSphere Personalization in the event that you add the WebSphere Application Server Administrative Server node to the same WebSphere Domain as this node.

4. Catalog the TCP/IP node:

   ```
   db2 catalog tcpip node <node_name> remote <remote_hostname> server
   <service_name>
   ```

5. Catalog the database:

   ```
   db2 catalog db was4 as was40
   ```

6. Connect to the WebSphere database as follows:

   Syntax:

   ```
   db2 connect to db was40 using <db2admin_user> using <password>
   ```

7. Verify the connection to the database:

   ```
   db2 connect to was40 user <db2admin_user> using <db2admin_password>
   ```

8. List all databases as follows:

   ```
   db2 list db directory
   ```

Install WebSphere Application Server V4.0.1

To install the WebSphere Application Server V4.0.1, Advanced Edition, do the following:

1. Insert the WebSphere Application Server CD, and run Setup.
2. We accepted the default options unless noted:
   - Select **Custom Installation**.

     Note: In our example, we manually installed and configured the IBM HTTP Server prior to installing the WebSphere Application Server.
   - We installed the WebSphere Application Server to the c:\ibm\was directory.
3. After the installation, rename the `<WAS_HOME>\bin\createwasdb.scr` to `createwasdb.scr.bak`. In our example, we have already manually created the WebSphere Application Server repository database, which is the function of this script.

Install WebSphere Application Server V4 FixPak 4 (V4.0.4)

To install WebSphere Application Server V4 FixPak 4 (V4.0.4), do the following:

1. Ensure you have uninstalled E-fixes before installing FixPak 4.

   Note: Refer to the WC5403_Readme.pdf for details on how to determine if you have E-fixes installed, and how to remove them.

2. Ensure the following Windows services have been stopped:
   - IBM HTTP Server
   - IBM WS AdminServer 4.0
3. Download the WebSphere Application Server V4 FixPak 4 (V4.0.4) at:

   `http://www.ibm.com/software/webservers/appserv/support.html`
4. Install the WebSphere Application Server V4 FixPak 4.

   ```
   <temp_dir>\install
   ```

5. Review the FixPak log file to check for errors.

   ```
   <WAS_HOME>\logs\was40_AE_ptf_4.log
   ```

Install WebSphere Application Server V4.0.4 e-Fixes

After the installation of the WebSphere Application Server V4 FixPak 4, install the required e-Fixes for WebSphere Commerce as follows:

1. Enter the following URL to access the WebSphere Application Server support page for e-Fixes:

   http://www.ibm.com/software/webservers/appserv/support.html

2. Click **All Fixes, FixPaks and Tools**.
3. Select **4.0.4** from the search criteria list and then click **Submit**.
4. From the e-Fixes listing, download the following:
   - Cumulative security e-Fix (includes PQ63116 to resolve Admin server stop/start authentication errors)
   - Cumulative WebSphere plugin e-Fix
5. Set the Java PATH by entering the following in a command window:

   ```
   SET PATH=%PATH%;c:\ibm\was\java\bin
   ```

   Where c:\ibm\was is the directory where the WebSphere Application Server has been installed.

6. Install the e-Fix as follows for each e-Fix listed above:

   ```
   java -jar <e-Fix> -target <WAS_HOME>
   ```

7. Verify the e-Fix has been installed:
   - Verify a backup of the e-Fix has been created:
     `<WAS_HOME>\e-fix\<efix#>`
   - Verify the product.xml file has been updated:
     `<WAS_HOME>\properties\com\ibm\websphere\product.xml`
8. Ensure that you have restarted your system after installation.

Configure the WebSphere Application Server

To configure the WebSphere Application Server, do the following:

1. Start the WebSphere Administrative Server (IBM WS AdminServer 4.0 Windows service).
2. Verify the WebSphere Administrative Server has started by review the `<WAS_HOME>\logs\tracefile`.
3. Add host aliases to the default virtual host from the WebSphere Administrative Console.
   - `<hostname>:80`
   - `<hostname>:443`
   - `<fully_qualified_hostname>:80`
   - `<fully_qualified_hostname>:443`

4. Regenerate the Web server plugin.
5. To make the plugin-cfg.xml file updates take effect immediately, restart the IBM HTTP Server and restart the Default Application server.

Verify the WebSphere Application Server

After the WebSphere Application Server, verify the configuration as follows:

1. Ensure the following Windows services have been started:
   - DB2
   - IBM HTTP Server
   - IBM WS Admin Server 4.0
2. From the WebSphere Administrative Console, start the Default Application Server.
3. From a Web browser, enter the following URLs:

```
http://<hostname>/servlet/snoop
https://<hostname>/servlet/snoop
http://<hostname>/webapp/examples/showCfg
https://<hostname>/webapp/examples/showCfg
```

2.2.5 WebSphere Commerce installation

This section highlights the key steps for installing and configuring WebSphere Commerce V5.4, Business Edition and WebSphere Commerce FixPak V5.4.0.3.

The section is organized as follows:

- Install WebSphere Commerce V5.4
- Install WebSphere Commerce FixPak V5.4.0.3
- Post-install WebSphere Commerce FixPak V5.4.0.3

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
- *Installation Guide, IBM WebSphere Commerce FixPak V5.4.0.3* (WC5403_Readme.pdf)

Install WebSphere Commerce V5.4

To install WebSphere Commerce V5.4, Business Edition, do the following:

1. Insert the WebSphere Commerce V5.4 CD1 and run Setup.
2. We accepted the default options unless noted as follows:

– When selecting components that will be running on this node (WebSphere Commerce node), in a single-tier configuration, select **The commerce server, a Web server, and a database**.

> **Note:** This is not asking what components to install, but what components will be running on the server. In our example, we have installed the prerequisite components with a higher FixPak level prior to installing WebSphere Commerce.

– We installed WebSphere Commerce in the `c:\ibm\wc` directory.

> **Note:** Shortly after inserting WebSphere Commerce CD2, the installer will attempt to install WebSphere Application Server V4 e-Fixes. These e-fixes are out of date and will not be installed since the level of WebSphere Application Server is V4.0.4.

Install WebSphere Commerce FixPak V5.4.0.3

To install the WebSphere Commerce FixPak V5.4.0.3, do the following:

1. Stop the following Windows services:
   – IBM HTTP Server
   – IBM WS AdminServer 4.0
   – IBM WC Configuration Manager Server
2. Ensure that you have installed the prerequisite WebSphere Application Server V4 FixPak 4 and e-Fixes (refer to 2.2.4, “WebSphere Application Server installation” on page 17).
3. Download the FixPak from the following URL:

   `http://www.ibm.com/support/docview.wss?rs=497&uid=swg24001839`
4. Unzip the WebSphere Commerce FixPak V5.4.0.3 zip file and enter the following from a command prompt to start the FixPak installer:

   ```
   install_wc
   ```
5. Follow the FixPak installation prompts.
6. Review the log file for errors.

   ```
   <WC_HOME>\service\fixpak\5403\WCfixpack5403_wc_WIN.log
   ```

Post-install WebSphere Commerce FixPak V5.4.0.3

After the WebSphere Commerce FixPak V5.4.0.3 has been installed, there are several configurations steps that need to be completed. In our example, we are using WebSphere Application Server V4.0.4 and e-Fixes. Refer to the following technote for running WebSphere Commerce V5.4.0.3 on WebSphere Application

Server V4.0.3 (applies to V4.0.4). This section describes how to address the issues related to the MassLoad utility documented in the technote.

http://www.ibm.com/support/docview.wss?rs=494&uid=swg21054765

1. Add JITC_COMPILEOPT environment variable as follows to either the Windows system environment or adminserver.bat
   - Windows system environment
     - Variable: `JITC_COMPILEOPT`
     - Value:
       `JITC_COMPILEOPT=SKIP{com/ibm/wca/MassLoader/Formatter/JDBCFormatter$F`

   Or,

   - adminserver.bat (all on one line)
     ```
     SET
     JITC_COMPILEOPT=SKIP{com/ibm/wca/MassLoader/Formatter/JDBCFormatter$Form
     atWorker}{run}
     ```

   Note: In our case, we updated both the Windows environment variable and the adminserver.bat. We start the WebSphere Application Server Administrative Server using the adminserver.bat from this point forward. We updated the Windows environment as well to avoid problems running the MassLoader from the command line (Java utility to load XML data).

   Note: Stopping the WebSphere Administrative Server

   To stop the WebSphere Administrative Server in a controlled manner, especially when starting using adminserver.bat, we recommend using the WebSphere Control Program (wscp). For example:

   1. Change to the <WAS_HOME>\bin directory
   2. Enter `wscp` at the command line to start wscp.
   3. Enter to following command to stop the WebSphere Administrative Server from the wscp command line.

      ```
      Node stop /Node:wcserv1/
      ```

      Where `wcserv1` is the node name.

4. Horizontal cloning.

   Note: Do not use this work-around if using the IBM Commerce Enhancement Pack - October 2002 Edition (see note box below).

   When configuring horizontal cloning with WebSphere Application Server V4, the enterprise application name and enterprise application directory in

<WAS_HOME>\InstalledApps need to be the same. To work around a problem with the default <WC_HOME>\xml\config\was.deployed.EJB.xml, it has been recommended that the was.deployed.EJB.xml file be modified as follows:

```
<enterprise-application action="create" name="WC Enterprise App
$DRIVER_INSTANCE_NAME$"> )
```

Note: At the time of writing this Redpaper, performing the above stated work-around for horizontal cloning causes the ImportEJB.bat (CMRedeploy) for the IBM Commerce Enhancement Pack - October 2002 Edition to fail in subsequent steps. It appears that the enterprise application name “WebSphere Commerce Enterprise Application” is hardcoded in CMRedeploy. Performing the horizontal cloning work-around renames the enterprise application name to “WC Enterprise App” to match the directory. At the time of writing this Redpaper, a work-around to this problem was not known.

Do not use the horizontal cloning work-around if using the IBM Commerce Enhancement Pack - October 2002 Edition.

We advise the reader to check the WebSphere Commerce support page for the most current information on work-arounds or e-Fixes:

http://www-3.ibm.com/software/webservers/commerce/wc_be/support.html

5. Update the <WC_HOME>\bin\setenv.bat (Loader property file not found work-around) to include the following:

```
set JAVA_EXE=%JAVA_HOME%\bin\java -Xbootclasspath/p:c:\ibm\wc\lib\jlog.jar
```

Note: In order for the WebSphere Commerce <instance> application server to recognize this change, the application server will need to be restarted. In our example, we have not yet created the WebSphere Commerce <instance> application server.

6. Update WebSphere Commerce Staging Server.

   If you are using the WebSphere Commerce Staging Server feature, and have an existing DB2 WebSphere Commerce instance database, you will need to manually update the database by running the FixPak version of the wcs.stage.trigger.sql script found in the <WC_HOME>\schema\db2 directory.

   For example, enter the following from a DB2 command windows:

```
db2 connect to <wc_db> user db2admin using <password>
db2 -tvf wcs.stage.trigger.sql
```

Note: If you have not created a WebSphere Commerce instance or database yet, it is not necessary to update the database. The FixPak installer has updated the wcs.stage.trigger.sql file and it will be used when the WebSphere Commerce instance and database are created.

7. Copy search.xml to WebSphere Commerce instance XML directory.

   If you have created a WebSphere Commerce instance before the WebSphere Commerce FixPak V5.4.0.3 installation, you will need to manually copy the search.xml file now found in the `<WC_HOME>\instances\default\xml` directory to the `<WC_HOME>\instances\<instance_name>\xml` directory for each instance your have created.

Note: Once the WebSphere Commerce FixPak V5.4.0.3 is installed, the default instance directory contains the search.xml file and all newly created WebSphere Commerce instances will contain this file.

2.2.6 WebSphere Commerce Payments installation

This section highlights the key steps for installing WebSphere Commerce Payments. After the release of WebSphere Commerce V5.4, WebSphere Commerce Payments FixPak 3.1.3 was made available for download. The FixPak contains fixes and changes the name of the product to WebSphere Commerce Payments.

The section is organized as follows:

- Pre-requisites for WebSphere Payment Manager V3.1.2
- Install WebSphere Payment Manager V3.1.2
- Install WebSphere Commerce Payments FixPak V3.1.3

Note: For detailed installation instructions, refer to the following:

- *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
- *Installation Guide, IBM WebSphere Payment Manager for Multiplatforms*

Pre-requisites for WebSphere Payment Manager V3.1.2

Prior to installing WebSphere Payment Manager V3.1.2, do the following:

1. Start a DB2 command window.
2. Create payman database.

```
db2 create db pay1
```

3. Update applheapsz.

```
db2 update db cfg for pay1 using applheapsz 256
```

4. Catalog the WebSphere Commerce Payments database.

```
db2 catalog db pay1 as payman
```

> **Note:** When using the catalog database DB2 command the <node_name> parameter is optional if the database is local.
>
> If you use the <node_name> parameter when performing the catalog database operation, ensure you use the proper case of the <node_name> as known within DB2. If you are not sure, view the node name from the DB2 Control Center. If the database is local, using the <node_name> parameter is optional.
>
> Where <node_name> is the TCP/IP node cataloged in the configuration of the WebSphere repository database, "Create the WebSphere Application Server repository database" on page 18.

5. Verify the connection to the database:

```
db2 connect to payman user <db2admin_user> using <db2admin_password>
```

6. Ensure the following are started:
   - DB2
   - WebSphere Administrative Server

     WebSphere Administrative Server is started depending on how you implemented the environment variable for JITC_COMPILEOPT:

     - adminserver.bat

     Or,

     - IBM WS AdminServer 4.0 (Windows service)

Install WebSphere Payment Manager V3.1.2

To install WebSphere Payment Manager V3.1.2, do the following:

1. Insert the WebSphere Payment Manager V3.1.2 CD and run Install.
2. We accepted the default options, unless noted below:
   - We installed to the c:\ibm\wcpay directory.
   - Database: payman

Install WebSphere Commerce Payments FixPak V3.1.3

To install the WebSphere Commerce Payments FixPak V3.1.3, do the following:

1. Download the FixPak from the following URL:

   https://www6.software.ibm.com/dl/paymgr/srvupdts-p

2. Ensure the WebSphere Application Server Administrative Server is started (adminserver.bat or IBM WS AdminServer 4.0 Windows service).
3. Ensure the WebSphere Payment Manager application server is stopped (use the WebSphere Administrative Console or command to stop the application server).
4. Make sure the directory containing the ptf.class is in the `CLASSPATH` environment variable. For example:

   ```
   set CLASSPATH=%CLASSPATH%;.
   ```

5. Set the PATH environment variable for Java. For example:

   ```
   set PATH=%PATH%;c:\ibm\was\java\bin
   ```

6. To start the PTF installer, enter the following command with no extension:

   ```
   java <ptf_filename>
   ```

   Where <ptf_filename> is the PTF filename without the.class extension.

> **Note:** The PTF class file must be in a directory that is NOT where Payments is installed and that has write access.

2.2.7 Commerce Enhancement Pack installation

This section highlights the key steps for installing IBM Commerce Enhancement Pack - October 2002 Edition.

> **Note:** For detailed installation instructions, refer to the *Getting Started, IBM Commerce Enhancement Pack* guide included with the IBM Commerce Enhancement Pack - October 2002 Edition.

1. Backup the WebSphere Commerce instance database if one exits at this stage from a previous instance being created.
2. Download the Windows version of the IBM Commerce Enhancement Pack to a temporary folder from the following URL:

   http://www.ibm.com/software/webservers/commerce/epacks/v54/

3. Navigate to the temporary folder and unzip the IBM Commerce Enhancement Pack - October 2002 Edition. For example, we unpacked the zip file to c:\temp\cep. We will refer to the unzip directory as <CEP_HOME>.
4. Start the installation by executing **Setup**.
5. Follow the prompts to install and configure the IBM Commerce Enhancement Pack.

   Note: When running the WebSphere Commerce IC Checker to ensure that your WebSphere Commerce machine is configured properly all error messages can be safely ignored.

6. Check the following log file for errors to ensure that the IBM Commerce Enhancement Pack has installed properly:

   ```
   <WC_HOME>\logs\EPInstall.log
   ```

2.2.8 WebSphere Commerce instance creation

This section highlights the key steps for creating a WebSphere Commerce instance using the Configuration Manager.

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*

To create a WebSphere Commerce instance, do the following:

1. Ensure the following Windows services are started:
   - DB2
   - IBM WC Configuration Manager
   - IBM WS AdminServer 4.0

Note: If you chose to update the adminserver.bat with the JITC_COMPILEOPT environment variable, you will need to temporarily use the a Window service instead of adminserver.bat.

The Configuration Manager does not recognize that the WebSphere Administrative Server has been started when launched by adminserver.bat. The Configuration Manager will not allow you to create an instance if it does not believe that the WebSphere Administrative Server is started, which is needed to deployed the WebSphere Commerce enterprise application.

2. Ensure the IBM HTTP Server Windows service is *stopped*.

   This is done as a work-around to avoid an instance creation failure when the IBM HTTP Server is configured with SSL support prior to creating an instance.

3. Create WebSphere Commerce instance using the Configuration Manager which can be launch by clicking **Start -> Programs -> WebSphere Commerce -> Configuration**.

4. The default logon ID and password are webadmin/webibm. You will be prompted to change the password.

5. We accepted the default options unless otherwise noted.

   – Instance name: `wc1`
   – Database name: `wc1db`

   Note: Staging Server

   If you are planning on using this node as a staging server and using the WebSphere Commerce staging utilities, ensure the **Use Staging Server** checkbox is check within the Configuration Manager Instance Creation Wizard on the Database tab.

   If this is not a staging server or you are not planning on using the WebSphere Commerce staging utilities, do not check Use Staging Server.

6. Verify the instance creation.

   Review the log files found in the `<WC_HOME>\instances\<instance>\logs` directory.

7. Rebind the data sources after the instance creation.

   After the instance has been created, it may be necessary to run the following command to rebind the data sources (work-around for blank page for tools):

```
<WAS_HOME>\bin\wscp -f rebindDataSources.tcl
```

2.2.9 Commerce Enhancement Pack post-install configuration

This section highlights the key steps for post-install configuration of the IBM Commerce Enhancement Pack - October 2002 Edition

Note: For details, refer to the *Getting Started, IBM Commerce Enhancement Pack* guide included with the IBM Commerce Enhancement Pack - October 2002 Edition.

In our example, we created an instance after the Enhancement Pack was installed. For this reason, we need to complete some steps in this section.

1. Update the WebSphere Commerce instance database.
   a. Execute the `EP1_DBUpdate.db2.bat` script found in the `<WC_HOME>\EnhancementPack\bin` directory as follows from a DB command window:

      ```
      EP1_DBUpdate.db2 <wc_dbname> <dbuser> <dbpassword>
      ```

   b. To verify the DB2 script worked properly, review the `<WC_HOME>\logs\EPDBupdate.log` file.

   Note: As documented in the *Getting Started, IBM Commerce Enhancement Pack* guide, several errors will be listed in the EPDBUpdate.log for cases where the EP1_DBUpdate.db2.bat is run on an instance created after the IBM Commerce Enhancement Pack - October 2002 Edition is installed.

2. Ensure that the WebSphere Administrative Server is started (adminserver.bat), and that the WebSphere Commerce - <instance> application server is stopped.
3. We recommend that you create a backup of the WebSphere Commerce Enterprise Application prior to importing the new EJBs in the next step.
   a. Start the WebSphere Application Server Administration Console.
   b. Select and expand the **WebSphere Application Domain** -> **Enterprise Applications**.
   c. Select **WebSphere Commerce Enterprise Application - <instance_name>**, right-click and select **Export Application**.

d. Follow the on screen instructions, export the application to another location for backup. We created a directory c:\ibm\wc\wcentapp.bak and exported the enterprise application to this directory.

4. Import new EJBs.

   a. If you are using adminserver.bat to start the WebSphere Administrative Server, you will temporarily need to stop the server. The ImportEJB.bat does not recognize that the WebSphere Administrative Server is running when started using adminserver.bat.

      ```
      <WAS_HOME>\bin\wscp
      Node stop /Node:wcserv1/
      ```

   b. Start the IBM WS AdminServer 4.0 Windows service.

   c. Run the following command from a Windows command prompt:

      ```
      <WC_HOME>\EnhancementPack\bin\ImportEJB -instanceName <instance_name>
      <database_type>
      ```

      Where *<instance_name>* is the name of your WebSphere Commerce instance, and *<database_type>* is either DB2 or Oracle.

      For example:

      ```
      <WC_HOME>\EnhancementPack\bin\ImportEJB -instanceName wc1 DB2
      ```

   d. Verify the ImportEJB.bat worked properly by reviewing the `<WC_HOME>\logs\CMRedeploy_EJB.log` for errors.

5. If you redeploy EJBs into the WebSphere Commerce Enterprise Application, you may have to follow the instructions in technote 1066544 found at:

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

   Note: We did not need to perform this step for the configuration and procedure we have documented.

6. WebSphere Commerce instance database update for DB2.

   If an instance is created by WebSphere Commerce FixPak V5.4.0.3 and the instance uses DB2, then the instance's database is not configured properly. This misconfiguration causes a failure during publishing a store.

   a. Start a DB2 command window.

   b. Connect to the WebSphere Commerce instance database.

      ```
      db2 connect to <wc_database>
      ```

   c. To address this issue, start a DB2 command window, and enter the SQL statements listed in Example 2-1 on page 32.

Where <wc_database> is the WebSphere Commerce instance database name.

Example 2-1 WebSphere Commerce instance database update needed for DB2

```
db2 update db cfg for <wc_database> using applheapsz 16384
db2 update db cfg for <wc_database> using stmtheap 60000
db2 update db cfg for <wc_database> using app_ctl_heap_sz 8192
db2 update db cfg for <wc_database> using locklist 400
db2 update db cfg for <wc_database> using indexrec RESTART
db2 update db cfg for <wc_database> using logfilsiz 1000
db2 update db cfg for <wc_database> using logprimary 12
db2 update db cfg for <wc_database> using logsecond 10
```

d. To verify the database configuration updates, you can run the following command in a DB2 command window:

```
db2 get db cfg for <wc_database> | grep -i <parameter>
```

Where <wc_database> is the name of the instance database.

Where <parameter> is one of the following parameters: applheapsz, stmtheap, app_ctl_heap_sz, locklist, and so on.

For example:

```
db2 get db cfg for mall | grep -i applheapsz
```

Verify if the applheapsz value is 16384.

Note: If you do not have a grep utility, you may consider piping the output to a text file that can be searched in a text editor. For example:

```
db2 get db cfg for wc1db >c:\temp\dbcfg.out
```

7. Fashion Flow sample store (optional).

For the ITSO working example, we did not use the Fashion Flow sample store, therefore, this section is optional. If you plan on using the Fashion Flow sample you will need to perform the following steps:

a. Follow the instructions in technote number 1053492, 1054760 and 1066581. Technotes are accessible from the following URL:

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

b. Install JavaMail 1.2 for e-mail activities as described in the post-installation steps of the *Getting Started, IBM Commerce Enhancement Pack* guide included with the IBM Commerce Enhancement Pack - October 2002 Edition.

8. Stop the IBM WS AdminServer 4.0 Windows service, and start the WebSphere Administrative Server using the updated adminserver.bat.

2.2.10 WebSphere Commerce administration tools verification

This section describes how to verify that the WebSphere Commerce administration tools are working properly.

To verify the WebSphere Commerce administration tools, do the following:

1. Ensure the following are started:
   - DB2 Server Windows services
   - IBM HTTP Server Windows service
   - WebSphere Administrative Server (adminserver.bat)
2. From the WebSphere Administration Console, start the WebSphere Commerce <instance> application server.
3. Verify WebSphere Commerce Store Services:
   a. Start Store Services as follows:

      `https://<hostname>:8000/storeservices`

      or

      From Windows, **Start -> Programs -> IBM WebSphere Commerce -> Store Services**.

   b. Verify that you can logon.
      - User ID: `wcsadmin`
      - Password: `wcsadmin` (default)

      At this stage, we want to make sure that the tool is able to start without errors.
4. Verify WebSphere Commerce Administration Console:
   a. Start the WebSphere Commerce Administration Console as follows:

      `https://<hostname>:8000/adminconsole`

      or

      From Windows, **Start -> Programs -> IBM WebSphere Commerce -> Administration Console**.

   b. Verify that you can logon.
      - User ID: `wcsadmin`
      - Password: `<password>`
5. Verify WebSphere Commerce Accelerator:
   a. Start the WebSphere Commerce Accelerator as follows:

      `https://<hostname>:8000/accelerator`

      Or,

From Windows, **Start -> Programs -> IBM WebSphere Commerce -> WebSphere Commerce Accelerator**.

b. Verify that you can logon.

- User ID: `wcsadmin`
- Password: `<password>`

2.2.11 WebSphere Commerce Payments configuration

This section describes the necessary steps to configure the WebSphere Commerce Payments server.

If WebSphere Commerce Payments is started when the store is published, the payment server is configured during the store publishing process. If not, start WebSphere Commerce Payments and republish the store or manually configure WebSphere Commerce Payments.

Start WebSphere Commerce Payments

To start WebSphere Commerce Payments, do the following:

1. Start the following Windows services:
   - DB2
   - IBM HTTP Server
   - WebSphere Administrative Server (adminserver.bat)
2. Start the WebSphere Commerce Payments application server from the WebSphere Administrative Console.
3. Start the WebSphere Commerce Payments Server for the `<WCPAY_HOME>` directory from the command line as follows:

   ```
   IBMPayServer
   ```

4. When prompted, enter the database password for the payman database.

Verify the WebSphere Commerce Payments Admin Console

To verify the WebSphere Commerce Payments Administration Console, do the following:

1. Start the WebSphere Commerce Payments Administration Console as follows:
   - `http://<hostname>/webapp/PaymentManager`

   Or,

   - From Windows, **Start -> Programs -> WebSphere Commerce Payments -> Commerce Payments Logon**.

c. Verify that you can logon.

- User ID: `wcsadmin`
- Password: `<password>`

2. The WebSphere Commerce Payments Administration Console can be used to setup additional methods of credit card payment (Visa, American Express), add payment cassettes (SET, Cyber Cash, etc), create accounts, approve orders, etc.

2.2.12 Database backup

Before publishing a store, we recommend that you backup the WebSphere Commerce instance database. For details, refer to Appendix , "Backing up a DB2 database" on page 259.

2.2.13 Enable WebSphere Commerce portal adapter

To enable the WebSphere Commerce portal adapter, do the following:

Note: For more information, refer to the *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*.

1. Stop the WebSphere Commerce <instance> application server from the WebSphere Application Server Administration Console.
2. During the IBM Commerce Enhancement Pack - October 2002 Edition installation, the `<WC_HOME>\schema\db2\wcs.portal.sql` script is executed by the EP1_DBUpdate.db2.script. This sql script updates the WebSphere Commerce instance database tables PVCDEVMDL, PVCDEVSPEC, and PVCMDLSPEC for the Portal and WAP Portal adapter.

   By default wcs.portal.sql script sets the deviceFormatId to the next available entry number. You must ensure the MODEL_ID in the PVCDEVMDL table updated by the wcs.portal.sql script matches the deviceFormatId in the instance XML file (Example 2-2 on page 36).

   Ensure the wcs.portal.sql file has been executed. When installing the IBM Commerce Enhancement Pack - October 2002 Edition this sql file is executed by the EP1_DBUpdate.db2 script. If you have executed the EP1_DBUpdate.db2.script as documented in the installation procedure, this step has already been completed.
3. Determine the MODEL_ID in the PVCDEVMDL table for the adapter (PORTAL and WAP PORTAL). For example, to query the database for the PORTAL adapter, enter the following from a DB2 command window:

   ```
   db2 connect to <wc_dbname> user <db2_admin> using <password>
   ```

```
db2 select * from pvcdevmdl
```

Record the value of the `MODEL_ID` for the desired adapter.

> **Note:** You may consider piping the output to a text file for recording purposes. For example:
>
> ```
> db2 select * from pvcdevmdl>c:\temp\model_id.out
> ```

4. Change to the directory of the WebSphere Commerce <instance>.xml file:

   ```
   <WC_HOME>\instances\<instance_name>\xml
   ```

5. Backup the original <instance>.xml file to <instance>_org.xml.
6. Update the WebSphere Commerce <instance>.xml file found in the `<WC_HOME>\instances\<instance>\xml` directory with a text editor to enable the WebSphere Commerce Portal adapter.
   a. Update deviceFormatId.
      i. Search deviceFormatId within the HttpAdapter <tag> for PORTAL in the <instance>.xml file.
      ii. Update the value of the deviceFormatId to match the MODEL_ID recorded for the adapter.
   b. Enable the adapter.

      Enable the deployment descriptor of the WebSphere Commerce portal adapter by changing the `enabled="false"` value to `enabled="true"` in the <instance>.xml file as seen in Example 2-2.
   c. Save the <instance>.xml file.

Example 2-2 Enable the WebSphere Commerce Portal adapter

```
<HttpAdapter deviceFormatId="-3"
   deviceFormatType="PVCAdapter"
      deviceFormatTypeId="-1"
   enable="true"
   factoryClassname="com.ibm.commerce.portaladapters.HttpPortalBrowserAdapter"
   name="PORTAL">
<PVCAdapter bufferTimeout="5"preferredLogonTimeout="20"registrationMode="0"/>
</HttpAdapter>
```

When complete with the changes to the <instance>.xml file, save the file and restart the WebSphere Commerce <instance> application server.

2.2.14 Disable the WebSphere Commerce cache

When using the Portal adapter and the WAP Portal adapter, the JSP directory includes JSPs with the same name as the standard HTTP adapter clients. The WebSphere Commerce cache can not differentiate the JSPs that have been cached for the different devices (adapters). This may cause a problem when rendering the JSPs for the requesting device type. For this reason, the WebSphere Commerce cache must be disabled when more than one of the following adapters is used:

- HTTP adapter
- Portal adapter
- WAP Portal adapter

To disable the WebSphere Commerce cache, do the following:

1. Start the WC Configuration Manager Server Windows service.
2. Start the WebSphere Commerce Configuration Manager.
3. Log on to the Configuration Manager.
4. From the left pane, select and expand **WebSphere Commerce -> <node> -> Caching Subsystem**.
5. Uncheck the **Enable Cache** checkbox and click **Apply**.
6. Select **Components -> CacheDaemon**.
7. Uncheck the **Enable Component** checkbox and click **Apply**.
8. Exit Configuration Manager.
9. Restart the WebSphere Commerce <instance> application server.

You have now completed the base configuration of the WebSphere Commerce node. If you are using VMWare or Ghost (test purposes), this is an appropriate time to create an image.

2.3 Directory Server (LDAP) node implementation

The IBM SecureWay Directory V3.2.2 server is used by the WebSphere Portal Server for authentication and user lookups and is used by the WebSphere Commerce node as a user registry and for authentication. When configuring both nodes for single sign-on, the directory is shared amongst the participating applications and users only have to logon one time. In our scenario, the users will authenticate from the WebSphere Portal, which is configured to use IBM SecureWay Directory, and a LtpaToken will be passed to the single sign-on participating application, WebSphere Commerce.

This section describes the high level steps to install the IBM Directory Server node within the ITSO test environment. In this example, the IBM Directory Server is not dependent on other nodes. The DB2 Server is installed on the IBM Directory Server node. Since authentication will be routed through the IBM Directory Server, we did not want to have additional network traffic to retrieve each user authentication from a remote database server.

Note: More detailed information can be found in the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*
- Redbook *IBM WebSphere Portal V4.1 Handbook*, SG24-6883

This high level implementation steps for the IBM Directory Server are as follows:

1. Windows 2000 Server installation
2. DB2 Server installation
3. IBM HTTP Server installation
4. IBM SecureWay Directory installation
5. IBM SecureWay Directory e-Fix installation
6. IBM SecureWay Directory configuration
7. IBM SecureWay Directory verification
8. Import data for WebSphere Portal and WebSphere Commerce
9. Configure WebSphere Commerce for LDAP
10. Enable SSL between WebSphere Commerce and LDAP

2.3.1 Windows 2000 Server installation

In preparation for the installation of DB2, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.
2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Create a token object
   - Increase quotas
   - Log on as a service
   - Replace a process level token

3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

2.3.2 DB2 Server installation

For details on installing and configuring the DB2 Server, refer to 2.7, “DB2 Server node implementation” on page 86. The following tasks need to be completed:

- Install DB2 UDB V7.2 Enterprise Edition.
- Install DB2 V7 FixPak 7 (7.1.0.68).
- Update JDBC level to JDBC2.

2.3.3 IBM HTTP Server installation

For details on installing and configuring the IBM HTTP Server, refer to 2.2.3, “IBM HTTP Server installation” on page 16. The following tasks need to be completed:

- Install the IBM HTTP Server.
- Configure the IBM HTTP Server.
- Verify the IBM HTTP Server.

2.3.4 IBM SecureWay Directory installation

WebSphere Portal V4.1.4 and commerce enabled portals provided in the IBM Commerce Enhancement Pack - October 2002 Edition, require IBM SecureWay Directory V3.2.2 for a single sign-on configuration with WebSphere Commerce.

IBM SecureWay Directory V3.2.2 is included with the WebSphere Portal V4.1.4 Enable offering.

Note: For detailed information on installing and configuring the IBM SecureWay Directory V3.2.2, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- Redpaper *WebSphere Portal V4.1, Windows 2000 Installation,* REDP3593
- Product Guide *IBM SecureWay Directory V3.2.2 Installation Guide*

To install the IBM SecureWay Directory V3.2.2, do the following:

1. Ensure the IBM HTTP Server is stopped.

2. IBM SecureWay Directory V3.2.2 is included with the WebSphere Portal V4.1.4 Enable offering. It can also be downloaded from the following URL:

   http://www.ibm.com/software/network/directory/downloads/

3. To install IBM SecureWay Directory, insert the WebSphere Portal V4.1.4 - SecureWay Directory for Windows CD, run **Setup** from the \swd\win\ldap32_us directory.

   Note: The installation directory may vary depending on the distribution source of IBM SecureWay Directory.

4. We accepted the default options unless noted.
5. When the Installed Applications window appears, the setup program will detect which software has already been installed, click **Next**.

   In our example, we have already install the following software components:

   – GSKIT (GSKit 5 is included with the IBM HTTP Server V1.3.19.3)
   – DB2 (DB2 server installation)
   – Web Server (IBM HTTP Server installation)

6. When the Select Components window appears, click **Custom**.
7. When the Choose Destination Location window appears, we entered `c:\ibm\ldap` for the installation directory and then clicked **Next**.
8. When the Custom Installation window appears, we checked the following options and then clicked **Next**.

   – Client 3.2.2
   – Server 3.2.2
   – Client SDK
   – DMT and Java

9. When the Select Program Folder window appears, we accepted the default and clicked **Next**.
10. When the Configure window appears, deselect the following options and then click **Next**:

    – Set the directory administrator name and password.
    – Create the directory DB2 database.
      • Configure a Web server.

Note: We will configure the following in 2.3.6, “IBM SecureWay Directory configuration” on page 41. The window seen during the installation, can be accessed at any time to configure the IBM SecureWay Directory node.

The configuration can be performed during the install or post install. In either scenario, refer to 2.3.6, “IBM SecureWay Directory configuration” on page 41 for more detailed information.

11. When the Start copying files for IBM SecureWay Directory and Client SDK window appears, click **Next** to begin copying files.
12. After the installation is complete, review the Readme, select **Yes, I want to restart my computer now**, and the click **Finish**.

2.3.5 IBM SecureWay Directory e-Fix installation

After the installation of IBM SecureWay Directory V3.2.2, an e-Fix is required for WebSphere Portal V4.1.4. The IBM SecureWay Directory e-Fix is included with WebSphere Portal V4.1.4 on the Setup Manager CD.

Note: For detailed information on installing the IBM SecureWay Directory V3.2.2 e-Fix, refer to *e-Fix 3.2.2 - SWD-002 README for SecureWay Directory 3.2.2 Windows-based Server Installations*. This guide can be found at swd-eFix\win\win128-2.pdf of the WebSphere Portal V4.1.4 - Setup Manager CD.

Attention: After installing the e-Fix, make sure you have verified the files that have been copied and restart the system where SecureWay Directory is installed to avoid a problem with an older dll still being in memory.

2.3.6 IBM SecureWay Directory configuration

After the installation, it is necessary to configure the directory server. The following sections cover the initial setup of the directory server using the IBM SecureWay Directory V3.2.2 configuration utility:

- Create the directory administrator user.
- Create the directory database.
- Configure the Web server for directory administration.

Note: This configuration can be performed during the installation process or after installation, as is the case in our example. It is assumed that none of the three options has been configured. If any of the three steps has been set up during the installation process, do not check the option when prompted.

1. Ensure the IBM HTTP Server is stopped.
2. Launch IBM SecureWay Directory Configuration by clicking **Programs -> IBM SecureWay Directory -> Directory Configuration**.
3. Select the three available options as seen in Figure 2-2 and then click **Next**.

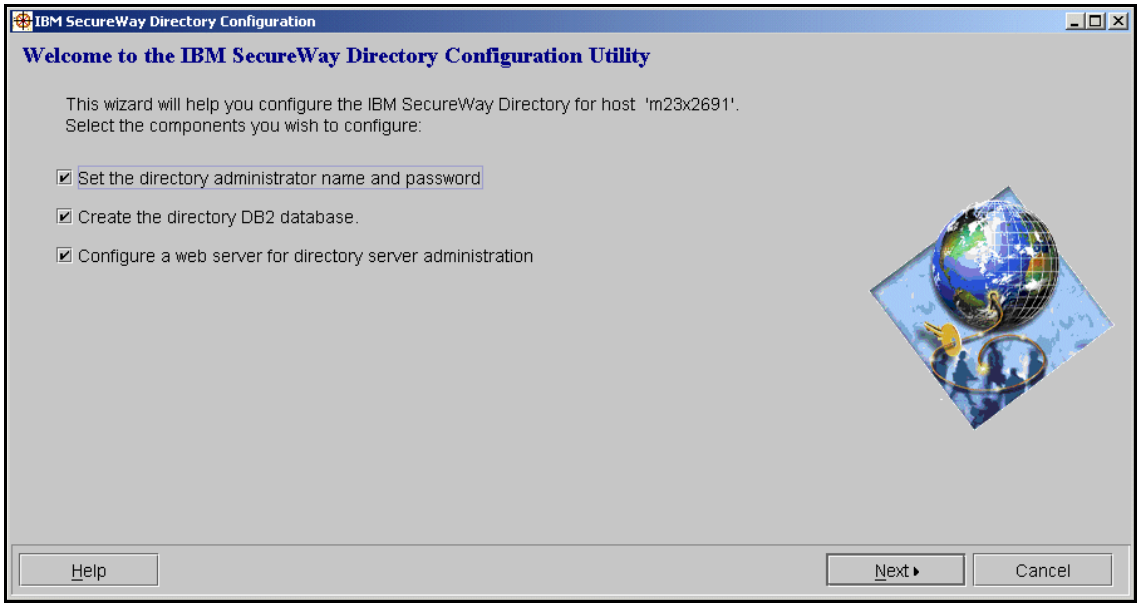


Figure 2-2 IBM SecureWay Directory V3.2.2 configuration options

4. Enter the Administrator DN and password, and then click **Next**. We recommend that you keep the default Administrator DN, `cn=root`.
5. When the DB2 Database window appears, select **Create a default LDAPDB2 database**, then click **Next**.
6. Select **Create a Universal DB2 database (UTF-8)** when prompted, and then click **Next**.
7. Select the drive where to create the database, and click **Next**.
8. A new window to configure the Web server appears. Select **IBM HTTP Server** and then click **Next**.

9. Enter the full path name of the Web server configuration file. By default, the configuration tool suggests a file path. Ensure it is correct and change it if needed. Click **Next**.
10. A Configuration Summary window appears as shown in Figure 2-3 on page 43. Review the values entered and click **Configure** to start the configuration process.

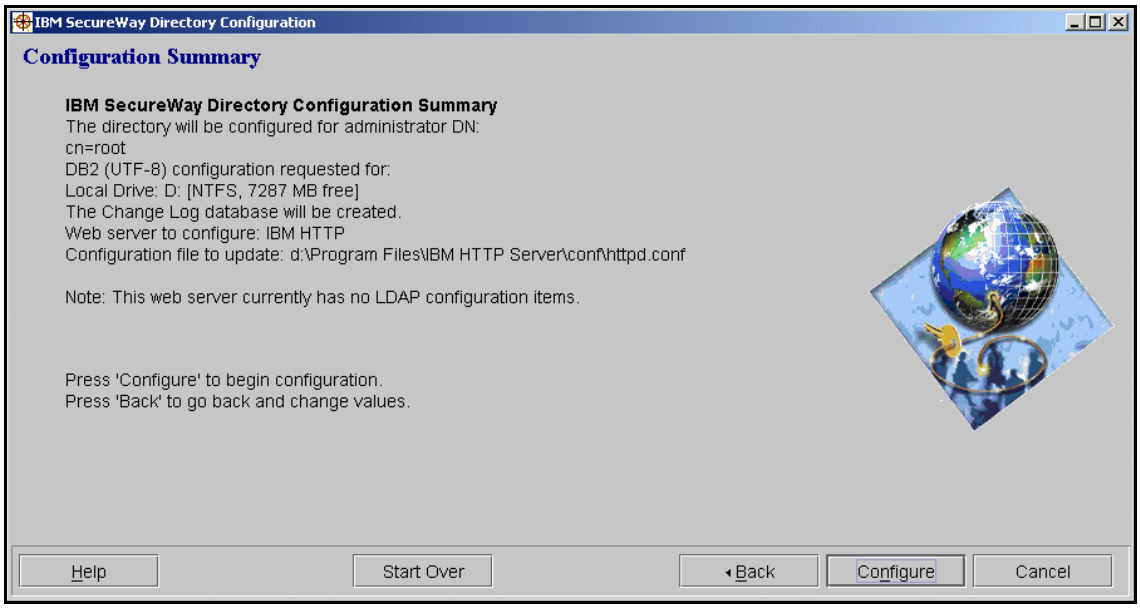


Figure 2-3 IBM SecureWay Directory configuration summary window

11. When the Configuration Summary window appears, review the results and then click **OK**.

2.3.7 IBM SecureWay Directory verification

After installing and configuring the IBM SecureWay Directory, we recommend that you verify the functionality of the server.

Starting services

The following services must be started for SecureWay Directory to be managed.

1. Start the IBM HTTP Server.

   This can be done in Windows Services or from the command line as follows:

```
c:\> net stop "IBM HTTP Server"
c:\> net start "IBM HTTP Server"
```

2. Start the IBM SecureWay Directory V3.2.2 server.

   This can be done in Windows Services or from the command line as follows:

   ```
   c:\> net start "IBM SecureWay Directory V3.2.2"
   ```

   Ensure the service “DB2 - LDAPDB2” is also started. To list the services running, type:

   ```
   c:\net start
   ```

   If service “DB2 - LDAPDB2” is not running, type:

   ```
   c:\net start "DB2 - LDAPDB2"
   ```

Verify the administration tools

Verify the following IBM SecureWay Directory administration tools:

1. Start the IBM SecureWay Directory Administration Web based tool.
   a. Enter the following URL in a Web browser.

      ```
      http://<ldap_server_hostname>/ldap
      ```

   b. Logon as follows:
      - Admin ID: `cn=root`
      - Password: `<password>`
2. Start the IBM SecureWay Directory Management Tool (DMT).

   To launch the DMT, select **Start -> Programs -> IBM SecureWay Directory -> Directory Management Tool**.

Note: For more detailed information on verifying the IBM SecureWay Directory, refer to the *WebSphere Commerce V5.4 Handbook*, SG24-6567.

2.3.8 Import data for WebSphere Portal and WebSphere Commerce

Now that IBM SecureWay Directory is installed, configured and the administration tools have been verified, the directory server needs to be configured for WebSphere Portal and WebSphere Commerce.

This section includes the following:

- Create a new suffix
- Import WebSphere Portal LDIF
- Import WebSphere Commerce LDIF
- DMT configuration for WebSphere Commerce

Create a new suffix

Before data can be imported into the IBM SecureWay Directory database, a suffix must exist.

1. Ensure the IBM SecureWay Directory service is started.
2. Start the IBM SecureWay Directory Administration Web-based tool by entering the following URL in a Web browser:

   `http://<hostname>/ldap`

3. When the Administration Logon windows appears, enter the following:
   - Admin ID: `cn=root`
   - Password: `<password>`
4. From the left navigation page, select and expand **Settings -> Suffixes**.
5. Enter the name of the suffix DN.

   This is the suffix DN that will be contained in the LDAP Data Interchange Format (LDIF) file used to load data as follows and then click **Update**. We entered the following:

   - Suffix DN: `dc=ibm,dc=com`

   Note: This is the base suffix that will be used by the WebSphere Portal Server and the WebSphere Commerce Server.

   - There is no space between the comma: `dc=ibm,dc=com`
   - A convention used for naming the suffix DN is to use the TCP/IP domain/hostname. For example, the IBM SecureWay Directory hostname is `ldap1.itso.ral.ibm.com`, and the suffix DN is `dc=ibm,dc=com`.

6. After adding the base DN suffix, stop and start the IBM SecureWay Directory Server.

Import WebSphere Portal LDIF

To import the WebSphere Portal LDIF file, do the following:

1. We created a WebSphere Portal LDIF from the WebSphere Portal template LDIF. The ITSO created sample WebSphere Portal LDIF file can be found:

   `c:\redp3684-code\ldif\wp-itso.ldif`

   Note: For detailed information on obtaining and unpacking the ITSO provided sample code and configuration files, refer to 4.1, “ITSO sample code” on page 184.

2. Modify the `wp-itso.ldif` file for your environment. For example, the DN suffix `dc=ibm,dc=com` may need to be changed for your environment.
3. From the IBM SecureWay Directory Administration tool, select and expand **Database -> Import LDIF**.
4. Enter the full path of the LDIF file containing the WebSphere Portal data as follows and then click **Import**:

   ```
   c:\redp3684-code\ldif\wp-itso.ldif
   ```

5. A summary report is displayed when the operation is finished.
6. After the LDIF file has been imported, delete the file from the system.

Note: Restarting the IBM SecureWay Directory is not needed after importing and LDIF file.

Import WebSphere Commerce LDIF

To import the WebSphere Commerce LDIF file, do the following:

1. Ensure prerequisite software mentioned in the *Getting Started, IBM Commerce Enhancement Pack* have been completed.
2. We modified the WebSphere Commerce LDIF from the Commerce Enhancement Pack to include cn=user on the following line:

   ```
   dn: uid=wcsadmin,cn=users,dc=ibm,dc=com
   ```

   Without the update, the wcsadmin user was not visible from the DMT.

   The ITSO created sample WebSphere Commerce LDIF file can be found:

   ```
   c:\redp3684-code\ldif\wc-itso.ldif
   ```

   Note: For detailed information on obtaining and unpacking the ITSO provided sample code and configuration files, refer to 4.1, “ITSO sample code” on page 184.

3. From the IBM SecureWay Directory Administration tool, select and expand **Database -> Import LDIF**.
4. Click **Clear Results**.
5. Enter the full path of the LDIF file containing the WebSphere Commerce data as follows and then click **Import**:

   ```
   c:\redp3684-code\ldif\wc-itso.ldif
   ```

6. A summary report is displayed when the operation is finished.
7. After the LDIF file has been imported, delete the file from the system.

Note: Restarting the IBM SecureWay Directory is not needed after importing and LDIF file.

DMT configuration for WebSphere Commerce

After the WebSphere Commerce LDIF file has been imported, we need to do the the following using the DMT to configure the Directory Server:

- Start the DMT.
- Change the wcsadmin password.
- Create the wasadmin user.
- Create an organization.

Start the DMT

1. To start the Directory Management Tool, click **Start -> Programs -> IBM SecureWay Directory -> Directory Management Tool**.
2. By default, the DMT is started as an anonymous user. To logon as an authenticated user do the following:
   a. After starting the DMT, select and expand **Server -> Rebind**.
   b. When the Rebind window appears, select **Authenticated**, enter the following and then click **OK**:
      - User DN: `cn=root`
      - User password: `<password>`

Change the wcsadmin password

To change the `wcsadmin` password, which was created via the wc-itso.ldif import, do the following:

Note: The wcsadmin password must match the password used when accessing WebSphere Commerce.

1. From the DMT, select and expand **Directory tree -> Browse tree**.
2. Select and expand the base DN (for example, `dc=ibm,dc=com`).
3. Double-click **uid=wcsadmin**.
4. Enter the wcsadmin password of the WebSphere Commerce node in the userPassword field of the DMT and then click **OK**.

Create the wasadmin user

To create the wasadmin user do the following:

Note: The wasadmin user will be used later as the Security Server ID during the configuration of WebSphere Security.

1. From the DMT, select and expand **Directory tree -> Browse tree**.
2. Select the base DN suffix (for example, `dc=ibm,dc=com`)
3. Click **Add** from the menu bar.
4. When the Add an LDAP Entry window appears, enter the following and then click **OK**:
   - Entry type: select **User**
   - Parent DN: `dc=ibm,dc=com`
   - Entry RDN: `cn=wasadmin`
5. A more detailed Edit an LDAP User window will appear, enter the following and then click **Add**:
   - objectClass (Object class): select **Top** from the pull-down list
   - sn (Last name): `wasadmin`

     We found that if we did not enter a value in the sn field, we received an error that would not allow the creation of this user.

   From the Business tab, enter the following:

   - userPassword: `<wasadmin_password>`

   From the Other tab, enter the following:

   - uid: `wasadmin`

Create an organization

To create the following organizations for the B2B store sample:

```
o=BuyerOrgA,dc=ibm,dc=com
o=BuyerOrgB,dc=ibm,dc=com
```

For each organization, do the following:

1. From the DMT, select and expand **Directory tree -> Browse tree**.
2. Select the base DN suffix (for example, `dc=ibm,dc=com`)
3. Click **Add** from the menu bar.
4. When the Add an LDAP Entry window appears, enter the following and then click **OK**:

- Entry type: select **Organization**
- Parent DN: `dc=ibm,dc=com`
- Entry RDN: `o=BuyerOrgA`

5. When the Add an LDAP Entry window appears, click **Add**.
6. Repeat the process for `BuyerOrgB`.
7. Click **Refresh tree**.
8. After the refresh, the directory entries should look like Figure 2-4 on page 49.

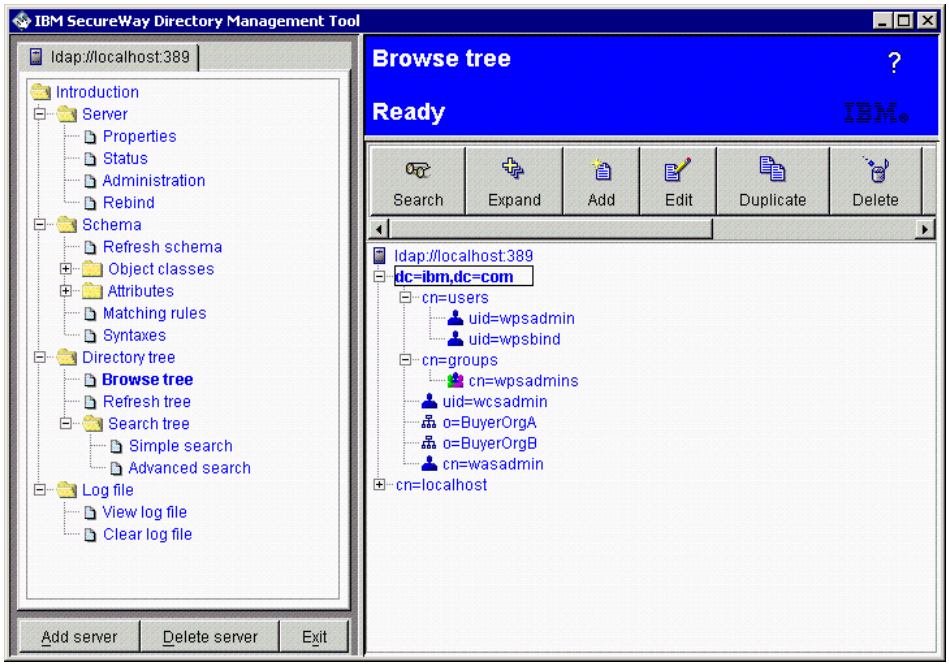


Figure 2-4 SecureWay Directory organization entries

9. Click **Exit** to close the DMT.

2.3.9 Configure WebSphere Commerce for LDAP

This section describes the configuration steps needed on the WebSphere Commerce node to configure WebSphere Commerce to use the SecureWay Directory Server (LDAP).

Note: More detailed information can be found in the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*

This section is organized as follows:

- Configure the WebSphere Commerce instance for LDAP.
- Configure the ldapentry.xml mapping file.
- Update WebSphere Commerce instance database for LDAP.
- Verify the LDAP and WebSphere Commerce configuration.

Configure the WebSphere Commerce instance for LDAP

To change the authentication mode used by WebSphere Commerce instance, complete the following steps:

1. Ensure the following tasks have been completed:
   - "Create a new suffix" on page 45
   - "Import WebSphere Commerce LDIF" on page 46
   - "DMT configuration for WebSphere Commerce" on page 47
2. Start the WC Configuration Manager Server Windows service.
3. Start the WebSphere Commerce Configuration Manager and logon.
4. From the left pane, select and expand **WebSphere Commerce -> <node> Instance List -> <instance_name> -> Instance Properties -> Member Subsystem**.
5. A pane should be displayed with the current Authentication Mode. From the Authentication Mode pull-down, select **LDAP**.
6. New fields should appear in the right pane under the Authentication Mode. For our example, we entered the following and then clicked **Apply**:
   - Authentication Mode: select **LDAP**
   - LDAP Version: `V3`
   - LDAP Type: select **IBM SecureWay**
   - Single sign-on: check **Single Sign-on**
   - Host: `<ldap_hostname>`

     For example, `wcldap1.itso.ral.ibm.com`
   - Port: `389`
   - Administrator Distinguished Name: `cn=root`
   - Administrator Password: `<password>`
   - Confirm Password: `<password>`
   - LDAP Authentication Mode: select **Simple**
   - Time out: `240`
   - Entry File Name: `c:/ibm/wc/xml/ldap/ldapentry.xml`
7. Close the Configuration Manager.
8. An additional value needs to be modified manually in the WebSphere Commerce <instance>.xml file. The value cannot be changed from the Configuration Manager.

a. To change the value, open a text editor and edit the <instance>.xml file found at:

   `<WC_HOME>\instances\<instance_name>\xml\<instance>.xml`

b. Locate the attribute `MigrateUsersFromWCSdb` in the <instance>.xml file. It is found inside the `<Directory>` tag in the <instance>.xml file, found inside `<MemberSubSystem>` tags.

c. Modify the attribute `MigrateUsersFromWCSdb`, set the value to ON (for example, `MigrateUsersFromWCSdb="ON"`). This property forces WebSphere Commerce to replicate the users from the WebSphere Commerce instance database to the LDAP directory database.

Note: The actual replication of information from LDAP to WebSphere Commerce does not occur until the WebSphere Commerce store registered user logs on to the store.

9. The changes made to the WebSphere Commerce <instance>.xml will not be in effect until the WebSphere Commerce <instance> application server has been restarted. We instruct you to restart the application server after modifying the ldapentry.xml file in the next step.

Configure the ldapentry.xml mapping file

For details on how to configure the ldapentry.xml mapping file, refer to the *WebSphere Commerce V5.4 Handbook*, SG24-6567. We will use an ldapentry.xml the provided with the IBM Commerce Enhancement Pack - October 2002 Edition for commerce enabled portals (wcsportalldap.xml).

1. Change directory to the `<WC_HOME>\xml\ldap`.
2. Backup the existing ldapentry.xml file to ldapentry_org.xml.
3. Copy the Commerce Enhancement Pack provided `<CEP_HOME>\Base\WPS_WCS_LDAP_Integration_Config\wcsportalldap.xml` to the `<WC_HOME>\xml\ldap\ldapentry.xml` on the Commerce Application Server node.

Note: The path and filename of the ldapentry.xml need to be the same as defined in the WebSphere Commerce <instance>.xml file. This information was entered using the Configuration Manager in, “Configure the WebSphere Commerce instance for LDAP” on page 50.

4. Restart the WebSphere Commerce <instance> application server from the WebSphere Application Server Administrative Console.

Update WebSphere Commerce instance database for LDAP

The WebSphere Commerce instance database needs to be updated as part of the LDAP configuration for the organizations and users created in previous steps (review the contents of the updatedbforldap.sql).

1. Open a DB2 command window on the WebSphere Commerce node.
2. Connect to the WebSphere Commerce instance database.

   ```
   db2 connect to <wc_dbname> user <dbadmin> using <password>
   ```

3. Change to the directory of the commerce enabled portals unzip directory included with the IBM Commerce Enhancement Pack - October 2002 Edition.

   ```
   <CEP_HOME>\Base\WPS_WCS_LDAP_Integration_Config
   ```

4. Execute the following sql script to update the WebSphere Commerce instance database:

   ```
   db2 -tvf updatedbforldap.sql
   ```

   Note: The `updatedbforldap.sql` file may need to be updated for your environment.

5. Disconnect from the database.

   ```
   db2 disconnect <wc_dbname>
   ```

Verify the LDAP and WebSphere Commerce configuration

After all of the configuration for WebSphere Commerce and LDAP has been completed, we recommend that you verify the functionality of the WebSphere Commerce tools, before enabling WebSphere Application Server security, single sign-on and SSL.

Refer to, “Verify the administration tools” on page 44 for details.

2.3.10 Enable SSL between WebSphere Commerce and LDAP

By default, the connection between the Commerce Application Server node and the Directory Server node is not secure. Within a test environment, this section is optional. Enabling SSL in a production environment is recommended.

Note: For details on how to SSL enable the connection between the nodes, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*

2.4 WebSphere Portal node implementation

This section describes the high level steps to install the WebSphere Portal node within the ITSO runtime environment. WebSphere Commerce V5.4 commerce enabled portals provided in the IBM Commerce Enhancement Pack - October 2002 Edition officially support WebSphere Portal Enable V4.1.3a. In our scenario, we used WebSphere Portal Enable V4.1.4.

Note: We also used WebSphere Portal Extend V4.1.4, but not document specific procedures for this package WebSphere Portal (not officially supported by WebSphere Commerce.

The WebSphere Portal node can be installed by one of the following approaches:

- WebSphere Portal Setup Manager umbrella install

  The Setup Manager works reasonable well when all components are installed on the same node.

  Note: For details on installing WebSphere Portal using Setup Manager, refer to the WebSphere Portal product documentation.

- WebSphere Portal component install

  Installing the components of the WebSphere Portal environment is desirable in a multi-node environment such as our example, or if you are someone responsible for troubleshooting the environment. This type of install will provide a better understanding of how to verify and debug each component before proceeding to install and configure the next component of the WebSphere Portal node. This section documents the procedure for the component based WebSphere Portal installation.

  Note: For more information on the WebSphere Portal component installation, refer to the following:

  - Redpaper *WebSphere Portal V4.1, Windows 2000 Installation,* REDP3593
  - Redbook *IBM WebSphere Portal V4.1 Handbook*, SG24-6883

These high level implementation steps for the WebSphere Portal node are as follows:

1. Windows 2000 Server installation
2. DB2 Server installation

3. Create databases for the WebSphere Portal node
4. IBM HTTP Server installation
5. WebSphere Application Server installation
6. WebSphere Personalization installation
7. WebSphere Portal installation
8. Commerce Enhancement Pack configuration

2.4.1 Windows 2000 Server installation

In preparation for the installation of DB2, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.
2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Create a token object
   - Increase quotas
   - Log on as a service
   - Replace a process level token
3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

2.4.2 DB2 Server installation

In our example scenario, the DB2 Server is initially installed on the WebSphere Portal node to host the WebSphere Application Server repository for WebSphere Portal and WebSphere Personalization, and the WebSphere Portal databases. After the WebSphere Portal node has been verified to test the functionality with WebSphere Commerce, we will configure a connection to a remote DB2 Server and migrate the local application databases to the remote DB2 Server.

- Install DB2 UDB V7.2 Enterprise Edition.
- Install DB2 V7 FixPak 7 (7.1.0.68).
- Update JDBC level to JDBC2.

For details on the installation of the DB2 Server, refer to 2.7, “DB2 Server node implementation” on page 86.

2.4.3 Create databases for the WebSphere Portal node

This section describes the steps needed to create databases required for the WebSphere Portal node.

1. Open DB2 Command window.
2. Create WebSphere Application Server repository database for the WebSphere Portal node by executing the following commands in the DB2 Command Window:

   ```
   db2 create db was4wp
   db2 update db cfg for was4 using applheapsz 1024
   ```

3. Create the WebSphere Portal database by executing the following commands in the DB2 Command Window:

   ```
   db2 create db wpsdb using codeset UTF-8 territory US
   db2 update db cfg for wpsdb using applheapsz 1024 app_ctl_heap_sz 1024
   ```

   Where <wpsdb> is the WebSphere Portal database.

4. Create the Member Services database, as follows:

   ```
   db2 create db wmsdb using codeset UTF-8 territory US
   ```

   Where <wmsdb> is the Member Services database.

   Note: This database can be created as part of the WebSphere Portal database, but we recommend that you create it separately as documented in this step.

5. Update the database configuration for Member Services.

   ```
   db2 update db cfg for wmsdb using applheapsz 16384
   db2 update db cfg for wmsdb using stmtheap 60000
   db2 update db cfg for wmsdb using app_ctl_heap_sz 8192
   db2 update db cfg for wmsdb using locklist 400
   db2 update db cfg for wmsdb using indexrec RESTART
   db2 update db cfg for wmsdb using logfilsiz 1000
   db2 update db cfg for wmsdb using logprimary 12
   db2 update db cfg for wmsdb using logsecond 10
   ```

   Note: If you did not create a separate Member Services database, substitute wmsdb with wpsdb with the database configuration updates listed.

2.4.4 IBM HTTP Server installation

For details on installing and configuring the IBM HTTP Server, refer to 2.2.3, “IBM HTTP Server installation” on page 16. The following tasks need to be completed:

- Install the IBM HTTP Server.
- Configure the IBM HTTP Server.
- Verify the IBM HTTP Server.

Note: In our example, we first installed the IBM HTTP Server on the WebSphere Portal node. Later we migrated to a remote Web server configuration as described in 2.8, “Remote Web server node implementation” on page 91.

2.4.5 WebSphere Application Server installation

This section describes the high level installation steps for the WebSphere Application Server on the WebSphere Portal node.

Notes:

- For details refer to the procedures documented in 2.2.4, “WebSphere Application Server installation” on page 17.
- The WebSphere Application Server repository database procedure is slightly different for this node and has already been completed in 2.4.3, “Create databases for the WebSphere Portal node” on page 55.

1. Create the WebSphere Application Server repository database.

   Refer to, 2.4.3, “Create databases for the WebSphere Portal node” on page 55.
2. Install WebSphere Application Server V4.0.1.

   Refer to, “Install WebSphere Application Server V4.0.1” on page 19.

   Note: During the installation of WebSphere Application Server you will be prompted for the WebSphere database name. We created this database in a previous step as `was4wp`. We did so to avoid a naming problem with other repositories in the event the database is migrated to a common remote database server.
3. Install WebSphere Application Server V4 FixPak 4 (V4.0.4).

   Refer to, “Install WebSphere Application Server V4 FixPak 4 (V4.0.4)” on page 19.
4. Install WebSphere Application Server V4.0.4 e-Fixes.

   Refer to, “Install WebSphere Application Server V4.0.4 e-Fixes” on page 20.

Note: In addition to the e-Fixes listed in, “Install WebSphere Application Server V4.0.4 e-Fixes” on page 20, install PQ66355 on the WebSphere Portal node.

The e-Fix can be downloaded from:

http://www.ibm.com/software/webservers/appserv/support/

5. Configure the WebSphere Application Server.

   Refer to, “Configure the WebSphere Application Server” on page 20.

6. Verify the WebSphere Application Server.

   Refer to, “Verify the WebSphere Application Server” on page 21.

2.4.6 WebSphere Personalization installation

WebSphere Personalization is used to personalize content of the Web site using the WebSphere Personalization Rules and Resource Engines. We have chosen to manually install the WebSphere Personalization to avoid an installation dependency issue we encountered with the WebSphere Portal installation when using Setup Manager.

To install WebSphere Personalization of the WebSphere Portal node, do the following:

1. Ensure the following Windows services are started on the WebSphere Portal node and the Directory Server node:

   WebSphere Portal node:

   – DB2
   – IBM HTTP Server
   – IBM WS AdminServer 4.0

   Directory Server node:

   – DB2
   – IBM HTTP Server
   – IBM SecureWay Directory V3.2.2

2. Create the WebSphere Portal application server.

   a. Start the WebSphere Administrator’s Console on the WebSphere Portal node.

   b. From the Administrative Console, navigate to **WebSphere Administrative Domain -> Nodes -> <node> -> Application Servers**. Right-click and select **New**.

c. When the Create Application Server window appears, enter the following:

- Application Server name: `WebSphere Portal`

> **Note:** Spelling, case sensitivity and spacing are very important. The application server name WebSphere Portal is required by WebSphere Portal.

d. You should see the following message, and then click **OK**.

```
Command "EJBServer.create" completed successfully
```

3. Insert the IBM WebSphere Portal CD containing the WebSphere Personalization Recommendation Engine into the CD-ROM drive. Navigate to the personalization folder. Copy the personalization folder, including sub folders, to the local file system on the WebSphere Portal node (for example, c:\temp).
4. Navigate to the c:\temp\personalization\silent\response_files\nt folder. Right-click **nt** -> **Properties**.
5. Uncheck the Read-only box if checked, and then click **Apply**. Select **Apply changes to this folder, sub folders and files**. Click **OK**. Confirm File Attribute Changes window.

> **Note:** This step is only necessary if you copied the folder using Windows Explorer. If you copied the directory from a command prompt, the Read-only flag is not preserved from the CD.

6. Change to the `c:\temp\personalization\silent\response_files\nt` directory.
7. Open the `pzn_silent_server.txt` file in a text editor and modify the file as follows (see Example 2-3):

   From:

   ```
   -W bean28.appServer="Default Server"
   ```

   To:

   ```
   -W bean28.appServer="WebSphere Portal"
   ```

Example 2-3 pzn_silent_server.txt

```
# This is the response file for Personalization v4.0 Server
# specify silent install
-silent
# Application server where Personalization Server will be added.
-W bean28.appServer="WebSphere Portal"
```

```
# set WAS context roots for installable enterprise applications
-W EmailEarContext.replaceValue="/wps/PersEmail"
-W RuntimeEarContext.replaceValue="/wps/PersAdmin"
# global properties for preventing pop-up messages
-G replaceExistingResponse="yesToAll"
-G replaceNewerResponse="yesToAll"
-G removeExistingResponse="yesToAll"
-G removeModifiedResponse="yesToAll"
-G createDirectoryResponse="yes"
```

Important: It is crucial that the entry of WebSphere Portal exactly as shown, otherwise the install will not work. This includes case-sensitivity and any blank spaces.

8. Save the file.
9. Navigate to the `C:\temp\personalization\silent\nt` directory.
10. Double-click **pzn_silent_server.bat**. This will begin the install of Personalization in silent install mode. The file `pzn_silent_server.txt` that was modified will be used for the installation.

Note: The silent install (`pzn_silent_server.bat`) will take several minutes to complete.

11. After the installation, restart the WebSphere Portal node.

2.4.7 WebSphere Portal installation

This section describes how to install and configure the WebSphere Portal server for the ITSO commerce enable portal environment.

To install the WebSphere Portal Server, do the following:

1. Ensure the following Windows services are started:

   WebSphere Portal node:

   – DB2
   – IBM HTTP Server
   – IBM WS AdminServer 4.0

   Directory Server node:

   – DB2
   – IBM HTTP Server
   – IBM SecureWay Directory V3.2.2

2. Insert the IBM WebSphere Portal Server CD in the CD-ROM drive. Navigate to the `wps` directory and run `install.bat`.
3. When the Welcome window appears, click **Next**.
4. When the WebSphere Portal Installer Prerequisites window appears, click **Next** provided your system meets the requirements.

   Note: In our example, we have higher levels of WebSphere Application Server (FixPak 4), DB2 (FixPak7), SecureWay Directory (3.2.2 e-Fix 2).

5. When prompted for the directory name to install, accept the default (`c:\ibm\PortalServer`) and click **Next**.
6. When the WebSphere Portal Installer prompts you to select Standard or Development, we selected **Standard** for our runtime environment and then click **Next**.
7. When prompted for to select the installation type, we selected **Standard** and then clicked **Next**.
8. When prompted to select the authentication mode for Member Services, we selected **Database + LDAP** and then clicked **Next**.
9. When prompted to enter the Distinguished Name (DN), we entered the following and then clicked **Next**:
   - Administrator UID: `wpsadmin`
   - Administrator Password: `<wpsadmin_password>` (default wpsadmin)
   - Confirm Password: `<wpsadmin_password>`

   Note: The `wpsadmin` user was created during the WebSphere Portal LDIF import step in, “Import WebSphere Portal LDIF” on page 45 during the Directory Server node configuration.

10. When prompted to select the global security for the WebSphere Application Server, we selected **Configure global security** and then clicked **Next**.
11. When prompted to enter the LTPA password for the WebSphere Application Server, we entered the following then clicked **Next**:
    - LTPA Password: `wpsbind`
    - Confirm Password: `wpsbind`

    Note: The LTPA password is defined for the WebSphere Application Server, when configuring security and clicking on Generate Keys. This password must be consistent.

12. When prompted if you plan to use a third-party authorization and authentication engine, we selected **No** and then clicked **Next**.

> **Note:** In our example, we are using the IBM SecureWay Directory V3.2.2.

13. When prompted to select the type of LDAP directory server, we selected **IBM SecureWay Directory** and then clicked **Next**.
14. When prompted to enter the access information for the LDAP server, we entered the following and then clicked **Next**:
    - Server hostname: `wcldap1.itso.ral.ibm.com`

      Enter the fully qualified hostname of the LDAP server.
    - Port: `389`

    Enter the username and password who has read and update privileges on the LDAP server.
    - User DN: `cn=root`
    - Password: `<password>`
    - Confirm Password: `<password>`

> **Note:** Ensure that the WebSphere Portal node can connect to the Directory Server node using the fully qualified hostname before click Next.
>
> For example, can you ping the node by hostname, fully qualified hostname and IP address.

15. When prompted for the connection data to the LDAP server, we entered the following and then clicked **Next**:
    - Suffix: `dc=ibm,dc=com`

      This is the base suffix created in, “Create a new suffix” on page 45.
16. When prompted to select the customize LDAP settings, we selected **Use default LDAP settings** and clicked **Next**.

> **Note:** In our example, we previously configured the IBM SecureWay Directory Server and imported the WebSphere Portal and WebSphere Commerce LDIF files containing users.

17. When prompted to enter the node name of your system running WebSphere Application Server, we entered the node name and clicked **Next**.

Note: In our example, the WebSphere Portal node has its own WebSphere Application Server Administrative Server and domain (for example, `wcportal1`).

18. When prompted to enter the URL of the portal home page, we entered the following and then clicked **Next**:
    - Hostname: `wcportal1.itso.ral.ibm.com`

      This is the fully qualified hostname of the WebSphere Portal Web server virtual host. This will later be modified when adding the remote Web server for the WebSphere Portal.
    - Base URI: `/wps`
19. When the type the names of the portal pages window appears, we accepted the default and then clicked **Next**:
    - Home page: `/portal`
    - Customized page: `/myportal`
20. When prompted to enter the Internet security configuration and proxy window appears, we left the values blank and clicked **Next**. In our scenario, our network configuration does not use a proxy.
    - Proxy server hostname:
    - Proxy port:
21. When prompted to select the appropriate option, we selected **Deploy base portlets into Portal Server** and then clicked **Next**.
22. When prompted to select the type of database, we selected **IBM DB2** and then clicked **Next**.
23. When prompted to select an option to store portal data, we selected **Initialized an existing database** and then clicked **Next**.

    In our example, we created our database in 2.4.3, “Create databases for the WebSphere Portal node” on page 55.
24. When prompted for the database access or creation information, we entered the following and then clicked **Next**:
    - Database user: `<db2admin_user>`
    - Database password: `<password>`
    - Confirm password: `<password>`
    - Database name: `wpsdb`

      The WebSphere Portal database was created in 2.4.3, “Create databases for the WebSphere Portal node” on page 55.

25. When prompted for the JDBC driver used to access the Portal Server database, we accepted the default settings and clicked **Next**.
26. When prompted to select an option for Member Services data, we selected **Initialize an existing database** and then clicked **Next**.

    In our example, we created the Member Services database in 2.4.3, “Create databases for the WebSphere Portal node” on page 55.
27. When prompted for the database access or creation information, we entered the following and then clicked **Next**:
    - Database user: `<db2admin_user>`
    - Database password: `<password>`
    - Confirm password: `<password>`
    - Database name: `wmsdb`

      The WebSphere Portal Member Servers database was created in 2.4.3, “Create databases for the WebSphere Portal node” on page 55.
28. When prompted for the JDBC driver used to access the Member Services database, we accepted the default settings and clicked **Next**.
29. When the WebSphere Portal Server directory is displayed, we clicked **Next** to begin copying files.
30. After installing the files, you should see a message, `The database has been initialized successfully`. Click **Next**. This step is for the WebSphere Portal database.
31. Next you will see the Member Services database being initialized. When it is complete, you should see a message, `The database has been initialized successfully`. Click **Next**.
32. After more updates to the Member Services database, click **Next**.
33. The next step will configure the application server. When done, click **Next.**
34. You should see a message, `The configuration of the application server is complete.` Check the configuration output for errors and click **Next**.
35. You will see a warning message, `An external HTTP Server is running on this machine. You must stop and restart it before proceeding`. We restarted the IBM HTTP Server and then clicked **Next**.

    Note: Stop and start the IBM HTTP Server before proceeding.
36. A series of files will be displayed being backup. When complete click **Next**.
37. After the application server is configured, click **Next**.

38. After the window checking administrative server is complete, you will see a window like Figure 2-5 on page 64 listing a procedure that needs to be completed before continuing the installation.

Important: You are required to complete the procedure listed in Figure 2-5 on page 64 and documented below before proceeding. Do not click Next until you have completed this procedure. Failure to complete the procedure will result in an installation failure.

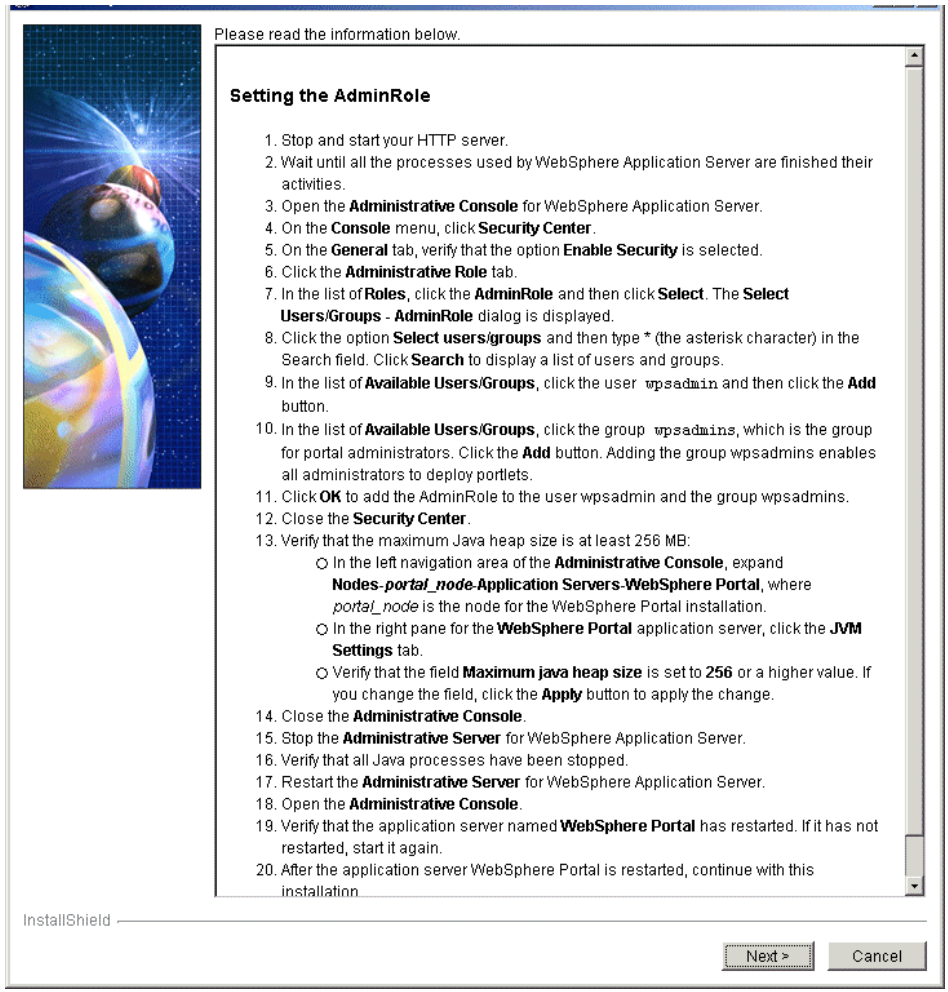


Figure 2-5 WebSphere Portal - Setting the Admin Role

To complete the steps listed in Figure 2-5, do the following:

a. Restart the IBM HTTP Server.

b. Start the WebSphere Administrative Console.

c. When the Login at Target Server window appears, enter the following and then click **OK**:

   - Realm/Cell Name: `wcportal1.itso.ral.ibm.com`
   - User Identity: `wpsbind`
   - User Password: `wpsbind`

   Note: The default port number is 389 for the LDAP node. For example, the Realm/Cell Name can be referenced as `<fully_qualified_hostname>:389`.

   During the WebSphere Portal installation, WebSphere Security is enabled. Once security is enabled, you will be prompted for the Security Server ID and password when the WebSphere Administrative Console is started.

d. Click **Console -> Security Center**.

e. Check **Enable Security** in the General tab.

f. Click the **Administrative Role** tab.

g. From the list of Roles, select **AdminRole** and then click **Select**.

h. When the Select Users/Groups - AdminRole window appears, check **Select users/groups.** Type a ***** in the search field and then click **Search**.

i. From the list of Available Users/Groups, select **uid=wpsadmin** under Users and then click **Add**.

j. From the list of Available Users/Groups, select **cn=wpsadmins** under Groups and then click **Add**. Then click **OK**.

k. Click **OK** to close the Security.

l. Select and expand the **WebSphere Administrative Domain -> Nodes -> <node> -> Application Servers**.

m. Select **WebSphere Portal**. In the right-hand pane, click the **JVM Settings** tab. Verify the Maximum Java heap size is set to 256 or higher. If you change the value, click Apply.

n. Click on your portal node under Nodes (for example, wcportal1). Right-click **Stop** to stop the WebSphere Administrative Server.

o. Restart the WebSphere Application Server Administrative Server.

p. Verify that the WebSphere Portal application server has been started. If not, start it. See note below.

Important: After restarting WebSphere Portal application server, check <PORTAL_HOME>\log\appserver-out.log file for any startup errors or Java exceptions. Even if the WebSphere Portal application server status is green (running) in WebSphere Application Server Administration Console, it does not necessarily mean that the application has started without errors.

In some cases you may see the following error message:

```
portal: Initialization failed!: java.lang.NoClassDefFoundError:
com/ibm/websphere/personalization/resources/Resource
```

If you see this error, you must do the following steps before clicking **OK** in the pop-up window, otherwise the installation will fail.

- Open WebSphere Application Server Administration Console.
- Select **WebSphere Portal** application server.
- Click **JVM Settings** tab.
- Add these two classpath entries to the Classpaths list:

  ```
  c:\ibm\was\personalization\lib\personalization.jar
  c:\ibm\was\personalization\lib\prCommon.jar
  ```

- Click **Apply**.
- Restart WebSphere Portal application server.

q. Verify that you can access the WebSphere Portal server from a Web browser by entering the following URL:

   `http://<portal_websrv_hostname>/wps/portal`

   You should see a WebSphere Portal page stating, `Your portal does not have any page groups`.

r. Verify the login by clicking on the Key symbol in the upper-right. Login with user ID `wpsadmin` and password `wpsadmin`.

39. Return to the WebSphere Portal Installer, and click **Next**.
40. You will see a window deploying portlets. When the deployment is complete, click **Next**.
41. When the WebSphere Portal Installer displays the message, `WebSphere Portal Final installation action`, review the text and click **Next**.
42. You should see a message, `The installShield Wizard has successfully install IBM WebSphere Portal Server. Choose Finish to exit the wizard`. Click **Finish**.

Congratulations, you have finished the WebSphere Portal installation.

2.4.8 Commerce Enhancement Pack configuration

This section describes the steps needed to install and configure the IBM Commerce Enhancement Pack - October 2002 Edition on the WebSphere Portal node in preperation for deploying commece enabled portlets.

Note: For more information, refer to *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*.

To configure the Commerce Enhancement Pack on the WebSphere Portal node, do the following:

1. Extract the `WebSphereCommerceEnabledPortal.zip` file provided with Commerce Enhancement Pack to C:\temp\CEP directory.
2. Install the commerce enabled portal enterprise application.
   a. Change directory to the following:

      `C:\temp\CEP\Base\PersonalizationUserHomePageBaseFolder\scripts`

   b. Create the CPS database used by the Commerce Enhancement Pack by executing the following:

      `createCPSDB.bat`

      You will be prompted for the following:

      - DB2 Admin UID: `db2admin`
      - Hostname: `wcportal1`
      - Password: `<db2_admin_password>`

   c. Ensure the WebSphere Administrative Server is running. This is required to create the CPS datasource in the next step.

   d. Create CPS datasource in WebSphere Application Server for the Commerce Enabled Portal Enterprise Application by executing the following:

      `createCPSDS.db2.bat`

      You will be prompted for the following:

      - CPS database name: `CPS`
      - Database admin UID: `db2admin`
      - Database admin password: `<db2_admin_password>`
      - DB2 JDBC Driver Path: `c:\ibm\sqllib\java\db2java.zip`
      - WAS node name: `wcportal1`
      - WAS path: `c:\ibm\was`

Note: The WebSphere node name is case sensitive.

Since the WebSphere Portal node is now configured for LDAP, a logon panel will be displayed. We logged on as `wpsadmin`.

e. Install the Commerce Portal Server enterprise application in the WebSphere Application Server.

`createCPSEA.bat`

You will be prompted with the following:

- WAS node name: `wcportal1`
- Install folder: `c:\ibm\cps`
- WAS path: `c:\ibm\was`

Note: WebSphere node name is case sensitive.

You will get a login window, since WebSphere Application Server security has been enabled. Login with wpsadmin.

f. Copy the jar files to WebSphere Application Server.

`modifyWAS.bat`

You will be prompted with the following:

- WAS Path: `c:\ibm\was`

Note: You will get a login window, since WebSphere Application Server security has been enabled. Login with wpsadmin.

g. For personalized accounts for the default store, run the following (optional):

`installNewWPSAttributes.bat`

You will be prompted with the following:

- WAS Path: `c:\ibm\was`
- Portal Path: `c:\ibm\PortalServer`

3. Verify that WebSphere Portal is functioning properly:

a. Ensure the following are started:

- DB2 - Windows service
- IBM HTTP Server - Windows service
- IBM WS AdminServer 4.0 - Windows service
- WebSphere Portal - application server

b. Enter the following WebSphere Portal home page URL in a Web browser:

   `http://wcportal1.itso.ral.ibm.com/wps/portal`

c. Verify that you can log in as `wpsadmin`.

d. Verify that you can log in with a user ID registered in WebSphere Commerce such as wcsadmin.

2.5 Enable single sign-on between WebSphere Portal and WebSphere Commerce

Now that the WebSphere Commerce node, Directory Server node, and WebSphere Portal node have been installed and configured, we can enable WebSphere security and enable single sign-on (SSO). This section explains how to enable single sign-on (SSO) of WebSphere Commerce with other secure applications such as WebSphere Portal, using the same Directory Server (LDAP). SSO allows the user to move between different applications (servers) without being prompted for a user ID and password (or certificate).

This section is organized as follows:

- Overview of single sign-on (SSO)
- WebSphere Portal configuration for SSO
- WebSphere Commerce configuration for SSO
- Verify single sign-on (SSO) configuration

Note: For more information on WebSphere Security and how to SSL enable the connection between the nodes, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- Redbook *IBM WebSphere V4.0 Advanced Edition Handbook*, SG24-6176
- Redbook *IBM WebSphere V4.0 Advanced Edition Security*, SG24-6520
- Product Guide *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*

2.5.1 Overview of single sign-on (SSO)

To enable SSO between different servers, the Lightweight Third Party Authentication (LTPA) mechanism is used. This mechanism uses a flag called an LtpaToken, which contains the user authentication information, the network domain in which the SSO should be valid, and the expiration time after the user is required to re-authenticate. The LtpaToken is encrypted using LTPA keys shared for all the SSO participating servers.

The token is created when the user successfully authenticates the first participating application (server). The server sends a *transient* cookie to the Web browser client. This means that the cookie resides in the Web browser memory and is not stored on the user's computer system. This type of cookie expires when the user closes the browser and is easily recognized by its name, *LtpaToken*.

The requirements for enabling SSO are as follows:

- Use the same LDAP directory for authentication.
- All SSO participating servers must be in the same DNS domain.
- The URLs must include the DNS domain (no IP addresses or hostnames).
- The Web browsers must be configured to accept cookies.
- The servers time and time zone must be synchronized (SSO token expiration time is absolute).
- All servers must share the LTPA keys to generate and decode the LtpaTokens.

WebSphere Commerce SSO implementation

The following considerations need to be kept in mind when planning to enable SSO with WebSphere Commerce:

- WebSphere Commerce cannot create a LtpaToken. The WebSphere Commerce Server does not create an LtpaToken when a user is authenticated using the default member subsystem login pages. A simple solution to this is to create a small WebSphere application just to perform the authentication. A production runtime environment may have something like a portal server with directory services that creates the LtpaTokens. The following steps will explain how to use the WebSphere security services to configure an existing application.
- To automatically sign on to the WebSphere Commerce Administrative tools, your must connect directly to the URL of one of the pages displayed after performing the logon. If you point to the logon page, a user ID and password must be supplied, because the page does not check if the LtpaToken is received.

Scenario for single sign-on

The following sections show how to implement a real SSO scenario. The scenario consists of two WebSphere Application Server applications on two different nodes:

- Node A running WebSphere Portal 4.1.4.
- Node B running WebSphere Commerce V5.4.

The scenario describes how to set up two different nodes for SSO and how a user logged into Node A can access the Node B application without being prompted for a user ID and password.

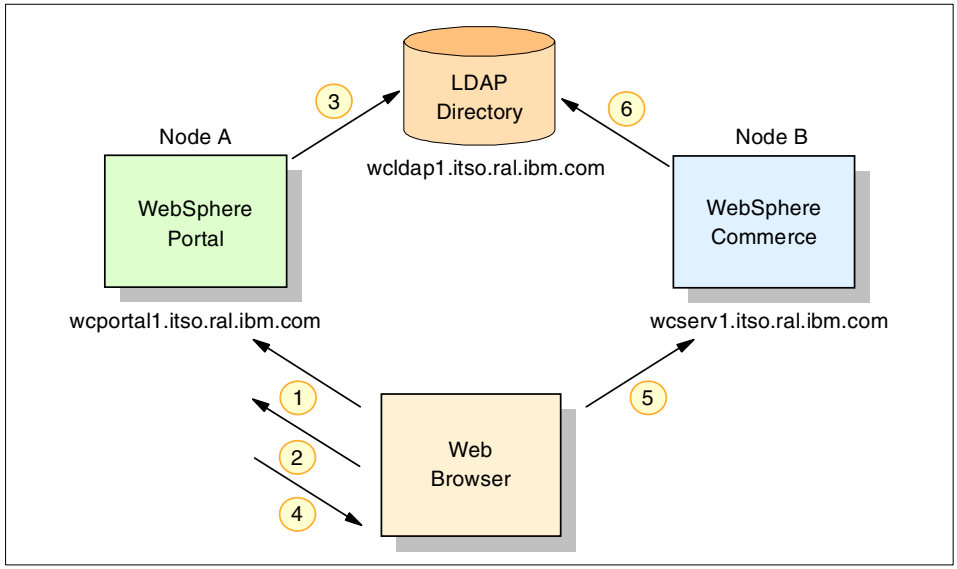


Figure 2-6 Sample scenario for SSO

The SSO authentication works as follows (see Figure 2-6):

1. The user enters the URL for the sample application on Node A. Since the application is configured to use LDAP as an authentication method, the WebSphere Application Server redirects to a logon page. In the case of WebSphere Portal, we will use the logon provided with the product.
2. The users enter a user ID and password.
3. The WebSphere Portal authenticates the users against the LDAP directory server.
4. The authentication is successful, so the WebSphere Application Server executes the application and the response is returned to the user. As SSO is also enabled, the WebSphere Application Server creates an LtpaToken and sends a cookie to the Web browser client.
5. The user accesses the WebSphere Commerce store on Node B. As Node B belongs to the same DNS domain, the Web browser client sends the cookie received in the previous step from Node A to Node B.
6. The WebSphere Commerce Server reads the cookie and decrypts the distinguished name (DN) of the user logged in and verifies the DN exists in

the LDAP directory. If the DN exists, the user is logged into the WebSphere Commerce store.

2.5.2 WebSphere Portal configuration for SSO

This section describes how to configure the WebSphere Portal node for single sign-on use with the WebSphere Commerce node.

The section is organized as follows:

- Enable WebSphere security/SSO for WebSphere Portal node.
- Verify the WebSphere security configuration.
- Verify the WebSphere Portal.
- Extract certificate from WebSphere Commerce Web server.
- Certificate trust security configuration.

Enable WebSphere security/SSO for WebSphere Portal node

In our example, WebSphere Commerce and WebSphere Portal are on separate nodes. For this reason, we must export the LTPA key (LtpaToken) from the WebSphere Portal node from WebSphere security, and import the key on the WebSphere Commerce node.

To configure global security settings for SSO in WebSphere, complete the following steps:

1. Open the WebSphere Administrative Console on the WebSphere Portal node. Logon as `wpsbind`, which is the default WebSphere Security Server ID set by the WebSphere Portal installation.
2. From the WebSphere Administrative Console menu bar, select **Console -> Security Center**.
3. Click the **General tab** and check **Enable security**.
4. Click the **Authentication** tab and choose **Lightweight Third Party Authentication (LTPA)**.
5. New options for LTPA setting will be available on the window. Specify the following LTPA settings:

   Note: Many of the options listed will have the proper value already set. We recommend that you carefully review each setting.

   – Token Expiration: `120`

     How many minutes can pass before a client using an LtpaToken must authenticate again in the Token Expiration field.

– Ensure the **Enable Single Sign-On (SSO)** is checked. The Domain field will be enabled once Enable Single Sign-On is checked.

– Domain: `itso.ral.ibm.com`

  Enter a DNS domain name in the Domain field. In our example we set this domain to `itso.ral.ibm.com`. This domain name is used when the HTTP cookie is created for SSO and determines the scope to which SSO applies.

From the same page, select the **LDAP** radio button and enter the following for accessing the LDAP directory:

– Security Server ID: `wpsbind,cn=users,dc=ibm,dc=com`

> **Note:** The Security Server ID is set during the WebSphere Portal installation. In our example, it was set to `wpsbind,cn=users,dc=ibm,dc=com`.
>
> We found that by changing the Security Server ID, we were not able to logon to WebSphere Portal with the wpsadmin or wpsbind user IDs.

– Security Server Password: `<password>`

– Host: `<fully_qualified_hostname_of_LDAP_server>`

  For example, we entered `wcldap1.itso.ral.ibm.com`.

– Directory Type: Select **Custom**

– Port: `389`

– Base Distinguished Name: `dc=ibm,dc=com`

  Base entry DN where the users can be found.

– Bind Distinguished Name: (we left field blank on the WebSphere Portal node, which is the default from the WebSphere Portal installation).

  This is the SecureWay Directory user that will do the bind to the LDAP directory (used to rebind as an authenticated user in the DMT).

– Bind Password: (left field blank)

After entering these values, the window should look like Figure 2-7 on page 74. Do not click Apply or OK just yet.

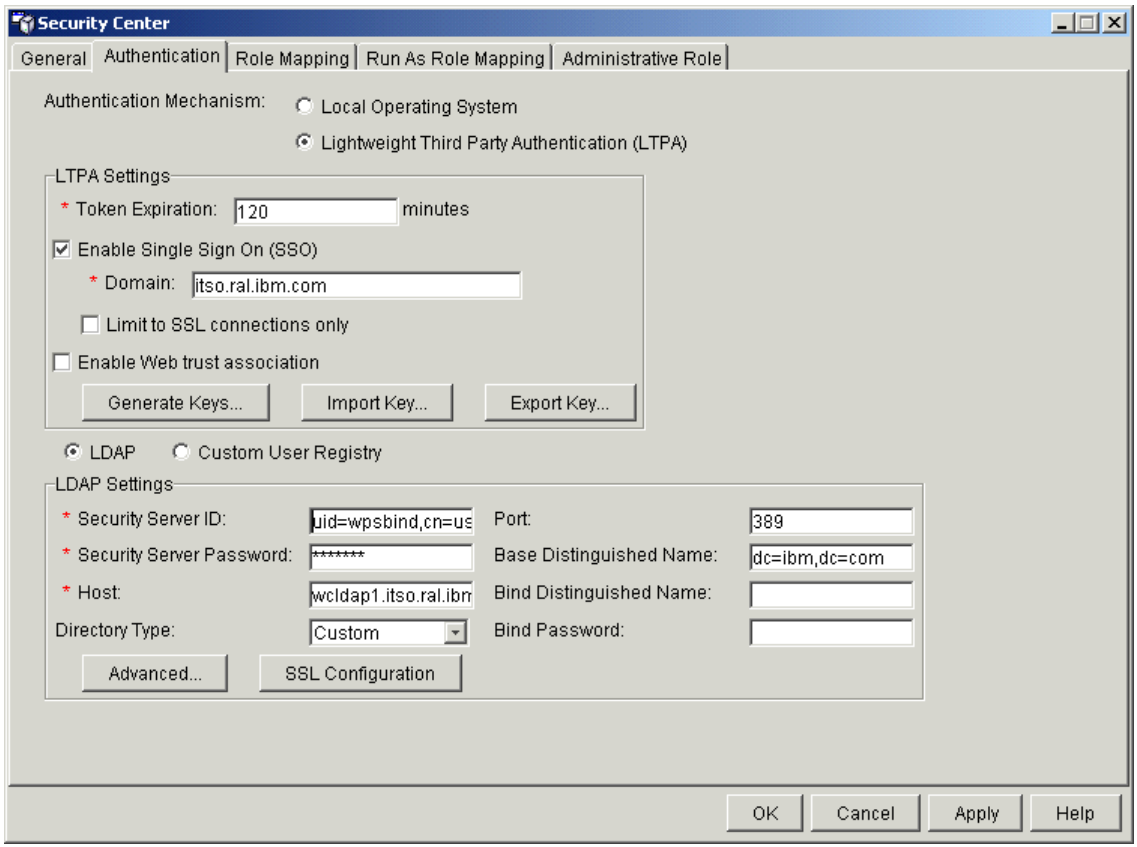


Figure 2-7 WebSphere security SSO and LDAP settings for the WebSphere Portal node

6. From the same page, click the **Generate Keys...** button to create the LTPA keys for encrypting the LtpaToken.
   a. You will be prompted for an LTPA password for protect the set of encryption keys, type in your password (for example, `wpsbind`).
   b. The LTPA keys will be shared with the WebSphere Commerce Server. Click **OK**.

   Note: A message `Command "Generate LTPA keys" created successfully` should appear in the message log of the WebSphere Administrative Console.

7. Once the LTPA Keys are generated, click **Export Key...** You will be prompted with an Export To File window. For example, we created an LTPA Key file called `wp_wasltpa.key` in the `c:\ibm\was\bin` directory.

Keep this file in a secure place because this keys will be imported into the WebSphere Application Server of the WebSphere Commerce node.

8. Now that all Security and single sign-on settings have been entered, click **OK**.
9. To verify the settings were saved, by reviewing the file:

   `<WAS_HOME>\properties\sas.server.props`

10. The WebSphere Administrative Server will need to be restarted for the changes made to be in effect.

Verify the WebSphere security configuration

To verify that WebSphere security is configured correctly, do the following:

1. Start the Default Server application server from the Adminstrative Console.
2. Enter the following URL in a Web browser:

   `http://<wcportal_hostname>/webapp/examples/showCfg`

3. The WebSphere basic authorization page should appear. We entered the following and then clicked **Submit Login**:

   – Account: `wasadmin`
   – Password: `<password>`

   You should see the IBM WebSphere Web Containter Configuration.

4. Stop the Default Server application server.

Verify the WebSphere Portal

Verify that the WebSphere Portal is functioning properly after the WebSphere security configuration as follows:

1. Ensure the following Windows services are started:

   – DB2 - Windows service
   – IBM HTTP Server - Windows service
   – IBM WS AdminServer 4.0 - Windows service

2. Start the WebSphere Portal application server.

   a. Start the WebSphere Administrative Console.

   b. When prompted for the logon information, we entered the following and then clicked **OK**, see:

      - Real/Cell Name: wcldap1.itso.ral.ibm.com® (this field is greyed out, can not enter info).
      - User Identity: `wpsadmin`
      - User Password: `<password>`

c. Select and expand **WebSphere Administrative Domain -> Nodes -> <node> -> Application Servers**.

d. Select the **WebSphere Portal** application server.

e. Right-click **Start**.

3. Enter the following WebSphere Portal home page URL in a Web browser:

   `http://wcportal1.itso.ral.ibm.com/wps/portal`

4. Verify that you can log in as `wpsadmin`.

5. Verify that you can log in with a user ID registered in WebSphere Commerce such as wcsadmin.

Extract certificate from WebSphere Commerce Web server

On the WebSphere Commerce node, extract the certificate from the IBM HTTP Server as follows:

1. Start the IBM HTTP Server IKeyMan Utility by clicking **Start -> Programs -> IBM HTTP Server -> Start Key Management Utility**.

2. From the menu bar, click **Key Database File -> Open**. Enter the path to the keyfile (key store). For example, c:\ibm\http\ssl\http_key.kdb.

   The key file is referenced in the <HTTP_HOME>\conf\httpd.conf file.

3. You will be prompted to enter the key store database password.

4. Select the certificate (for example, http_ssl) and click **Extract Certificate**.

5. When the Extract Certificate window appears, enter the following and click **OK**:

   – Data type: select **Base64 encoded ASCII data**
   – Certificate file name: `wc_http_cert.arm`
   – Location: `c:\ibm\http\ssl`

6. Close the IBM Key Management Utility.

Certificate trust security configuration

This section describe how to configure the IBM Secure Sockets Extension for WebSphere security and how to trust certificates between the WebSphere Portal node and WebSphere Commerce node.

1. Enable the IBM Secure Sockets Extension API by editing the security properties file on the WebSphere Portal node:

   `<WAS_HOME>/java/jre/lib/security/java.security`

   Verify that the following line exists and add it if necessary:

   `security.provider.`*`x`*`=com.ibm.jsse.JSSEProvider`

Where *x* is the sequence number (for example, 2 or 3).

2. Restart WebSphere Application Server Administrative Server (not needed if java.security was not modified) on the WebSphere Portal node.
3. Import the IBM HTTP Server certificate from WebSphere Commerce node to the WebSphere Portal Server.
   a. Copy the `wc_http_cert.arm` file from the WebSphere Commerce node to the `<WAS_HOME>/java/jre/lib/security` directory on the WebSphere Portal node.
   b. Issue the following command from `<WAS_HOME>/java/jre/lib/security` directory:

      ```
      ..\..\bin\keytool -import -alias WCCERT -file wc_http_cert.arm -keystore
      cacerts
      ```

   c. This command will import the SSL certificate into the default `cacerts` keystore. The default password for the `cacerts` keystore is `changeit`. Alias identifies this certificate in the keystore and it can be any name you choose.

2.5.3 WebSphere Commerce configuration for SSO

To configure WebSphere Application Server on the WebSphere Commerce node and change the WebSphere Commerce instance configuration to support SSO, do the following:

- Enable WebSphere security/SSL for WebSphere Commerce.
- Modify WebSphere Commerce instance configuration.
- Modify ldapentry.xml file.
- Modify existing users for objectClass ePerson (optional).

Enable WebSphere security/SSL for WebSphere Commerce

To enable WebSphere security and SSO on the WebSphere Commerce node, do the following:

1. Copy the exported LTPA Key file from the WebSphere Portal node to the WebSphere Commerce node.

   For example, we copied the `c:\ibm\was\bin\wp_wasltpa.key` from the WebSphere Portal node to the `c:\ibm\was\bin` directory on the WebSphere Commerce node.
2. Start the WebSphere Administrative Console on the WebSphere Commerce node.
3. From the WebSphere Administrative Console menu bar, select **Console -> Security Center**.

4. Click the **General tab** and check **Enable security**.
5. Click the **Authentication** tab and choose **Lightweight Third Party Authentication (LTPA)**.
6. New options for LTPA setting will be available on the window. Specify the following LTPA settings:

   Note: Many of the options listed will have the proper value already set. We recommend that you carefully review each setting.

   – Token Expiration: `120`

     How many minutes can pass before a client using an LtpaToken must authenticate again in the Token Expiration field.

   – Ensure the **Enable Single Sign-On (SSO)** is checked. The Domain field will be enabled once Enable Single Sign-On is checked.

   – Domain: `itso.ral.ibm.com`

     Enter a DNS domain name in the Domain field. In our example we set this domain to `itso.ral.ibm.com`. This domain name is used when the HTTP cookie is created for SSO and determines the scope to which SSO applies.

   From the same page, select the **LDAP** radio button and enter the following for accessing the LDAP directory:

   – Security Server ID: `wasadmin`

   – Security Server Password: `<password>`

   – Host: `<fully_qualified_hostname_of_LDAP_server>`

     For example, we entered `wcldap1.itso.ral.ibm.com`.

   – Directory Type: Select **Custom**

   – Port: `389`

   – Base Distinguished Name: `dc=ibm,dc=com`

     Base entry DN where the users can be found.

   – Bind Distinguished Name: `cn=root`

     This is the SecureWay Directory user that will do the bind to the LDAP directory (used to rebind as an authenticated user in the DMT).

   – Bind Password: `<password>`

   After entering these values, the window should look like Figure 2-8 on page 79. Do **not** click Apply or OK just yet.

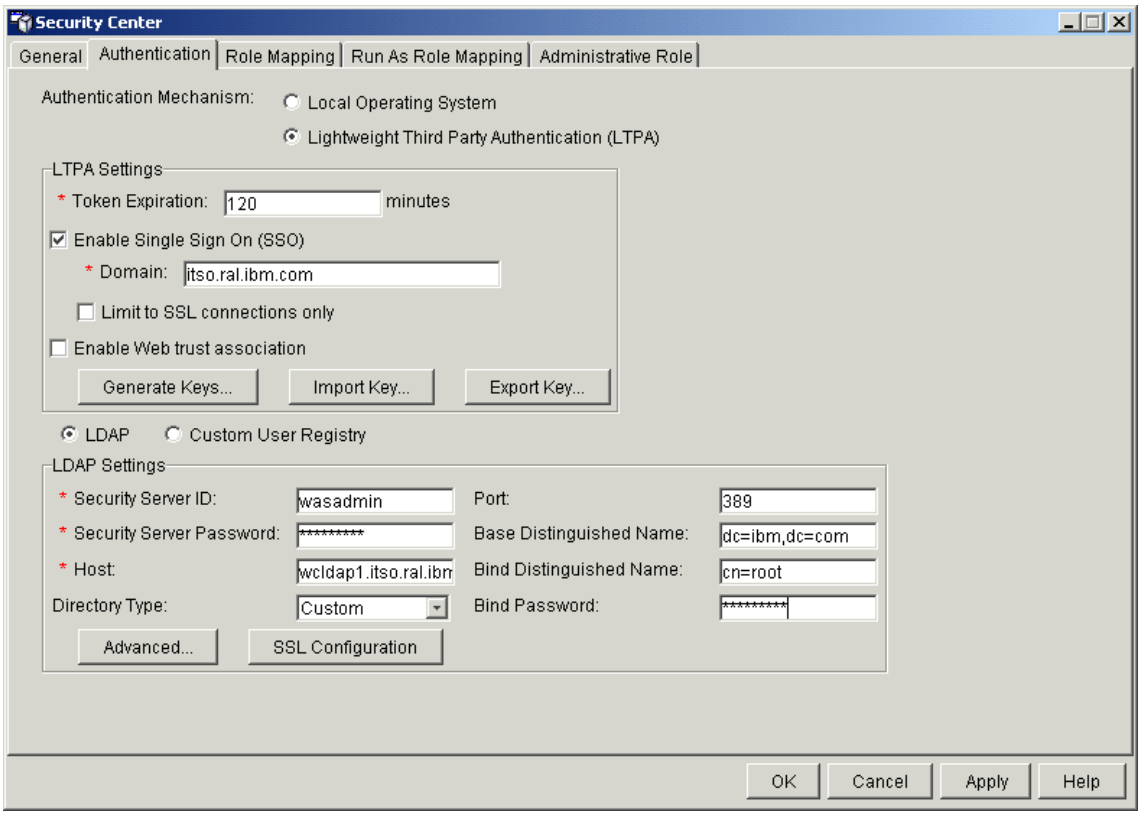


Figure 2-8 WebSphere security SSO and LDAP settings for the WebSphere Commerce node

7. Click **Advanced** tab, enter the following and then click **OK** (see Figure 2-9 on page 80):
   - Initial JNDI Context Factory: `com.sun.jndi.ldap.LdapCtxFactory` (default)
   - User Filter: `(&(uid=%v)(objectclass=inetOrgPerson))`
   - Group Filter: `(&(cn=%v)(objectclass=groupOfUniqueNames))`
   - User ID Map: `*:uid`
   - Group ID Map: `*:cn`
   - Group Member ID Map: `groupOfUniqueNames:uniqueMember`
   - Certificate Mapping: `Exact Distinguished Name`

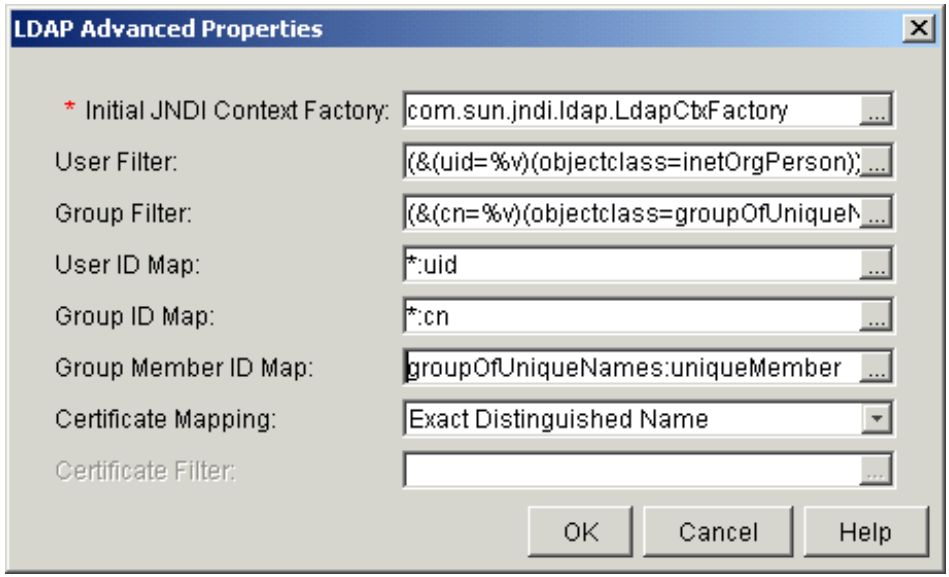


Figure 2-9 LDAP Advanced Properties

8. From the same page, click **Import Key...**
   - Select the Key file copied from the WebSphere Commerce node. In our example, imported the `c:\ibm\was\bin\wp_wasltpa.key` copied above from WebSphere Portal node to the WebSphere Commerce node.
   - Type the same password entered when the key was generated (in our example, `wpsbind`). Reenter the password and click **OK**.
9. Click **OK** to accept security settings.
10. Restart the WebSphere Administrative Server for the security changes to take effect before proceeding (adminserver.bat).
11. Start the WebSphere Administrative Console on the WebSphere Commerce node. Logon using the Security Server ID `wasadmin`.
12. From the WebSphere Administrative Console menu bar, select **Console -> Security Center**.
13. Click the **Role Mapping** tab.
    a. Select the enterprise application for the WebSphere Commerce instance (for example, WebSphere Commerce Enterprise Application - wc1).
    b. Click **Edit Mappings...**.
    c. Under Roles, select **WCSecurityRole**, and then click **Select**.
    d. Check **Select users/groups** checkbox.

e. Enter `wasadmin` in the search field, and then click **Search**.

f. Select the **cn=wasadmin, dc=ibm,dc=com** under Available Users/Groups, and then click **Add**. The result of this action should look like Figure 2-10.

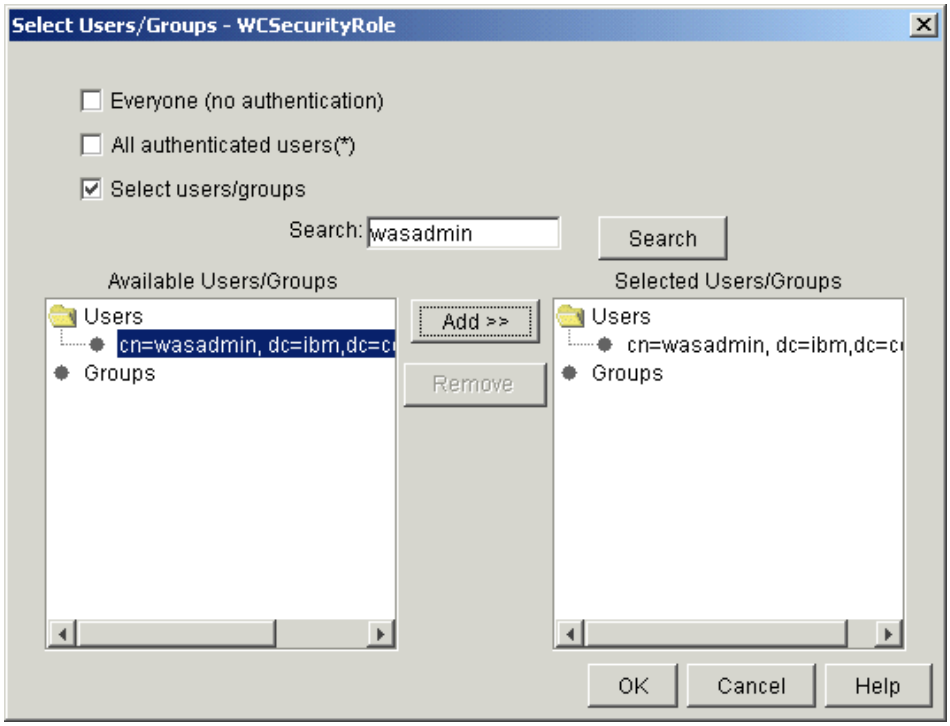


Figure 2-10 Select Users/Groups - WCSecurityRole

g. Click **OK** for the next two pop-up screens.

h. Click **Apply** and then **OK**.

Modify WebSphere Commerce instance configuration

Configure the WebSphere Commerce instance to use WebSphere Application Server security and enable SSO by completing following steps:

1. We will need to temporarily start the WebSphere Administrative Server from Windows services so that the WC Configuration Manager recognizes the Administrative Server is started. Ensure you have stopped the server if running via adminserver.bat.
2. Start the WC Configuration Manager Server Windows service.

3. Start the WebSphere Commerce Configuration Manager. Enter the user ID `webadmin` and <password> when prompted.

   Note: Remember to start the IBM WC Configuration Manager Server Windows service prior to launching the application.

4. Select **WebSphere Commerce -> <node-name> -> Instance List -> <instance_name> -> Instance Properties -> Security**.
5. Check the **Enable Security** check box.
6. A message window advises you that WebSphere Application Server security has to be set up before enabling security. Click **Yes**, assuming you have configured this option by following this procedure.
7. Select **LDAP User Registry** and enter the user ID and password of the WebSphere Security Server ID (for example, `wasadmin`) and then click **Apply**.
8. When the message `Successfully configured security for WebSphere Commerce` appears, click **OK**.
9. Select **WebSphere Commerce -> <node-name> -> Instance List -> <instance_name> -> Instance Properties -> MemberSystem**. In the right pane, check the **Single sign-on** check box and click **Apply** (should already be set).
10. Close the WebSphere Commerce Configuration Manager.
11. Modify the <wc_instance>.xml file.
    a. Using a text editor, open the WebSphere Commerce instance configuration file <instance_name>.xml located in the <wc_home>/instances/<instance_name>/xml directory.
    b. Locate the tag `<SessionManagement>`.
    c. Under the tag `<cookie >` add the attribute `sslauth=”false”`.

       Note: It is our understanding that this value is used to accept the LtpaToken properly when it comes from a non-SSL page and vice versa.

       The resulting code should look like Example 2-4.

Example 2-4 Sample <wc_instance>.xml for security

```
<SessionManagement name=”Session Management">
      <url-rewriting display="false"
                     enabled="false" />
      <cookie acceptance="false"
```

```
                age="-1"
                display="false"
                domain=""
                enabled="true"
                path="/"
                sslauth="false"
                persistence="wcs" />
</SessionManagement>
```

d. Save the file and close the text editor.

Modify ldapentry.xml file

To implement SSO, the entries for users in the LDAP directory must include the objectclass ePerson. In the default mapping file (ldapentry.xml), the entries created by WebSphere Commerce member subsystem do not include the ePerson objectclass.

To add the objectclass ePerson, complete the following steps:

1. Open the file ldapentry.xml with a text editor.

   The ldapentry.xml file should be located in <wc_home>/xml/ldap/.

2. Locate the tag <ldapocs> and add the objectclass ePerson.

   ```
   <ldapocs objClass="top;person;organizationalPerson;inetOrgPerson;ePerson"/>
   ```

Important: Only add the ePerson objectclass for the entryType=”User” (do not add to Organization or OrganizationalUnit).

3. Save the changes and close the text editor.
4. Restart the WebSphere Commerce <instance> application server.

Modify existing users for objectClass ePerson (optional)

From this moment on, all new users registered will be created including the *objectclass ePerson*, but all users already registered in need of SSO with the WebSphere Commerce node, will need to be updated in the LDAP directory.

Note: For our example, this is not necessary. The only users we have created that do not have the objectClass ePerson are wpsadmin and wpsbind, neither of which require single sign-on with the WebSphere Commerce node.

To modify the existing users, complete the following steps on the Directory Server node:

1. Before proceeding, we recommend that you backup the SecureWay Directory database (for example, LDAPDB2) containing directory information. Refer to , “Backing up a DB2 database” on page 259.
2. Open the IBM SecureWay Directory Web Administration tool.
3. Logon as cn=root.
4. Select **Database -> Export LDIF**.
5. Specify a file name and export all the entries (we accepted the default c:\<swd_home>\var\ldap\export.ldif). Click **Export**.
6. Using a text editor, open the exported LDIF file and add the following line to all the User entries:

   ```
   objectclass: ePerson
   ```

7. Save the changes and close the text editor.
8. Using the DMT, rebind and connect as an authenticated user, `cn=root`.
9. Backup the LDAP database LDAPDB2. Refer to, “Backing up a DB2 database” on page 259 for details.
10. Select **Browse tree** and delete all the entries under `the DN` that are going to be updated.
11. Use the IBM SecureWay Directory Web Administration tool to import the modified entries. Select **Database -> Import LDIF** and type the name of the modified LDIF file and then click **Import**. For example, `c:\<swd_home>\var\ldap\export.ldif`.

    You may have to click **Clear Results** prior to the import if not displayed.

    Depending on how many users you have updated/deleted, you will get an error message saying that the update was not successful but it will tell you how many entries were updated.
12. Refresh the tree in DMT and verify that the new entries now include the ePerson objectclass. Expand the tree and double-click one of the modified entries. Expand the list of values for the objectclass (Object Class) field. The ePerson value should now be included in the list.

Is this the best way to update the object class?

An additional class can be added using the DMT. The problem is you have to do it one by one, and it is probably better to use LDIF files for large amounts of entries to update.

2.5.4 Verify single sign-on (SSO) configuration

To test the implementation, we must connect to the sample application, authenticate, and then move to WebSphere Commerce without being prompted for a password.

To test the SSO enablement, complete the following steps:

1. Ensure the following are started on each of the nodes:
   - WebSphere Portal node:
     - DB2
     - IBM HTTP Server
     - IBM WS AdminServer 4.0
     - WebSphere Portal application server
   - Directory Server node:
     - DB2
     - IBM HTTP Server
     - IBM SecureWay Directory V3.2.2
   - WebSphere Commerce node:
     - DB2
     - IBM HTTP Server
     - WebSphere Administrative Server (adminserver.bat)
     - WebSphere Commerce Payments application server + IBMPayServer.bat
     - WebSphere Commerce <instance> application server
2. Close all Web browser windows opened. Launch a new Web browser window and type the URL for the WebSphere Portal:

   `http://wcportal1.itso.ral.ibm.com/wps/portal`
3. Logon to the WebSphere Portal with the `wcsadmin` user, and then click **Submit Login**.

   Important: The selected user entry in the LDAP directory must have been updated to include the objectClass ePerson as described in, “Modify existing users for objectClass ePerson (optional)” on page 83.
4. On the WebSphere Commerce node, start the Default Server.
5. In the same Web browser window, enter the URL for showCfg sample application on the WebSphere Commerce node:

   `http://<wc_hostname>/webapp/examples/showCfg`

Note: If you have already published a WebSphere Commerce store, you can enter the store URL:

For example:

```
http://<wc_hostname>/webapp/wcs/stores/servlet/tooltech/index.jsp
```

Congratulations! You have now configured the WebSphere Commerce node, WebSphere Portal node, and Directory Server node for single sign-on.

2.6 Deploy the ITSO B2B CEP store

Refer to 5.1, “Deploy ITSO B2B CEP store to runtime environment” on page 208 for details on deploying the ITSO B2B CEP store.

2.7 DB2 Server node implementation

This section describes the high level steps to install the DB2 Database Server node within the ITSO test environment in preparation for hosting databases for the WebSphere Application Server respository, WebSphere Commerce instance database, WebSphere Commerce Payments database, and WebSphere Portal database.

Note: More detailed information can be found in the redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567, and the product installation guides.

The high level installation and configuration procedures for the DB2 Server node are as follows:

- Windows 2000 Server installation
- DB2 Server installation
- Verify the DB2 instance TCP/IP connection port
- Migrate WebSphere Commerce databases to remote DB2 server

2.7.1 Windows 2000 Server installation

In preparation for the installation of WebSphere Commerce and supporting components, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.

2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Create a token object
   - Increase quotas
   - Log on as a service
   - Replace a process level token
3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

2.7.2 DB2 Server installation

This section highlights the key steps for installing and configuring the DB2 Server for use with WebSphere Application Server and WebSphere Commerce.

The DB2 Server installation is organized as follows:

- Install DB2 UDB V7.2 Enterprise Edition.
- Install DB2 V7 FixPak 7 (7.1.0.68).
- Update JDBC level to JDBC2.

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- Product Guide *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*

Install DB2 UDB V7.2 Enterprise Edition

The high level steps to install the IBM DB2 UDB V7.2, Enterprise Edition, are as follows:

1. Insert the IBM DB2 UDB V7.2, Enterprise Edition CD and run Setup.
2. We accepted the default options unless noted as follows:
   - Select **DB2 Enterprise Edition** and **DB2 Application Development Client**.
   - Select **Custom** for the installation type. Refer to the noted documentation for details.
   - We installed DB2 to the c:\ibm\sqllib directory.
   - Create an instance.

Install DB2 V7 FixPak 7 (7.1.0.68)

We installed DB2 V7 FixPak 7 (7.1.0.68), which can be downloaded at:

ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/db2ntv7/FP7_WR21311/

Update JDBC level to JDBC2

Update the JDBC level to JDBC2 as follows:

1. Stop all DB2 Windows services.
2. Run usejdbc2.bat found in the <DB2_HOME>\java12 directory.
3. The inuse file found <DB2_HOME>\java12 directory should state the following:

   JDBC 2.0

4. To verify the JDBC functionality (optional).

   We have included an IBM site where a JDBC test tool and instructions (jdbctest.java, jsread2.html) for verification can be downloaded and run on your system. Enter the following ftp link in a Web browser:

   ftp://ftp.software.ibm.com/software/websphere/info/tools/jdbctest

2.7.3 Verify the DB2 instance TCP/IP connection port

To verify the DB2 instance TCP/IP connection port, do the following:

1. On the DB2 Server node, execute the following command in DB2 command Window:

   db2 get dbm cfg

2. In the output of the command, search for the following as seen in Figure 2-11 on page 89:

   TCP/IP Service Name (SVCENAME) =

   This is the DB2 instance connection port. If instead of a number you see a name such as db2cDB2, look up the corresponding port number for this name in C:\WINNT\system32\drivers\etc\services file. (On AIX, Solaris and Linux, this file is called /etc/services). This port number or service name will be needed during the DB2 client configuration for Commerce Application Server node (WebSphere Application Server, WebSphere Commerce, WebSphere Commerce Payments), and the WebSphere Portal node.

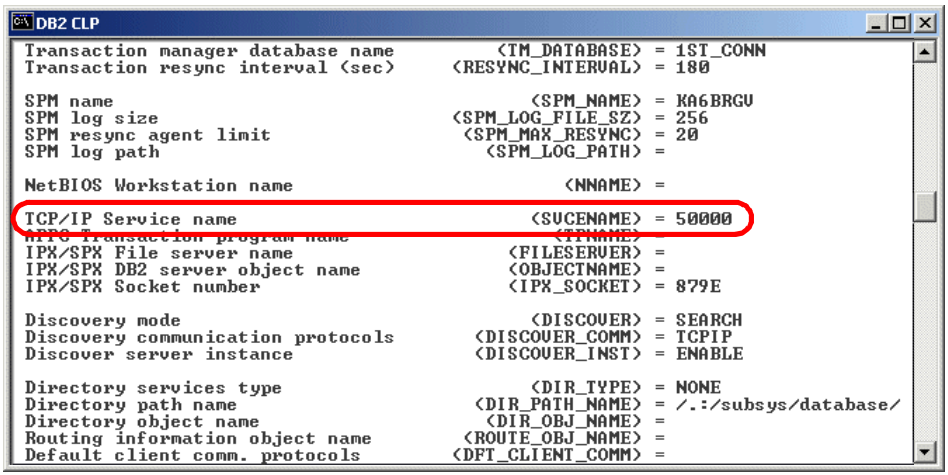


Figure 2-11 DB2 instance TCP/IP connection port

2.7.4 Migrate WebSphere Commerce databases to remote DB2 server

In our example, we installed the DB2 Server on the Commerce Application Server node. Now that we have verified that the node is working, we can move the application databases to the remote DB2 server. This step is optional.

Note: There are many ways to configure this type of environment. It is possible that the DB2 client and server be configured from the start. We chose this approach to demonstrate how to migrate databases and to avoid problems with the IBM Commerce Enhancement Pack - October 2002 Edition installer when using a remote database server node.

The DB2 Server includes the components of the DB2 client. For this reason, we do not have to install the DB2 client. The procedure documented will explain how to migrate application databases (WebSphere Application Server, WebSphere Commerce and WebSphere Commerce Payments databases) and configure the DB2 client on the Commerce Application Server to communicate with the remote DB2 database server.

Note: For detailed installation instructions, refer to the following:

- Redbook *WebSphere Commerce V5.4 Handbook*, SG24-6567
- Product Guide *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*

Backup the application databases

For information on the backup of DB2 databases refer to, “Backing up a DB2 database” on page 259.

Restore the application databases

For information on the restore of DB2 databases refer to, “Restoring a DB2 database” on page 259.

Configure and verify the DB2 client/server connectivity

This section describes how to configure the DB2 client and server and verify that they are communicating properly.

Catalog the TCP/IP node

From a DB2 command window, type the following command:

Syntax:

```
db2 catalog tcpip node <node_name> remote <server_name> server
<port_number>
```

The `<port_number>` is the DB2 instance connection port found on the server in the services file. Alternatively, in place of the port number a service name can be used. If a service name is used, the port and service name must be added to the DB2 client system services file so that it can resolve where to find the system.

Example:

```
db2 catalog tcpip node wcdb2 remote wcdb2 server 50000
```

Attach to the remote DB2 Server

From a DB2 command window, type the following command:

Syntax:

```
db2 attach to <node_name> user <db2_username> using <db2_user_password>
```

Example:

```
db2 attach to wcdb2 user db2admin using <password>
```

Catalog the databases

Once you have attached to the remote DB2 Server, catalog the databases from a DB2 command window:

```
db2 catalog db <db_name> at node <node_name>
```

Note: When migrating the databases from the Commerce Application Server node to the DB2 Database Server node, you will need to catalog the following databases:

- WebSphere Application Server repository database
- WebSphere Commerce instance database
- WebSphere Commerce Payments database

Verify the runtime environment

After the databases have been restored to the remote DB2 server, verify that the WebSphere Commerce runtime environment is working properly before proceeding with a standard store.

2.7.5 Migrate WebSphere Portal database to remote DB2 server

After everything has been verified, the WebSphere Portal database can be migrated to the remote DB2 Server. Refer to 2.7.4, “Migrate WebSphere Commerce databases to remote DB2 server” on page 89 for general guidelines.

Note: This step is optional, if you are not using a remote DB2 Server for your runtime configuration.

- Create a new instance on the DB2 Server.
- In this case you will configure the connection from the WebSphere Portal node to the remote DB2 Server.
- Next backup the databases and restore them on the remote DB2.
- Catalog the databases on the WebSphere Portal node.
- Verify the runtime environment.

2.8 Remote Web server node implementation

After a commerce enabled portal store has been verified using the WebSphere Portal node, Commerce Application Server node, and Directory node, we added a remote IBM HTTP Server.

Note: For detailed information, refer to the following:

- *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *IBM WebSphere V4.0 Advanced Edition Handbook*, SG24-6176
- *IBM WebSphere V4.0 Advanced Edition Scalability*, SG24-6192

3

Implement the development environment

When developing a commerce enabled portal application, it is important to consider how the various application components and development tools can be configured to provide a rich end-to-end development environment. At the core of an end-to-end development environment is the ability to develop the application code, deploy, test and debug.

This chapter provides an explanation of the development environment that we used to develop, deploy, test and debug the application samples. The chapter also provides detailed procedures about how to implement the development environment and debug commerce enable portal applications. We have devised and documented many unique procedures and techniques to provide source level debug and test capability.

The chapter is organized into the following sections:

- Plan for an end-to-end development environment
- WebSphere Portal development test node
- Development tools node

3.1 Plan for an end-to-end development environment

This sections describes an end-to-end approach for developing and testing commerce enabled portal solutions, and includes the following topics:

- Motivation for an end-to-end development environment
- Solution overview for an end-to-end development environment
- Development environment configurations
- Hardware and software used in the development environment

3.1.1 Motivation for an end-to-end development environment

In general a customer develops WebSphere Commerce applications using the recommended development tools, such as VisualAge for Java and WebSphere Studio, which are provided in WebSphere Commerce Studio V5.4. VisualAge for Java allows a WebSphere Commerce developer to debug the entire site, including Java code and JavaServer Pages. The JSP debugger is a powerful tool to step through the generation process of the generated markup language.

WebSphere Portal solutions are developed using the WebSphere Studio Application Developer. Due to some restrictions within WebSphere Studio Application Developer a portal developer is only able to debug a portal application when the WebSphere Application Server Advanced Single Server Edition is installed. For the WebSphere Application Server Advanced Edition the code must be deployed on the runtime environment with no debug support.

Note: At the time of writing this Redpaper, IBM WebSphere Commerce V5.4 did not officially support WebSphere Studio Application Developer.

The following capabilities are desired when developing a commerce enabled portal solution:

- An end-to-end development and test scenario to develop, test and debug commerce enabled portals including WebSphere Commerce code.
- Leverage the full VisualAge for Java development environment capabilities to develop and test stores including HTML, commerce portlet JSP and WAP WML JSPs within the VisualAge for Java WebSphere Test Environment.
- Leverage the full WebSphere Application Developer Studio to develop, test and debug portlets.
- Simulate single sign-on to leverage the Commerce Enhancement Pack functionality.

Note: The WebSphere Test Environment of VisualAge for Java does not support WebSphere security settings and single sign-on services.

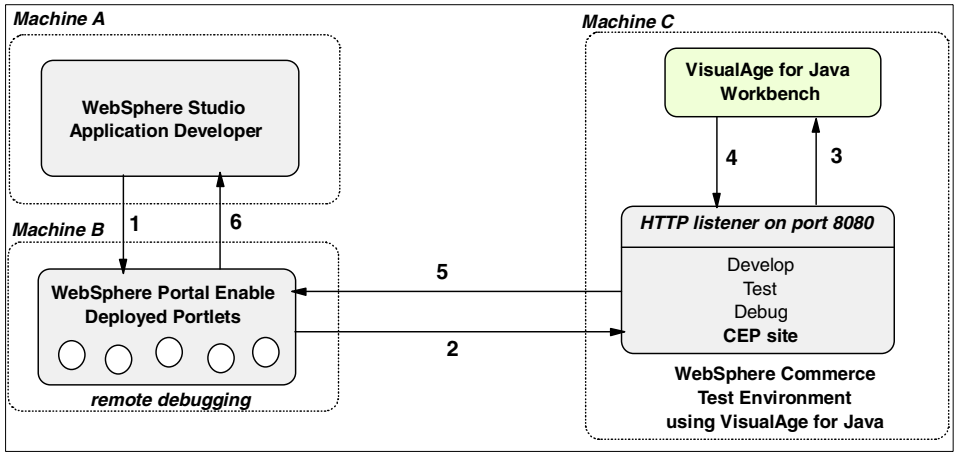


Figure 3-1 End-to-end interaction flow

Figure 3-1 illustrates the interaction flow in the commerce enabled portal end-to-end development solution and summarizes the following actions:

1. The portal developer on `Machine A` debugs the portal site using WebSphere Studio Application Developer by using hooks into a commerce portlet for debugging on `Machine B`.

   Note: The components of `Machine A` and `Machine B` can be installed on the same node.

2. The portlet communicates with the WebSphere Test Environment of VisualAge for Java while sending a HTTP URL controller command to port 8080.
3. The WebSphere Test Environment receives the portlet request. The environment is configured to stop if a request is recognized. The debugger is started and the WebSphere Commerce developer is now able to debug the code. The JSP debugger can be used by starting the tool and setting break points in advance of the request for the JSP.
4. The URL controller command redirects to a JavaServer Page which renders the appropriate markup language. The JSP debugger can be used to follow the generation process.

5. When the JSP compilation and execution completes the response it passed to the portlet.
6. The portal developer debugs the WebSphere Commerce response and controls the portal rendering.

3.1.2 Solution overview for an end-to-end development environment

This section provides detailed information for identifying which changes are necessary to achieve an end-to-end test and debug of commerce enabled portal solutions. A brief overview of the solution is displayed in Figure 3-2.

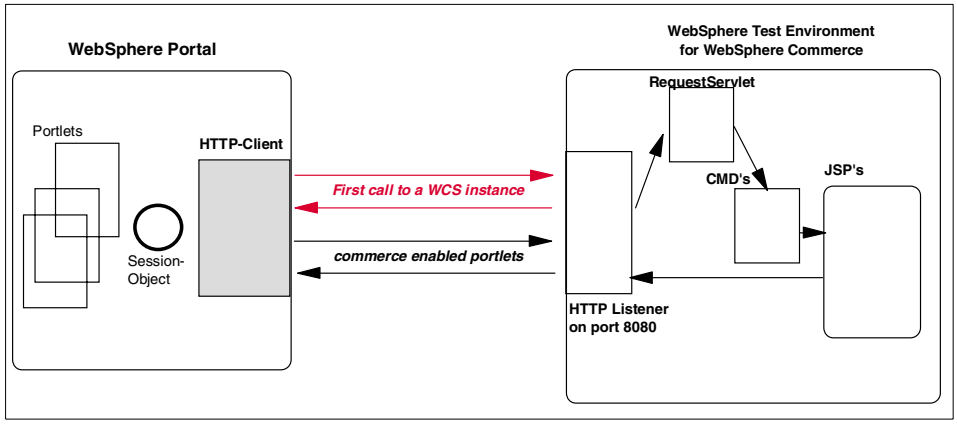


Figure 3-2 Enabling end-to-end testing

WebSphere Commerce

The developer is able to develop customized code and JavaServer Pages provided by WebSphere Commerce Studio. After installing and configuring the Commerce Enhancement Pack a developer is now able to develop commerce enabled JavaServer Pages as well as business logic to serve HTML/WML portal request. The WebSphere Test Environment remains unchanged.

Security and single sign-on services are not supported within the test environment but can be simulated. Therefore, the WebSphere Commerce developer has to create users within the WebSphere Commerce environment. In general the developer uses the WebSphere Commerce Administrator Console.

User authentication is based on the WebSphere Commerce services. WebSphere Commerce offers the `Logon` URL controller command to authenticate a user by the designated `logonId` and `logonPassword`. The `Logon` command creates a set of WebSphere Commerce cookies which authenticates/identifies the user.

The WebSphere Commerce `Logon` URL controller command is listed in Example 3-1 is one line.

Example 3-1 Logon URL command

```
http://<hostname>:8080/webapp/wcs/stores/servlet/Logon?logonId=wcsadmin&logonPa
ssword=password0&URL=http://host_name:8080/webapp/wcs/stores/dummy.html&storeId
=10051&reLogonURL=http://host_name:8080/webapp/wcs/stores/dummy.html
```

WebSphere Portal

A portal developer uses WebSphere Studio Application Developer to develop and test portlets. In order to test commerce enabled portlets a WebSphere Portal environment must be configured properly with the Commerce Enhancement Pack while disabling the single sign-on. Since some WebSphere Commerce URL controller commands force user authentication a `Logon` URL controller command is required to authenticate the portal user.

Note: For each WebSphere Portal user a corresponding WebSphere Commerce user must be created with the same user ID and password settings.

Whenever a WebSphere Portal user uses a commerce enabled portal application the instance data is associated to the user session. The commerce enabled portal solution manages a `SessionObject` which is shared by all commerce portlets of the WebSphere Portal user. This `SessionObject` provides services to maintain session relevant data to issue a WebSphere Commerce http request. The commerce portlet internally sends a WebSphere Commerce http requests using the HttpClient.

To authenticate a WebSphere Portal user with WebSphere Commerce we have to send once and only once a `Logon` URL controller command to fetch the WebSphere Commerce cookies. Once we received the cookies they are handled by the `SessionObject` as part of the commerce enabled portal solution.

Commerce enabled portlets share the user-based cookies to send authenticated WebSphere Commerce http requests.

Summary

The approach presented enables developers with an environment to develop, test and debug WebSphere Portal and WebSphere Commerce applications in a commerce enabled portal solution. The advantages are as follows:

- Simulation of single sign-on services without changing the commerce enabled portal functionality included in the IBM Commerce Enhancement Pack - October 2002 Edition.
- End-to-end commerce enabled portal development and test.
- Develop, test and debug portal HTML/WAP WML JSPs and business logic using VisualAge for Java.
- Develop, test and debug portlets using WebSphere Application Developer Studio.

3.1.3 Development environment configurations

There are many different possibilities for configuring the development environment components. We have described four different configurations that highlight the key issues and limitations:

- Single-node development environment
- Multi-node development environment separate portal runtime
- Multi-node development environment by developer type
- Multi-node team development environment
- Team runtime environment for testing

Front-end development only requires WebSphere Studio Application Developer

If you are only developing portlets and Web assets such as JSPs (standard or portlet JSPs) you do not need to install and configure WebSphere Commerce Studio (VisualAge for Java and WebSphere Studio). For front-end development, only WebSphere Studio Application Developer is needed for debug you will still need to run VisualAge for Java WebSphere Test Environment.

Single-node development environment

As the name implies, all development tools and unit test runtime environment components are installed on a single node. The advantage to this type of configuration is that it is self contained and makes deployment of the development assets to the runtime environment easier. The biggest disadvantage is the vast memory requirement (2 GB recommended, 1.5 GB minimum).

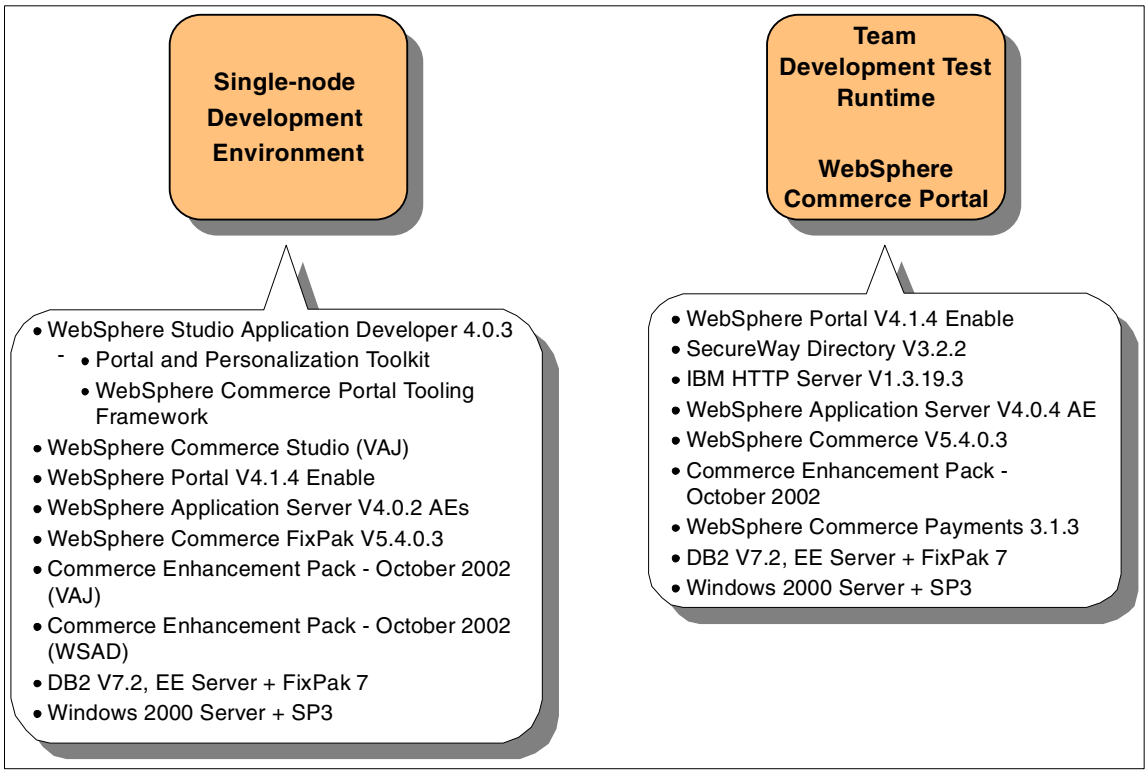


Figure 3-3 Single-node development environment

Figure 3-3 depicts a single-node development environment where all development tools and runtime components are installed on the same node.

Multi-node development environment separate portal runtime

In this development environment configuration, the development tooling and unit test runtime environment components are installed on separate nodes. The advantage of this type of environment is reduced system memory requirements for each node. The disadvantage is that you will now need more than one system for the development environment. Also, the deployment of application assets being developed becomes slightly more complicated.

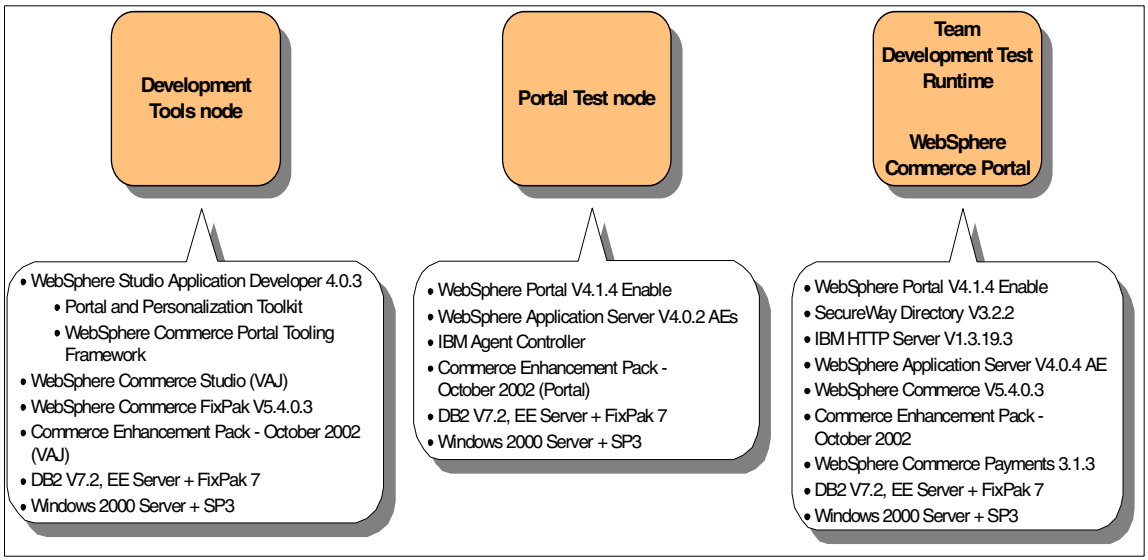


Figure 3-4 Multi-node development environment separate portal runtime

Figure 3-4 depicts a multi-node development environment with separate nodes for development tools and the WebSphere Portal and DB2 Server runtime environment.

Multi-node development environment by developer type

A medium to large size project will very often have developers focused on a particular task or more specialized. For example, you may have separate developers for portal and commerce. To support this type of development team, we recommend the approach depicted in Figure 3-5 on page 101. This type of multi-node development environment is separated by the type of developer creating the application assets (WebSphere Commerce and WebSphere Portal).

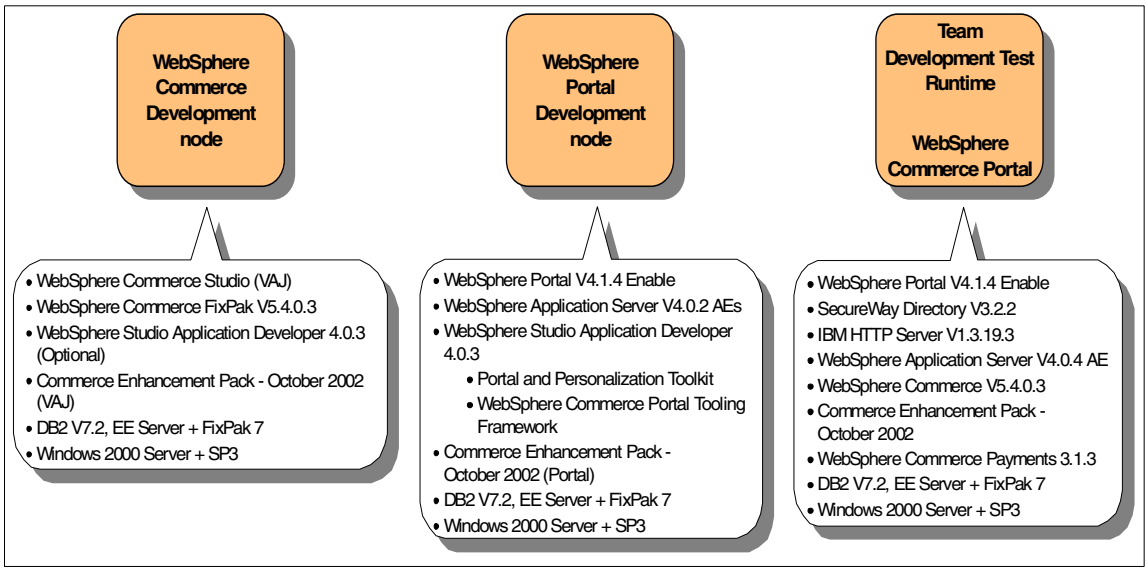


Figure 3-5 Multi-node development environment by developer type

Multi-node team development environment

Any project that more than a few developers working on the same code needs a versioning scheme, often addressed by a tool, in order to keep the code coordinated among the programmers. The larger the project, the greater the need to have a source control system so that the developers do not change the same section of code simultaneously, and so that consistency, levels and correctness in the code are preserved.

For this purpose IBM VisualAge for Java ships with an integrated team development environment based on a shared repository. This repository is managed by the team server (EMSRV). Using shared databases located on a remote database server also reduces resource utilization on the developer's workstations.

WebSphere Studio Application Developer is a file based, integrated development environment (IDE). It maintains a local workspace where all the project data is stored in files. The history of resource changes cached, allowing the developer to compare and replace resources with earlier versions. WebSphere Studio Application Developer provides an open architecture that provides plug-in support for different Software Configuration Management systems (SCMs). The team development model of the workbench enables the use of pluggable third party repositories rather than mandating the use of a proprietary repository. Application Developer delivers adapters for two SCM products, Rational® ClearCase® LT and Concurrent Versions System (CVS).

When involved in a project with team development, it is important to integrate and test the components of the team in a test runtime environment. This enviornment is also needed to test function that may not have been able to be tested on the developer's workstation such as single sign-on.

Separate repository server, which ran EMSRV team server. This is where the common repository for all our Java code.

Separate version controlling server like Rational CrearCase for all assets developed in WebSphere Studio Application Developer.

Each developer has their own WebSphere Commerce development environment or WebSphere Portal development environment set up on their workstation. Developers can develop the code on their workstation and fully test it before deploying it to the test and staging server (see Figure 3-6).

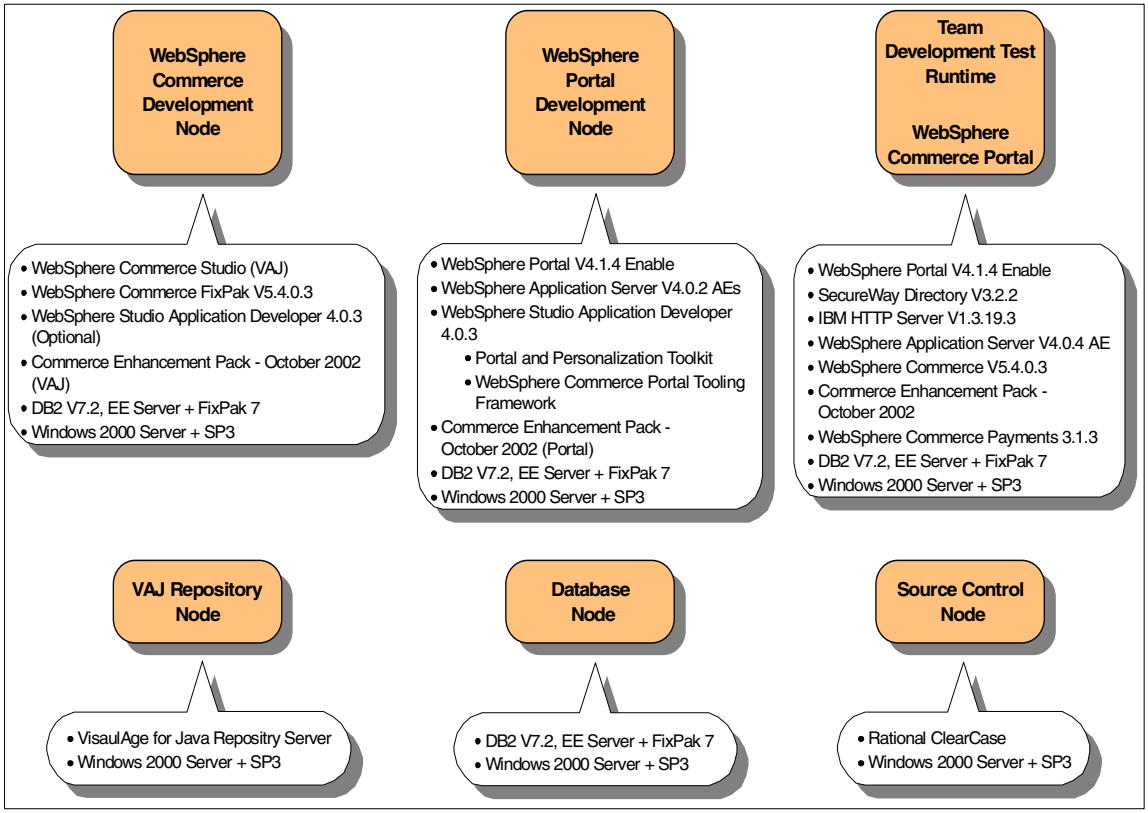


Figure 3-6 Multi-node team development environment

Team runtime environment for testing

After the unit testing has been completed within the previously described development environments, we recommend that you deploy and test your commerce enabled portal in a fully functional commerce enabled portal runtime environment that includes a directory server and single sign-on. This is important for several reasons. First, there are certain features developed that can not be tested until they are deployed in the full runtime environment. Second, a team runtime environment server promotes working out integration issues with software created by other developers.

For details on implementing a commerce enabled portal runtime environment, refer to Chapter 2, “Implement the runtime environment” on page 7.

3.1.4 Hardware and software used in the development environment

This section describes the hardware and software used by the ITSO for the Multi-node development environment separate portal runtime development environment described in, “Multi-node development environment separate portal runtime” on page 99.

Hardware used for the ITSO development environment

Depending on the capability and capacity of your development hardware, all components can be installed on one node. In our example, we installed the development environment software on two separate nodes.

Development tools node

We used the following hardware for the development tools node:

- IBM NetVista™ PC (6792-MHU)
  - 1 CPU, Intel P4 1.8 Ghz
  - 1 GB Memory
  - 40 GB Hard Disk
  - Intel(R) PRO/100+ PCI Adapter

WebSphere Portal development node

We used the following hardware for the WebSphere Portal development node (VisualAge for Java):

- IBM NetVista PC (6792-MHU)
  - 1 CPU, Intel P4 1.8 Ghz
  - 1 GB Memory
  - 40 GB Hard Disk
  - Intel(R) PRO/100+ PCI Adapter

Software used for the ITSO development environment

There are many possible configurations for a WebSphere Commerce Portal development environment. Table 3-1 lists the components needed for a WebSphere Commerce Portals development environment and the memory used by each component. The development environment includes a unit test runtime environment where the development assets can be tested as well as application development tools.

Table 3-1 Component summary memory usage

Software component	Version	Memory usage
Windows 2000	Service Pack 3	64 MB **Note**: This will vary depending on the installation options selected.
WebSphere Application Server V4 Advanced Edition * WebSphere Commerce <instance> application server * WebSphere Commerce Payments application server * WebSphere Portal Server application server * WebSphere Admin server	V4.0.4	 384 MB 128 MB 256 MB 100 MB
DB2 UDB V7.2 Enterprise Edition + FixPak 7 (Server)	V7.1.0.68	30 MB each for LDAP, WPS, WAS for portal, MALL, WAS for WC, Payment, PNS
DB2 UDB V7.2 Enterprise Edition + FixPak 7 (Client) Note: Used for remote database connectivity.	V7.1.0.68	
IBM HTTP Server V1.3.19 + WebSphere Application Server V4 FixPak 4 (V4.0.4)	V1.3.19.3	10 MB
WebSphere Commerce V5.4 * WebSphere Commerce FixPak V5.4.0.3 * IBM Commerce Enhancement Pack - October 2002 Edition	V5.4.0.3	
WebSphere Commerce Payments * WebSphere Payment Manager V3.1.2 * WebSphere Commerce Payments FixPak V3.1.3	V3.1.3	
IBM SecureWay Directory	V3.2.2	20 BM
WebSphere Portal V4 Enable	V4.1.4	

Software component	Version	Memory usage
WebSphere Studio Application Developer * WebSphere Portal Toolkit V4.1.4 * WebSphere Commerce Portal Toolkit (IBM Commerce Enhancement Pack - October 2002 Edition) * WebSphere Personalization Toolkit V4.1.4 * IBM Agent Controller (installed by default with WebSphere Studio Application Developer)	V4.0.3	400 MB
IBM Agent Controller (installed on WebSphere Application Server)	V4.0.3	
WebSphere Commerce Studio V5.4 (includes VisualAge for Java V4 EE, WebSphere Studio V4). * WebSphere Commerce FixPak V5.4.0.3 * IBM Commerce Enhancement Pack - October 2002 Edition	V5.4.0.3	300 MB 64 MB

3.2 WebSphere Portal development test node

This section describes the components installed and configured on the WebSphere Portal development test node used to deploy, unit test, and debug portlets. This node also includes a DB2 server to host the VisualAge for Java WebSphere Test Environment database (PNS) and the WebSphere Commerce instance database.

The high level procedure to install and configure the WebSphere Portal development test node are as follows:

1. Windows 2000 Server installation
2. DB2 Server installation
3. WebSphere Application Server AEs installation
4. WebSphere Personalization installation
5. WebSphere Portal Enable installation
6. WebSphere Portal debug ID configuration
7. IBM Agent Controller installation (WSAD)

Note: For more detailed information on configuring the components installed on the WebSphere Portal test node, refer to the following:

- *WebSphere Portal V4.1, Windows 2000 Installation,* REDP3593
- *WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883
- *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *WebSphere Portal V4.1 Developers Handbook*, SG24-6897

3.2.1 Windows 2000 Server installation

In preparation for the installation of WebSphere Commerce and supporting components, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.
2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Create a token object
   - Increase quotas
   - Log on as a service
   - Replace a process level token
3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

3.2.2 DB2 Server installation

This section describes the steps needed to install and configure the DB2 Server in ITSO development environment. We will install DB2 Server on the WebSphere Portal development test node to be used later to host the remote WebSphere Commerce instance database and for WebSphere Portal.

Install DB2 V7.2 Server

Refer to, “Install DB2 UDB V7.2 Enterprise Edition” on page 16.

Install DB2 V7.2 FixPak 7 (7.1.0.68)

Refer to, “Install DB2 V7 FixPak 7 (7.1.0.68)” on page 16.

Update the JDBC level to JDBC 2

Refer to, “Update JDBC level to JDBC2” on page 16.

Note: You must ensure that you have stopped all DB2 services prior to switching over to JDK 1.2. If this is not done properly, the WebSphere Portal installation will fail.

3.2.3 WebSphere Application Server AEs installation

This section describes the steps needed to install WebSphere Application Server V4.0.2, Advanced Single Server Edition (AEs). The debug environment in WebSphere Studio Application Developer only supports WebSphere Application

Server Advanced Single Server Edition. This section will explain how to install and configure of WebSphere Application Server Advanced Single Server Edition.

Note: The WebSphere Application Server Advanced Single Server Edition does not use a DB2 database to store configuration data.

The section is organized as follows:

- WebSphere Application Server V4.0.1 AEs installation
- WebSphere Application Server V4 AEs FixPak 2 installation
- WebSphere Application Server V4.0.2 AEs e-Fix installation
- WebSphere Application Server AEs verification

WebSphere Application Server V4.0.1 AEs installation

Using WebSphere Application Server Advanced Single Server Edition instead of WebSphere Application Server Advanced Edition posses some limitations:

- The current version of WebSphere Portal (V4.1.4) only supports the installation of WebSphere Portal Enable on WebSphere Application Server Advanced Single Server as it installs out of the box.
- Using WebSphere Application Server Advanced Single Server Edition also has some limitations with respect to testing security. Single Server Edition does not include WebSphere security. The login for WebSphere Application Server Advanced Single Server Edition does not allow for the use of an LDAP and single sign-on. Therefore testing applications that use single sign-on, such as the Commerce Enhancement Pack commerce enabled portals posses a challenge as it requires single sign-on.

The ITSO has devised and documented a workaround to address these limitations. For an overview of the solution refer to, 3.1, “Plan for an end-to-end development environment” on page 94.

To install WebSphere Application Server AEs, do the following:

1. From the WebSphere Application Server AEs CD, navigate to the <CD_Drive>\was\win directory and run Setup.
2. We accepted the default installation options, unless noted below.
   - Select **Custom Installation**
   - Select only the following components:
     - Application Server
     - Application and Development Tools
     - IBM JDK 1.3.0

Note: These are the components that are required for WebSphere Portal to install and function properly. An HTTP Server is not required and the samples are not used. The WebSphere Application Server Advanced Single Server Edition uses and internal HTTP service for development.

– We installed to the `c:\WebSphere\AppServer` directory. Take note of the install directory. This is needed later for the WebSphere Portal Toolkit configuration.

Important: Advanced Edition and Single Server Edition on same node?

If you already have installed WebSphere Application Server Advanced Edition on your machine and you want both to run on your machine. If it is the case then you must specify a different install directory for WebSphere Application Server Advanced Single Server Edition than WebSphere Application Server Advanced Edition installed.

For detailed information on installing WebSphere Application Server Advanced Single Server Edition and WebSphere Application Server Advanced Edition on the same machine see Running Co-Existing installation of WebSphere Application Server in WebSphere Application Server InfoCenter.

3. Select **No, I will restart my computer later** and click **OK** to complete the installation.

WebSphere Application Server V4 AEs FixPak 2 installation

After installation of WebSphere Application Server V4.0.1, Advanced Single Server Edition, we must install WebSphere Application Server FixPak 2 as required by WebSphere Portal. FixPak 2 will update both WebSphere Application Server and the bundled JDK.

Note: In the runtime environment of this Redpaper, as documented in Chapter 2, “Implement the runtime environment” on page 7, we installed WebSphere Application Server FixPak 4 as required by WebSphere Portal Enable V4.1.4. At the time of writing this Redpaper, the most current WebSphere Application Server FixPak level supported for the development environment using the WebSphere Application Server Advanced Single Server Edition is FixPak 2.

To install WebSphere Application Server V4 AEs FixPak 2, do the following:

1. Insert CD 3-3 from WebSphere Portal CDs into the machine where WebSphere Application Server Advanced Single Server Edition is installed.
2. Copy the <CD_Drive>\was\win\fixpack2 directory to a temporary directory on your hard drive. FixPak 2 needs to be able to write to the hard disk during the installation and cannot run directly from the CD.
3. Change to the temporary folder where FixPak2 has been copied and run install.bat.
4. We accepted the default installation options.

WebSphere Application Server V4.0.2 AEs e-Fix installation

After installing the WebSphere Application Server FixPak 2, install the e-Fix as follows:

1. Insert CD 3-3 from WebSphere Portal CDs into the machine where WebSphere Application Server Advanced Single Server Edition installed.
2. Copy the e-Fix from the following directory on the CD to a temporary directory.

   ```
   <CD_DRIVE>\was\eFixes\PQ56615_eFix_AEServer_AEsServer.jar
   ```

3. Set the path to include Java from a command window:

   ```
   set path=%path%;c:\ibm\wasaes\java\bin
   ```

4. Run the following command to install the e-Fix:

   ```
   java -jar <temp>\PQ56615_eFix_AEServer_AEsServer.jar
   ```

5. When prompted, enter the target directory where WebSphere Application Server is installed (for example, c:\ibm\wasaes) and press **Enter**.
6. Verify there were no errors or warnings on the screen. If needed, retry the operation after resolving the problem.
   a. Verify a backup of the e-Fix has been created:

      ```
      <WAS_HOME>\e-fix\<efix#>
      ```

   b. Verify the product.xml file has been updated:

      ```
      <WAS_HOME>\properties\com\ibm\websphere\product.xml
      ```

WebSphere Application Server AEs verification

After installing WebSphere Application Server Advanced Single Server Edition, FixPak 2and e-Fix, we recommend that you validate the installation before continuing with installation.

1. By default WebSphere Application Server Advanced Single Server Edition does not run as window service. So start the server by selecting **Start -> Programs -> IBM WebSphere -> Application Server AEs -> Start**

Application Server. It will be started in a command windows as shown in Figure 3-7. Do not close the window or it will stop the application server.

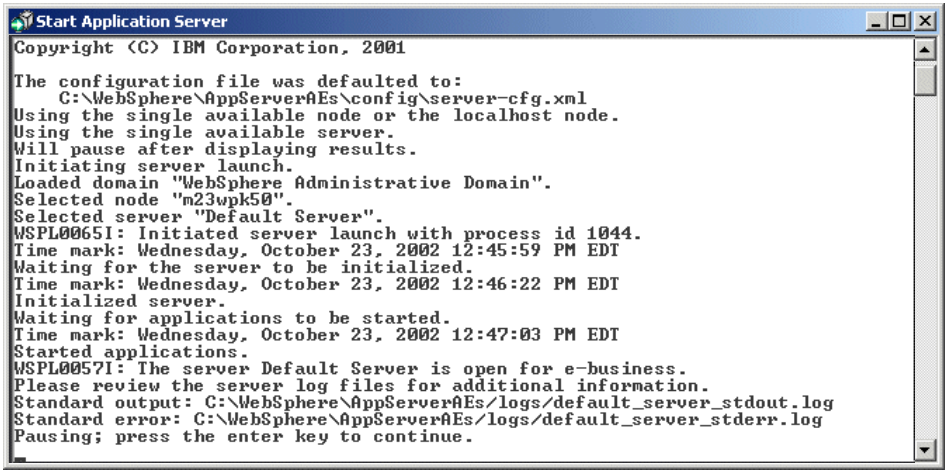


Figure 3-7 WebSphere Application Server AEs startup window

2. After the Application server start process is fully complete, open a Web browser window an enter the following URL:

   `http://<hostname>:9080/servlet/snoop`

   If you see the snoop servlet appeared as Figure 3-8 on page 111, this confirms that WebSphere Application Server Advanced Single Server Edition has been installed correctly.

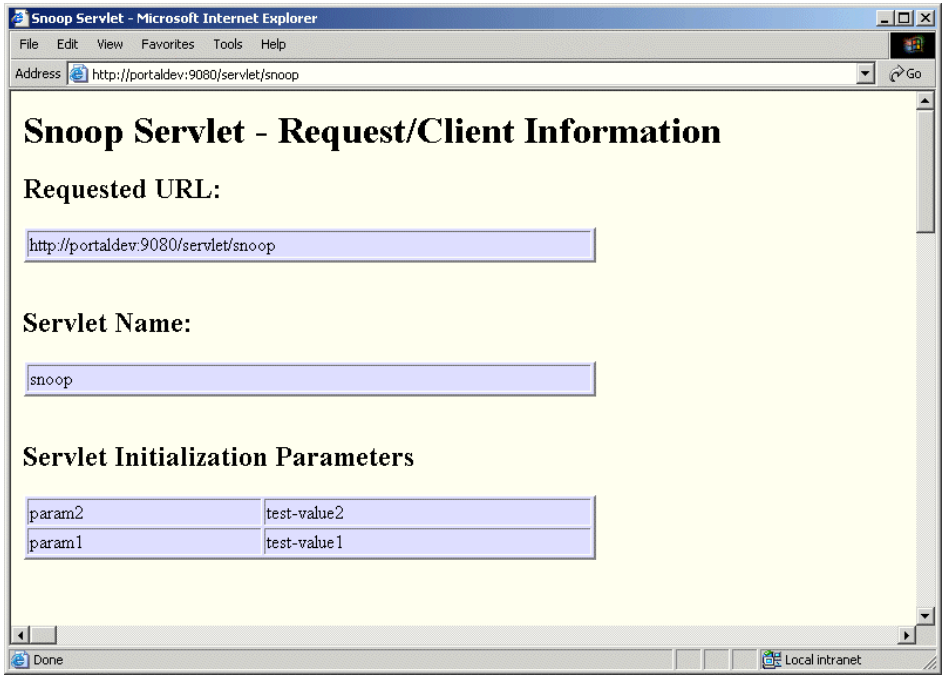


Figure 3-8 Verify WebSphere Application Server AEs - snoop servlet

3.2.4 WebSphere Personalization installation

Before installing WebSphere Portal, two JAR files from the WebSphere Personalization, `prCommon.jar` and `personalization.jar` are required to be deployed on the WebSphere Application Server.

The current version of WebSphere Personalization will not install on WebSphere Application Server Advanced Single Server (WAS AEs), so we must copy the two necessary personalization JARs manually in order to have WebSphere Portal function properly.

To manually install the WebSphere Personalization JARs, do the following:

1. Insert CD 3-3 from WebSphere Portal CDs into the machine where WebSphere Application Server Advanced Single Server Edition is installed.
2. Copy the following two files from `<CD Drive>\external\personalization` to `<WAS _HOME>\lib\app` directory:

   ```
   personalization.jar
   prCommon.jar
   ```

3.2.5 WebSphere Portal Enable installation

This section describes the steps needed to install WebSphere Portal in a WebSphere Application Server V4.0.2, Advanced Single Server Edition environment.

The section is organized as follows:

- WebSphere Portal Enable V4.1.4 installation
- WebSphere Portal install verification
- WebSphere Portal debug ID configuration

WebSphere Portal Enable V4.1.4 installation

WebSphere Portal Enable Version 4.1.4 must be installed on the same machine as WebSphere Application Server Advanced Single Server Edition. The installation will create a new XML file called `WebSpherePortal-cfg.xml` in the <WAS_HOME>\config directory. This file will be used later during the configuration of the WebSphere Portal Toolkit.

To install WebSphere Portal, do the following:

1. Ensure the WebSphere Application Server Administrative Server is running.

   Run the **StartServer.bat** batch file. The WebSphere Portal Server installation checks to see if WebSphere Application Server is started during the installation. Starting WebSphere Application Server now will help avoid any issues during the installation. Alternatively, start the server by selecting **Start -> Programs -> IBM WebSphere -> Application Server AEs -> Start Application Server**.

2. Insert CD 7 from WebSphere Portal CDs into the machine where WebSphere Application Server Advanced Single Server Edition installed.
3. Navigate to the <CD_Drive>\wps directory and run the **install.bat** to begin the WebSphere Portal installation process.
4. Click **Next** when prompted for the Welcome and Prerequisites screens.
5. After the install program checks to see if WebSphere Application Server is running, accept the alert message and click **Next**.

Note: This checker will show the following message, which can be ignored:

“Some prerequisites are missing. If you continue, the portal might not work correctly after installation. WebSphere Application Server edition: detected AEs, required is AE”.

6. Accept the default WebSphere Portal install path or select your own by typing or clicking the **Browse** button and then click **Next**. We installed to the c:\ibm\PortalServer directory.
7. Select **Standard** option in install type selection screen and click **Next**.
8. Enter the WebSphere Node name of the machine and click **Next**.

   This value should be automatically entered for you. You can also find this information by typing `hostname` from a command prompt and is typically your computer name under the Network Identification tab in Windows 2000. If you want to verify what WebSphere Application Server is using for the node name, open the <WAS_HOME>\config\server-cfg.xml file and search for "Node_1".
9. Enter the complete hostname of the machine along with port 9080 (for example, myserver.ibm.com:9080) and the base URI (/wps) then click **Next**.

   Note: The hostname is the name that WebSphere Portal will use when dynamically generating the URLs to access the portal. It is important that the hostname can be resolved remotely if WebSphere Studio Application Developer is installed on a separate machine. It does not have to be a fully qualified domain name, however; you must be able to access WebSphere Application Server from a browser using whatever name you specify here from the both the remote and local system.

   It is important that you use :9080 at the end of the hostname so that a HTTP server does not have to be installed. Port 9080 is the default port used by WebSphere Application Server as the internal HTTP transport. Record the hostname entered here as it must be the same name used during the WebSphere Portal Toolkit configuration. If they are different, the automatic login function will not work properly when debugging portlets.
10. Use the defaults for Home page and Customized page (/portal and /myportal) and click **Next**.
11. Enter a Proxy server hostname and Proxy port (if needed) and click **Next**. The only reason you would need a Proxy server is if you access Web sites through your portlets that are outside a firewall. Usually this is not the case in development and text environment. We left this blank in our environment.
12. Select the **Deploy base portlets into WebSphere Portal Server** option and click **Next**.

    The base portlets are not required for testing your portlet, however; this process also updates the default portlet skins information which is required for the portlet to be displayed on the Web page. Therefore, deploying the base portlets is the easiest way to complete this task.

13. Select the **IBM DB2** option as your database and click **Next**.

14. Select the **Create and initialize an existing database** and click **Next**. It is best to create a new database for use with the Portal Toolkit to avoid any conflicts with existing portal data or security configurations (see Figure 3-9).

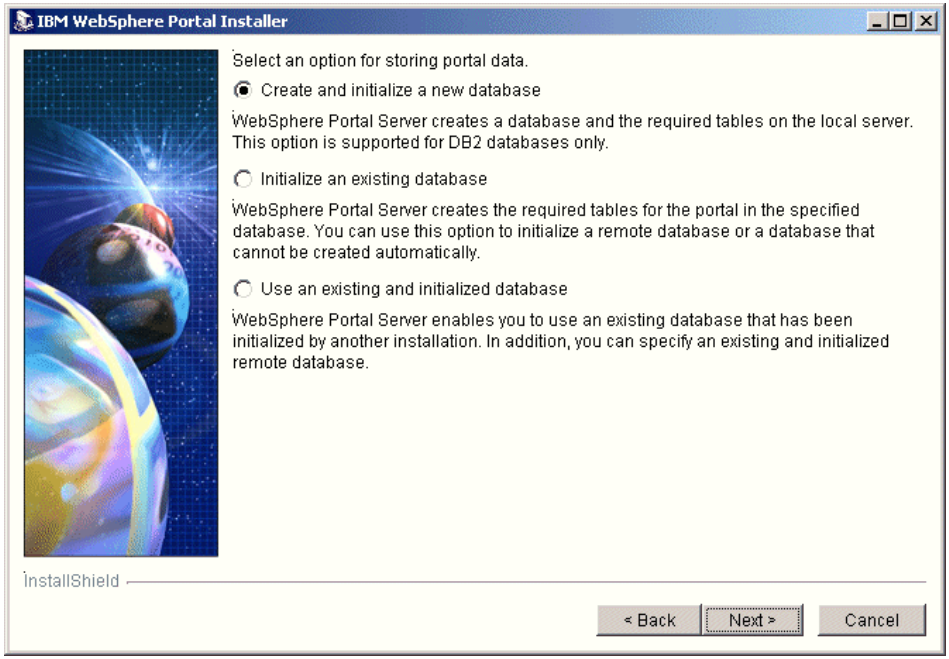


Figure 3-9 Option for storing portal data

15. Enter the database administrator username and password, and database name, and click **Next**. Use the same username and password that you used when IBM DB2 was installed. We entered `WPS41Dev` as seen in Figure 3-10 on page 115. The database will be dropped by the installation program if it already exists so make sure the database name is correct and not already being used before continuing.

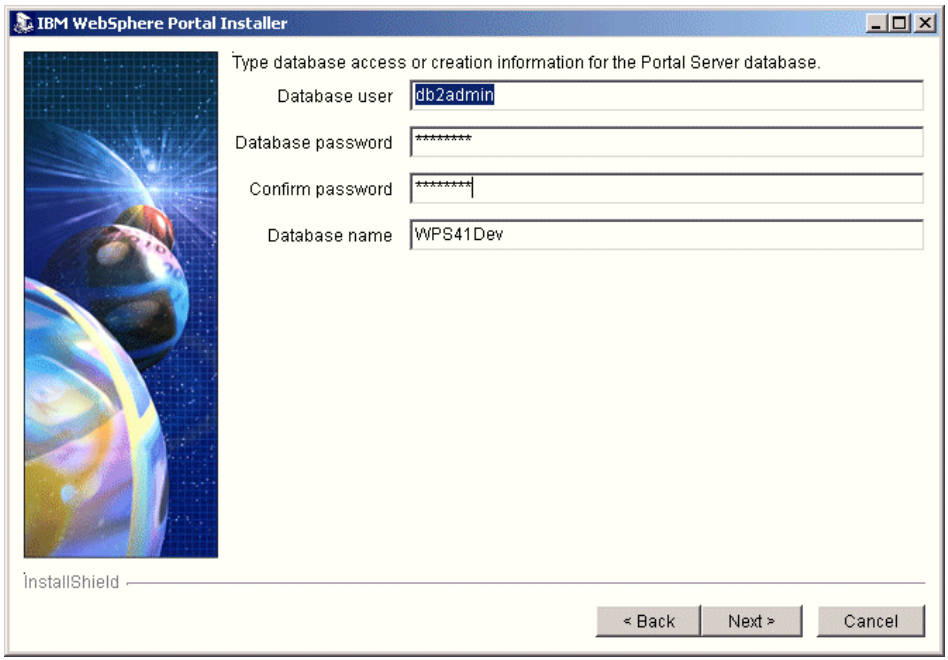


Figure 3-10 Portal database access

16. Enter the full path to the DB2 JDBC driver and click **Next**.

    This file is typically located in the directory where DB2 is installed (for example, `c:\ibm\sqllib\java\db2java.zip`). Ensure that DB2 is using JDBC2 as explained in, “Update the JDBC level to JDBC 2” on page 106.

    If setup cannot locate the file, it will ask you for the correct location again and will not continue.

17. Select the **Create and initialize a new database** option for Member Services and click **Next**.

18. Verify the database administrator username and password along with a member service database name and click **Next**. We entered `WMSDev` as seen in Figure 3-11 on page 116.

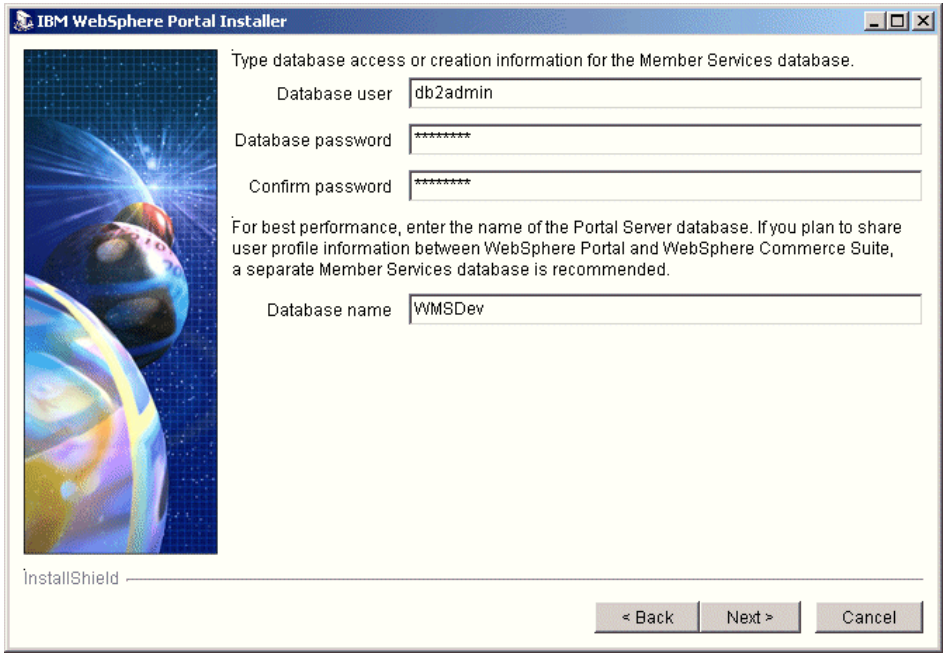


Figure 3-11 Member Services database access

Note: The database will be dropped by the installation program if it already exists so make sure the database name is correct and not already being used before continuing further.

19. Accept the default on the JDBC URL prefix page (`jdbc:db2`) and Click **Next**.
20. Click **Next** to start the installation process.

    The installation process will start copying the necessary files. You should see the message `Installing IBM WebSphere Portal Server` and a progress bar.
21. WebSphere Portal database will create during install process. When the database has been created successfully message display, review the message and click **Next**.
22. The WebSphere Portal database will be initialized. Review the message, and click **Next**.
23. The WebSphere Member Services database will be created. Review the message and click **Next** when prompted.
24. The WebSphere Member Services database will be initialized. Review the message and click **Next** when prompted.

25. Again the WebSphere Portal database will be updated based on the information created in the Member Services database. Review the message and click **Next** when prompted.
26. Now another update will process in WebSphere Member Services database. Click **Next** to continue while successfully done. You will see a window with the following message:

    ```
    "Wait while the application server is being configured".
    ```

27. At this point, three portal enterprise applications are deployed. This may take several minutes depending on your system configuration. Once the applications are deployed, click **Next**.

    Note: If this step stops after a few seconds, there may have been a problem with the installation. Review the error log, correct the problem, and restart the installation.

28. A series of backups are done prior to end of configuration process. Review the message and click **Next** when prompted.
29. The WebSphere Application Server is restarted with the new WebSphere Portal configuration by the installation program so that it can deploy portlets and make configuration modifications. Once WebSphere Application Server has restarted and the Next button is available, it is a good idea to verify that the base portal environment is configured properly, which can be verified as follows:
    a. Enter the following URL in the browser to make sure the EAR files were deployed properly. If there is a problem, the portlets will not deploy in the next step.

       ```
       http://<hostname>:9080/wps/portal
       ```

    b. After a few seconds, you should see the main portal screen with no page groups. If you do not see this screen then there was a problem with the installation of the EAR files and the installation process need to be start again.
30. Once the application server has been successfully restarted, click **Next** to deploy the portlets into WebSphere Portal.
31. The install process will check to make sure the application server is started. Once verified, click **Next** to proceed.
32. The installation process will now deploy the base portlets. At this point, there are approximately 229 configuration changes made to the database. This is the longest point during the configuration. It will take several minutes to complete. Click **Next** once the deployment has completed.

33. At this point the application server will stop and restart again. Click **Next** when prompt.
34. Click **Next** on the WebSphere Portal Final Installation Action and then click **Finish** to complete the installation.

WebSphere Portal install verification

After the WebSphere Portal installation, we recommend that you verify that the WebSphere Portal is working properly as follows:

1. Open a Web browser and enter the following URL:

   `http://<hostname>:9080/wps/portal`

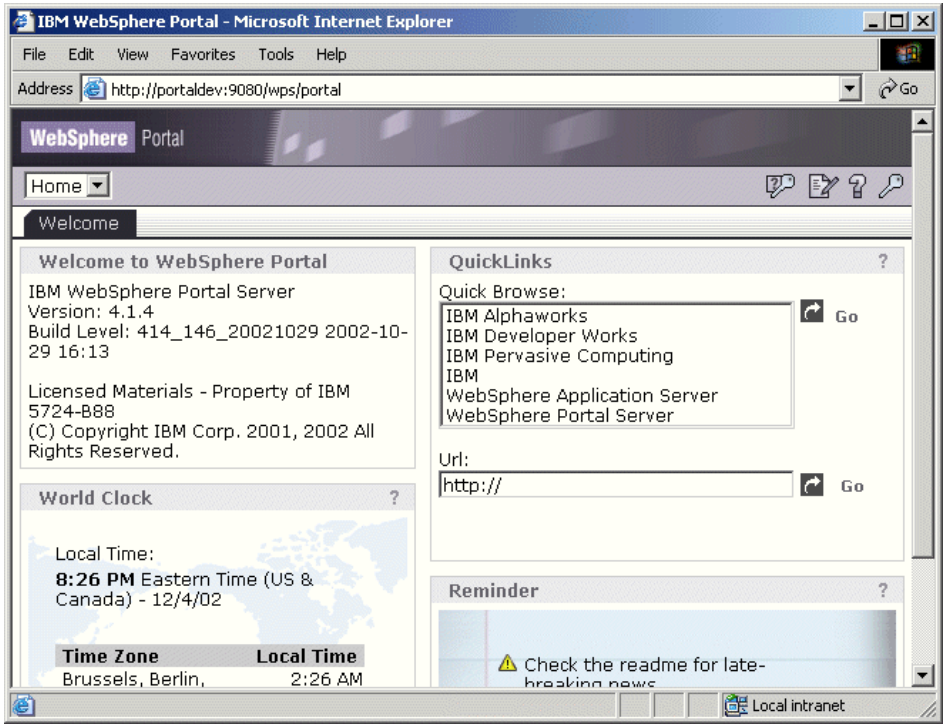


Figure 3-12 WebSphere Portal home page

2. Login by clicking the **Log in** icon  in the upper-right of the page. Enter the user ID `wpsadmin`, password `wpsadmin`, and then click **Log in** link at the bottom of the page.
3. After you have verified the log in, logoff by clicking by clicking the  icon in the upper left of the page.

3.2.6 WebSphere Portal debug ID configuration

In order to debug WebSphere Portal applications, we need to create a debug ID for use with the WebSphere Portal Toolkit. The debug ID is used by the Portal Toolkit to generate a page group and page for the user and deploy the developed portlets.

1. Start WebSphere Portal if it is not already started.

   If you have difficulties getting to the main Portal page, then stop and start WebSphere Application Server using these commands:

   ```
   stopServer.bat -configFile ..\config\WebSpherePortal-cfg.xml
   startServer.bat -configFile ..\config\WebSpherePortal-cfg.xml
   ```

2. Access the portal home, by entering the following URL:

   ```
   http://yourhost:9080/wps/portal
   ```

3. Click on the **self-registration** icon  in the top right corner of the page. On the resulting Welcome page, enter the following information to register the new user ID and then click **Continue** (see Figure 3-13 on page 120).

   – User ID: `wpsdebug`
   – Password: `wpsdebug`
   – First Name: `Wps`
   – Last Name: `Debug`

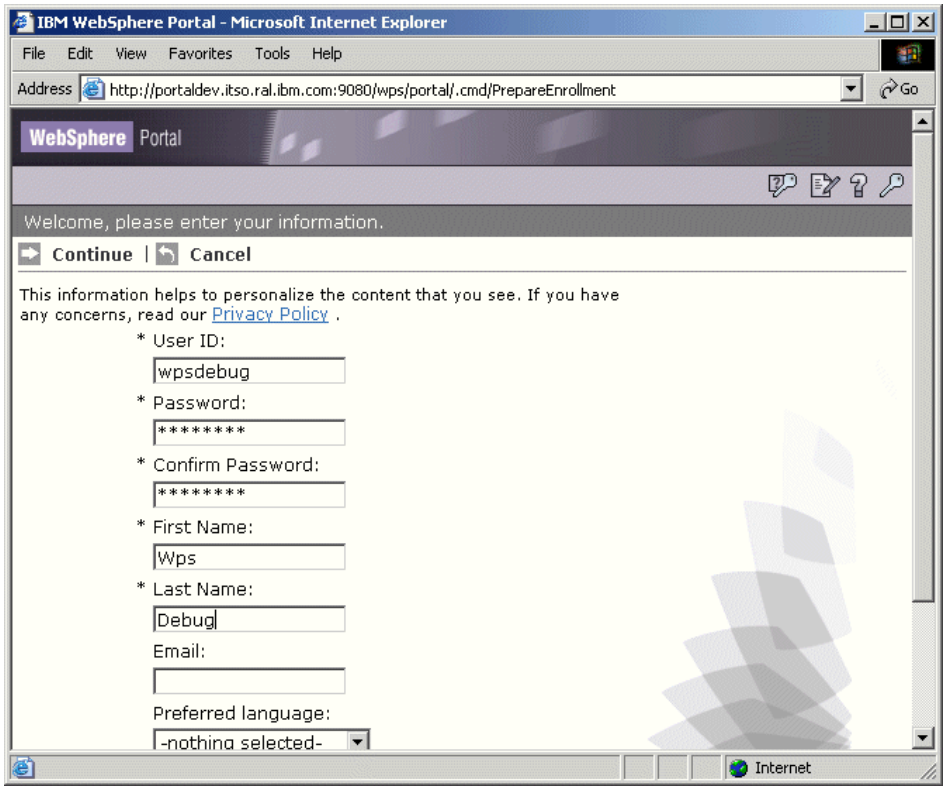


Figure 3-13 Debug user registration

4. When prompted, review the personal information and then click **Continue**.
5. You should see a message, `Your enrollment was successful`. Click **Continue**.
6. Stop WebSphere Application Server by entering the following command from the <WAS_HOME>\bin directory.

   ```
   stopServer.bat -configFile ..\config\WebSpherePortal-cfg.xml
   ```

7. Restart you system.

   During the configuration, a few environment variables have been set. It is necessary that the system be restarted before running WebSphere Studio Application Developer and the Portal Toolkit for the changes to take effect. If you do not restart, WebSphere Studio Application Developer will not start properly.

3.2.7 IBM Agent Controller installation (WSAD)

The IBM Agent Controller is a service used by WebSphere Studio Application Developer to remotely controll and debug the WebSphere Application Server applications on a remote machine.

Note: Only install the IBM Agent Controller if you have WebSphere Application Server installed on a separate machine from WebSphere Studio Application Developer. WebSphere Studio Application Developer automatically installs the IBM Agent Controller during it's installation.

1. Insert CD6-1 from the WebSphere Portal CDs into the machine where WebSphere Application Server is installed.
2. Navigate to the `<CD_drive>\wsad\win\RAC_install\windows` folder and run Setup.

   Note: If using a WebSphere Studio Application Developer CD, the path will be `<CD_drive>\RAC_install\windows`.

3. We accepted the default options, unless noted as follows:
   - We installed the IBM Agent Controller to: c:\ibm\IBM Agent Controller.
   - When prompted for the Java Home, click **Change**. Enter the path of the <WAS_HOME>\java and then click **Next** (for example, c:\ibm\wasaes\java).
   - Select **Complete installation** option and click **Next**.
   - Click **Install** to begin the installation.
4. Once installation is complete, click **Finish**.

The IBM Agent Controller service will be started automatically during the installation.

3.3 Development tools node

This section describes the components installed and configured on the development tools node used to develop, deploy, unit test, and debug commerce Java code (servlets, EJBs, JSPs), and commerce portlet JSPs. This node includes the VisualAge for Java WebSphere Test Environment, which the WebSphere Commerce Enterprise Application and store are deployed for unit test and debug purposes.

The high level procedure to install and configure the development tools node are as follows:

1. Windows 2000 Server installation
2. WebSphere Studio Application Developer installation
3. WebSphere Portal Toolkit installation
4. WebSphere Personalization Toolkit installation
5. WebSphere Studio Application Developer configuration for WebSphere Portal development
6. Testing the WebSphere Portal Server instance
7. DB2 installation
8. WebSphere Commerce Studio installation
9. VisualAge for Java configuration
10. WebSphere Commerce FixPak V5.4.0.3 installation
11. Commerce Enhancement Pack installation
12. Import the IBM Commerce Enhancement Pack repository
13. WebSphere Commerce Portal Tooling Framework
14. CEP installation on WebSphere Application Server AEs
15. Deploy the ITSO CEP B2B sample

Note: For detailed information on configuring the components installed on the development tools node, refer to the following:

- *WebSphere Commerce V5.4 Handbook*, SG24-6567
- *WebSphere Commerce V5.4 Developers Handbook*, SG24-6190
- *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*
- *Installation Guide, IBM WebSphere Commerce FixPak V5.4.0.3*
- *Getting Started, IBM Commerce Enhancement Pack*
- *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*

3.3.1 Windows 2000 Server installation

In preparation for the installation of WebSphere Commerce and supporting components, ensure the following tasks have been completed:

1. Install Windows 2000 Server and Windows 2000 Service Pack 3.

2. Ensure a administrator user is logged in for installation the of WebSphere Commerce and supporting software with the following user rights:
   - Act as part of the operating
   - Log on as a service
   - Create a token object
   - Increase quotas
   - Replace a process level token
3. Install Internet Explorer 5.5 and service pack or higher.
4. Verify the configuration of the TCP/IP network (hostname, IP address).

3.3.2 WebSphere Studio Application Developer installation

IBM WebSphere Studio Application Developer is an integrated development environment (IDE) for building, testing, and deploying J2EE applications. It provides integrated development support for building J2EE applications including HTML pages, servlets, JavaServer Pages, and EJBs. It can be used to create Web service applications with open standards, generate XML documents from DTDs, and enable a collaborative team development environment. WebSphere Studio Application Developer enables end-to-end local and remote testing and creates high-quality applications using wizards, code generators, and best practices.

We used WebSphere Studio Application Developer, WebSphere Portal Toolkit, and IBM WebSphere Commerce Portal Framework tools to develop commerce the commerce enabled portal.

To install WebSphere Studio Application Developer, do the following:

1. Ensure the IBM HTTP Server and the WebSphere Application Server Administrative Server are stopped.
2. Insert WebSphere Studio Application Developer V4.0.3 CD or CD6-1 from the WebSphere Portal CDs.
3. Navigate to the <CD-Drive>\wsad\win folder and run Setup.

   Note: The directory where setup is found may be different depending on the distriubtion media for WebSphere Studio Application Developer V4.0.3.

4. We accepted the default installation options unless noted as follows:
   - We installed to the `c:\ibm\wsad` directory.
   - We selected **J2EE Developer**.

– Select the version control interface you will be using. If currently you are not using CVS or Rational ClearCase then select **Other** options so that you can configure later (It only works if you install CVS later. After installation, if you decided to use Rational ClearCase then you have to uninstall WebSphere Studio Application Developer and reinstall with Rational ClearCase version control option). Click **Next**.

5. Click **Install** to install WebSphere Studio Application Developer to your Machine.
6. After installation is complete click **Finish** to complete the install process.

3.3.3 WebSphere Portal Toolkit installation

After the successful installation of WebSphere Studio Application Developer, install the WebSphere Portal Toolkit plug-in for WebSphere Studio Application Developer. The WebSphere Portal Toolkit is a plug-in designed to work with the WebSphere Studio Application Developer and provides the develop the tools and APIs for creating a portal application (portlets) or Portal Server instance and configuration.

To install the WebSphere Portal Toolkit, do the following:

1. Insert CD 3-3 from the WebSphere Portal CDs, run <CD_Drive>\PortalToolkit\install.bat to start the Portal Toolkit installation.
2. On the Welcome screen, click **Next** to begin the installation.
3. We accepted the default installation options.

   When the installation progress bar reaches 100% and “The InstallShield Wizard has successfully installed Portal Toolkit. Click Next to continue the wizard”, then click **Next** to continue.
4. Click **Finish** to complete the installation and exit the wizard.

3.3.4 WebSphere Personalization Toolkit installation

WebSphere Personalization Toolkit is similar to the WebSphere Portal Toolkit, used for developing personalization solutions. It is only required if will be developing personalziation rules and resources. The Personalization Toolkit installs as a plug-in for WebSphere Studio Application Developer and provides wizards for building personalization into your Web applications.

To install the Personalization Toolkit, do the following:

1. Insert CD 4 from the WebSphere Portal CDs into the same machine where WebSphere Studio Application Developer installed.

2. Navigate to the <CD_Drive>\personalization\nt directory and run **PznStudioDeveloperWizards** to start the Personalization Toolkit installation.
3. We accepted the default installation options.

   It should automatically detect the WebSphere Studio Application Developer installation directory according to your installation. Verify and if is not correct then enter the right path or browse through the **Browse** button where WebSphere Studio Application Developer has been installed. Click **Next**.
4. When the installation is complete, click **Finish** to close the wizard.

3.3.5 WebSphere Studio Application Developer configuration for WebSphere Portal development

At this point, we have already installed and configured all the base components to develop WebSphere Portal applications. Now we need to configure WebSphere Studio Application Developer for the purposes of debug and publishing to WebSphere Portal Server. In the ITSO development environment we used a two node approach where the WebSphere Portal runtime was on a remote node from the development node containing WebSphere Studio Application Developer and toolkits.

The following procedures describes how to configure the WebSphere Studio Application Developer environment for portal development, deployment, unit test and deploy in a two node environment.

1. Ensure you have installed and configured all the above mentioned components.
2. Ensure that the WebSphere Application Server Administrative Server is started on the remote WebSphere Portal runtime node.

   If not then open a command prompt on WebSphere Portal machine (WebSphere Portal development test node). Change directory to <WAS_HOME>\bin directory, and enter the following from the command line:

   ```
   startServer.bat -configFile ..\config\WebSpherePortal-cfg.xml
   ```
3. Verify that you can access to the portal by entering the following URL in a Web browser:

   ```
   http://<hostname>:9080/wps/portal
   ```

   Use wpsdebug as user id and password, which you have created in 3.2.6, “WebSphere Portal debug ID configuration” on page 119.
4. Set the access of the WebSphere Application Server directory on the WebSphere Portal development test node from WebSphere Studio Application Developer (Development Tools node).

If you are running all components on a single machine then you do not have to do this. In our example we are using a two nodes configuration, thus we will have to deploy portlets from WebSphere Studio Application Developer to the remote WebSphere Application Server machine (WebSphere Portal development test node). To make this possible, we need to set the access from the Development Tools node to the <WAS_HOME> directory on the WebSphere Portal development test node, where <WAS_HOME> is the WebSphere Application Server installed directory (for example, in ITSO this is C:\WebSphere\AppServer.

There are two basic options to achieve this:

a. Via FTP which required FTP server install and configured

b. Windows directory sharing.

   - Shared the directory where WebSphere Application Server Advanced Single Server Edition root directory resides.
   - Map the above shared directory on WebSphere Studio Application Developer machine.

For example, for the ITSO enviornment, the WebSphere Application Server Advanced Single Server Edition is installed on WebSphere Portal development test node under c:\ibm\. WebSphere Studio Application Developer installed on Development Tools node. So we make share C:\WebSphere\. On Development Tools node we mapped that shared folder as drive J. Now we will be able to access <WAS_HOME> by specifying J:\AppServer.

5. It is a best practice that the server-cfg.xml file and the deployed portlet application to be placed in a separate directory in order to keep the portal application and configuration separate from the original WebSphere Application Server and WebSphere Portal configuration.

   To do this, create a deployment directory for debugging under <WAS_HOME> directory. For example <WAS_HOME>\Debug. Here Debug is the new folder to deploy all newly developed or modified portlets.

6. Stop the WebSphere Application Server Administrative Server. In the future, WebSphere Studio Application Developer will start and stop the Application Server.

   Application server can be stopped as mentioned by issuing stopServer.bat. For example:

   ```
   stopServer.bat -configFile ..\config\WebSpherePortal-cfg.xml
   ```

7. Start WebSphere Studio Application Developer on the Development Tools node. From this point forward, the configuration will be performed on the Development Tools node.

To start WebSphere Studio Application Developer click **Start** -> **Programs -> IBM WebSphere Studio Application Developer -> IBM WebSphere Studio Application Developer**.

8. To create a Portal Server instance, do the following:

   a. From WebSphere Studio Application Developer menu, select **File -> New -> Other** as seen in Figure 3-14.

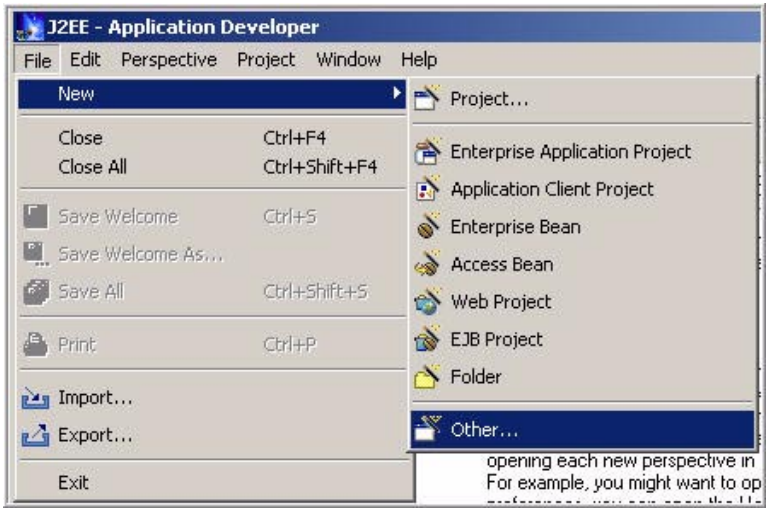


Figure 3-14 Create Portal Serve instance

   b. From the New window, select **Portal Server** on the left pane and **Portal server instance and configuration** on the right pane, and then click **Next** as seen in Figure 3-15.

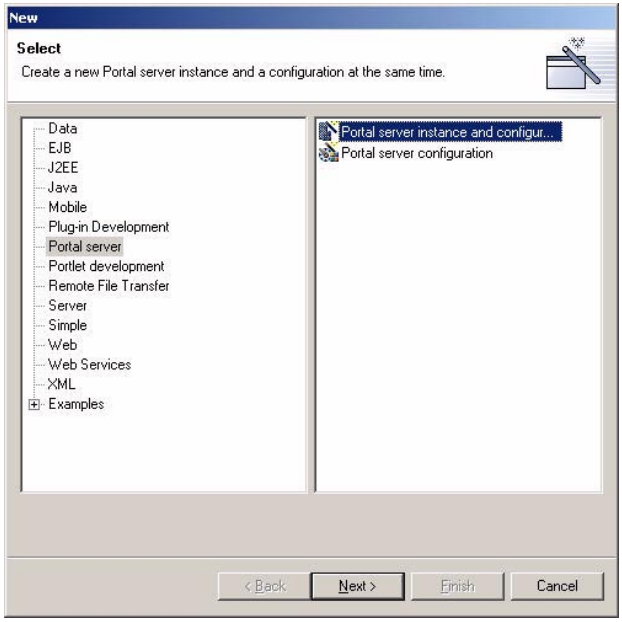


Figure 3-15 Portal Server window

c. On the WebSphere Portal Server setting window, set the following entries and click **Next** as seen in Figure 3-16.

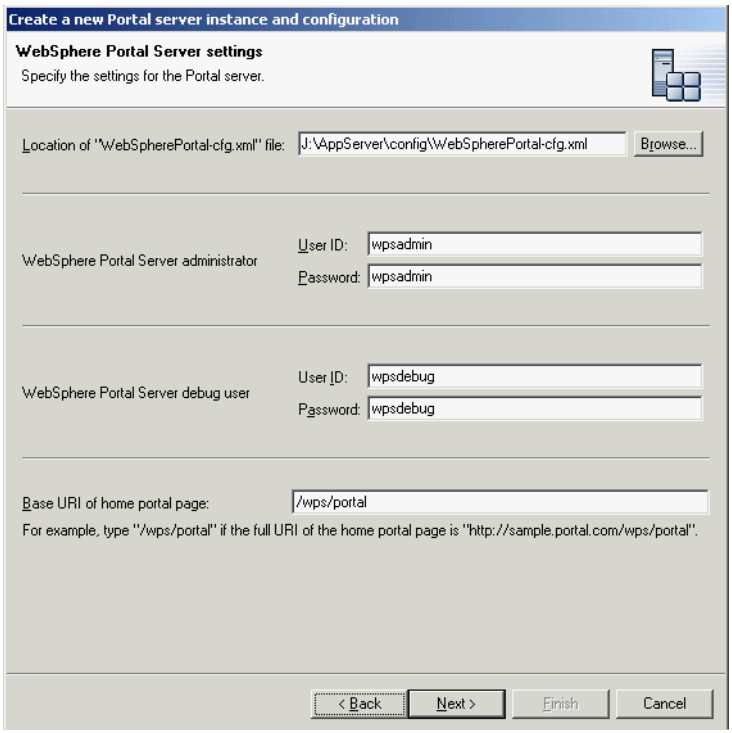


Figure 3-16 Portal Server instance settings

- For Location of “WebSpherePortal-cfg.xml” file field, browse the mapped drive. In a single node environment enter the local path for this file.

  This is the file created by the WebSphere Portal installation and is located in the <WAS_HOME>\config directory. For remote configuration, you will need to map a drive letter to a drive or directory that is shared on the remote machine that has access to the WebSphere Application Server directory. The Portal Toolkit will not recognize the file if there is not a drive letter associated with the location, so you cannot use the UNC (Universal Naming Convention) of the server share (for example, \\server\share is not acceptable). If this file is invalid or does not contain the Portal configurations, then you will not be able to proceed.
- WebSphere Portal Server administrator User ID: `wpsadmin`
- WebSphere Portal Server administrator Password: `wpsadmin`

  This is the wpsadmin account that was automatically created during the WebSphere Portal installation. Unless you have changed this user

id and/or password. It is required for the Portal Toolkit to update the page group and page information for displaying your portlet.

- WebSphere Portal Server debug user User ID: `wpsdebug`
- WebSphere Portal Server debug user Password: `wpsdebug`

  This account is the user ID that you created in 3.2.6, “WebSphere Portal debug ID configuration” on page 119. It is used to log into WebSphere Portal and will contain your portlets on a page called Debug configured by the Portal Toolkit during the server startup.

- Base URI of home portal page: `/wps/portal`

  This is the URI for the initial portal home page. It was configured during the WebSphere Portal installation. If you made changes to this URI during the WebSphere Portal installation, you will need to modify it here.

9. On the Create a new server instance and configuration page, set as follows and then click **Next**.
   – Server name: `RemotePortalDev`

     In the Server name field, enter any name you wish to use and is for display purposes only. For example, we entered `RemotePortalDev`.

     This server name will be displayed inside the folder name specified below. You will be prompted to create a new server project with the name that you specify if it doesn't already exist.

   – Folder: `PortalServerWC`

     In the Folder field, enter any name you want or use an existing folder that the configuration will be stored in. For example, we enter `PortalServerWC`.

     This folder name will be displayed on the server panel in WebSphere Studio Application Developer. You can use Servers if you want to have your Portal instances in the same folder as any other server instance you have created.

   – Server instance: select and expand **WebSphere Servers -> WebSphere v4.0 Remote Server**.

     This is the only configuration type supported by the Portal Toolkit. This setting tells the Portal Toolkit that files will need to be copied to the remote location and deployed remotely, and is not running on the internal WebSphere v4.0 Test Environment located in the plug-ins directory of Application Developer.

   – Template: select **None** from the drop-down menu. As no templates are required.

d. In the Template field under Server configuration type, select **None** from the drop-down menu (see Figure 3-17). As no templates are required.

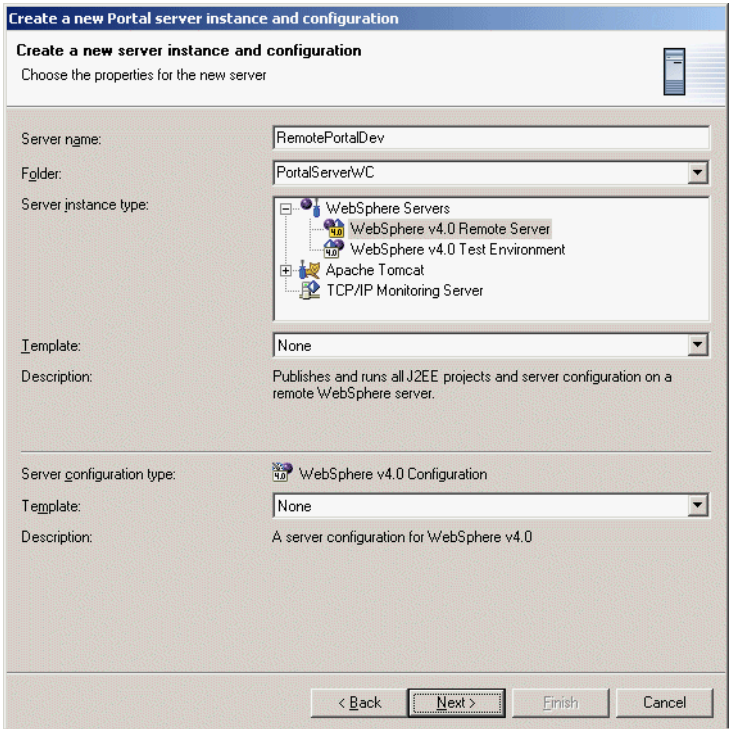


Figure 3-17 Properties for the new server

10. If you entered a new project name in the **folder** field in previous steps, then a confirmation window will appear for project creation. Click **Yes** to create a new server project with the name you have specified in above steps.
11. On the WebSphere Remote Server Instance page, set the following and click **Next**:

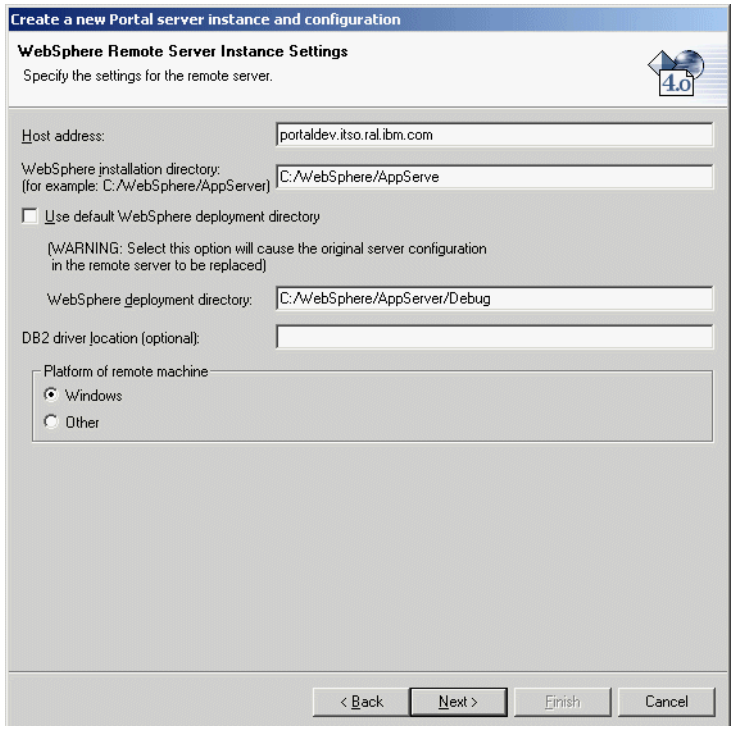


Figure 3-18 Remote server settings

– Host address: `<PortalTestNode_fullyqualified_hostname>`

  In the Host address field, enter your fully qualified server name (for example, we entered `portaldev.itso.ral.ibm.com`).

  This is the fully qualified domain name specified during the WebSphere Portal installation and the same name used in the browser for testing without the port appended. It is important that you use the exact same name specified when you installed WebSphere Portal. If this name is different, an error will occur when you use the automatic login feature of the Portal Toolkit, and you will be required to log in each time you debug your portlet. This name must be able to be resolved from the browser. If you are unsure, run the command ping <yourhostname> from the command line to see if an IP address is resolved. If you tested the Portal installation earlier, then this should not be a problem

– WebSphere installation directory: <WAS_HOME>

  In the WebSphere installation directory field, enter the installation path of the WebSphere Application Server Advanced Single Server Edition. For example, `C:/WebSphere/AppServer`.

If WebSphere Application Server is on a remote system and you have mapped a drive to access the remote system, then you will need to specify the actual local drive letter as you would specify on the remote machine; do not specify a mapped network drive.

– In our example, we deselected the checkbox for **Use default WebSphere deployment directory** so that we can use the debug folder for the configuration file.

  If this is not deselected then the configuration that is used will be in the <WAS_HOME>\config directory, and will overwrite the default server-cfg.xml used by WebSphere Application Server. It will also use the <WAS_HOME>\InstalledApps directory to deploy your portlet. This is perfectly acceptable. You can use this option if you are not going to be using the default server-cfg.xml file to run the default configuration for WebSphere Application Server itself. In the above scenario, the server-cfg.xml file and the deployed portlet application will be placed in a separate directory in order to keep the portal application and configuration separate from the original WebSphere Application Server and WebSphere Portal configuration. It is important that you understand which method you are going to use in order to configure the next configuration pane.

  If the WebSphere Application Server is on a remote system and you have mapped a drive to access the remote system, then you will need to specify the actual local drive letter as you would specify on the remote machine; do not specify a mapped network drive.

– WebSphere deployment directory: `<WAS_HOME>/debug`
– In the WebSphere deployment directory, enter the folder path you have created for the debug directory. For example, `C:/WebSphere/AppServer/Debug`.
– Under Platform of remote machine, select the **Windows** radio button. If you do not using Windows 2000, then specify **Other** as the platform.

12. Select **Create a new remote file transfer instance** options and **Copy file transfer mechanism**. Since you will not be using FTP as the transfer method, and click **Next**.

> **Note:** For single node development environment or you have installed both applications (WebSphere Application Server Advanced Single Server Edition and WebSphere Studio Application Developer) on same machine, you have to select **remote file transfer instance**.

13. Remote file transfer setting windows will be different based on the previous selection. In our example, we selected **Copy file transfer mechanism**. On the Remote File Transfer Settings page, set as following and click **Next** when done (see Figure 3-19 on page 134).

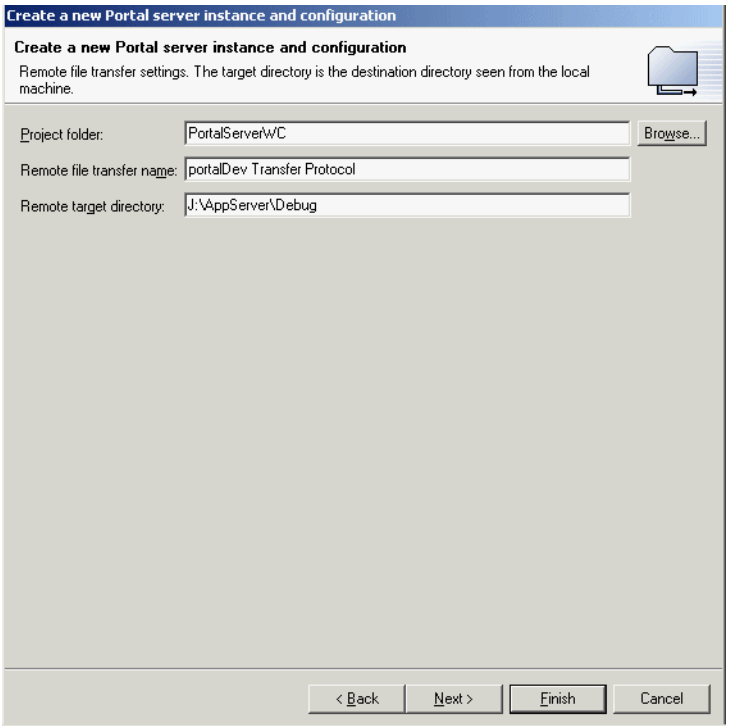


Figure 3-19 Remote file transfer settings

- In the Project folder, enter folder name for example, in ITSO we used `PortalServerWC`. This is the folder inside WebSphere Studio Application Developer where the configuration will be stored. You can use the same folder name specified throughout the configuration to keep everything in the same place.
- In the Remote file transfer name field, enter any name you want as long as it does not already exist. For example, we set `portalDev Transfer Protocol`. This name will be the file name displayed under the folder you specify.
- In the Remote target directory field, enter x:\<WAS_HOME>\Debug.

  This parameter is used to copy the files to the remote machine (or local machine, depending on your configuration). Your portlet application files and server configuration will be copied to the location specified here. If this directory is not configured properly, then WebSphere Application Server will be started using the configuration specified in the previous Remote Server Instance, which will cause WebSphere Application Server to fail since it cannot locate the proper files to debug. Before you click Next, you must create the directory specified here if it does not exist. If you do not do

this, then you will receive a warning when you attempt to publish your project, stating that the directory does not exist and that the publish will fail.

This must be the same location that was specified previously in the WebSphere Remote Server Instance screen for the WebSphere Deployment Directory. If WebSphere Application Server is on the same machine, then the fields will be identical. If you are using a remote configuration with a mapped drive, then the path may not be identical, but the paths must ultimately point to the same location.

For example, we mapped a J:\ drive to C:\WebSphere on the remote machine and the directory specified for the WebSphere Deployment Directory is C:\WebSphere\AppServer\Debug. Now we can specify as J:\AppServer\Debug for this parameter.

Note: This path must be accessible from the machine where the Portal Toolkit is installed in order for the files to be copied.

14. On the WebSphere Server Configuration Settings page, enter 9080 as the HTTP port number. Note that this was specified during the WebSphere Portal installation. By default, this should be 9080 because you are connecting directly to the HTTP transport of WebSphere Application Server. Click **Finish** and wait for the wizard to generate all necessary configurations.

3.3.6 Testing the WebSphere Portal Server instance

Now, it is time to verify the configuration from the previous section. To start the newly created WebSphere Portal instance do the following:

1. Make sure WebSphere Application Server is stopped on the WebSphere Portal development test node.
2. Open the WebSphere Studio Application Developer if it is not opened.
3. In WebSphere Studio Application Developer, select **Perspective -> Open -> Server** as seen in Figure 3-20 on page 136.

   Alternatively, **Perspective -> Open -> Server** depending on your WebSphere Studio Application Developer settings. If you have J2EE or Portlet perspective setting then you can use first option. If you have Server perspective then you second option from the menu.

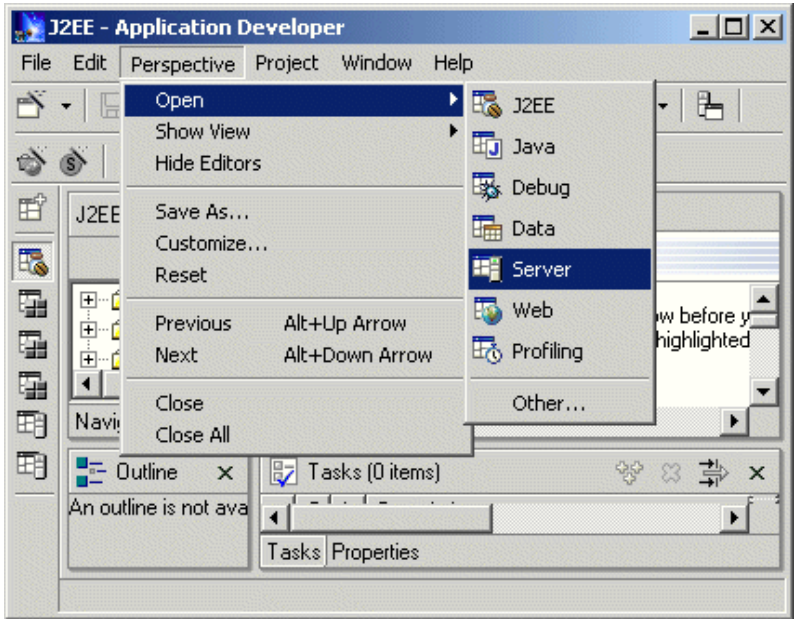


Figure 3-20 Open server perspective

4. In the result of the previous action you can see the newly created instance under Navigation pane, and server pane as illustrated in Figure 3-21 on page 137. If the Server pane is not visible then on the bottom right side of the screen, you should see a pane with several tabs. Click the **Servers** tab to view a list of servers. If you cannot find the tab, select **Perspective -> Show View -> Servers** from the main menu.

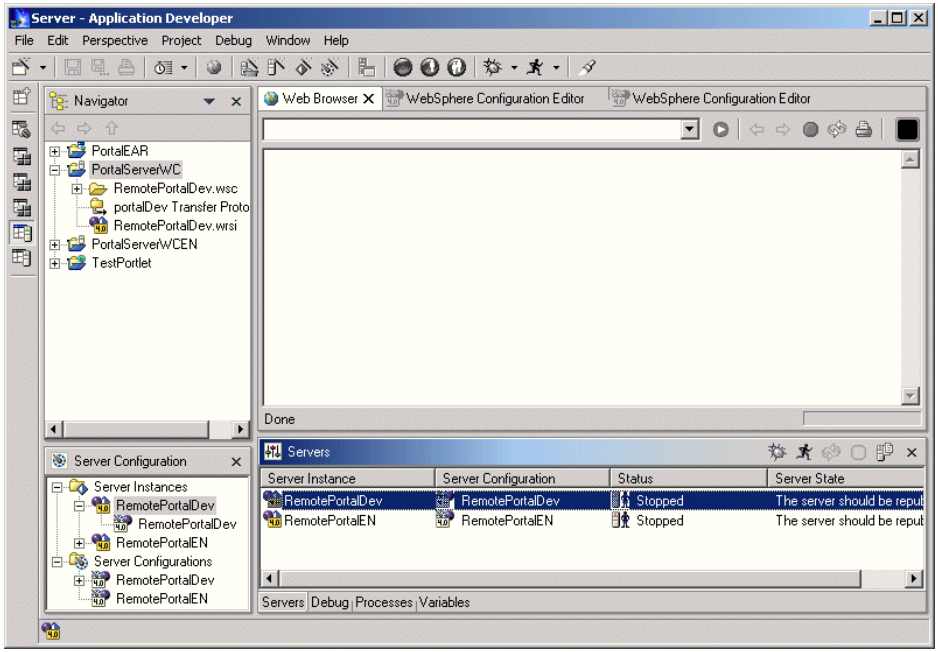


Figure 3-21 Server perspective

5. In the Servers pane select the newly created instance (RemotePortalDev). With the server selected, right-click and you can now see there are several options (Debug, Start, Restart, Stop, and Publish). You can activate these options by clicking the icon on top right corner of Server pane or by right-clicking and selecting the options from the list as show in Figure 3-22 on page 138.

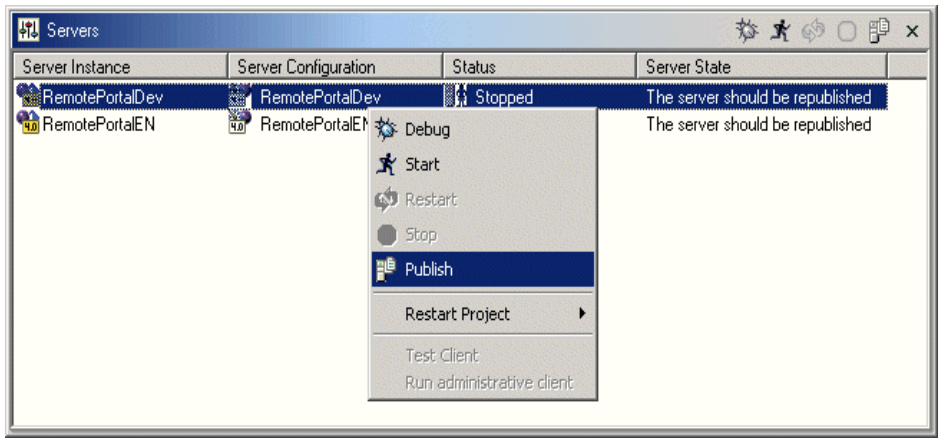


Figure 3-22 Servers options

Publish option copies the server configuration and all necessary assets to the target directory. For example, `c:\WebSphere\AppServer\Debug`.

Start not only starts the server but also publishes if WebSphere Application Server has detected any changes that affect the server.

Debug also publishes if necessary and the starts the server in debug mode. This means processing will be halted at defined and active debug points. Running in debug mode is a quite resource intensive process and unless your system is not virtually unlimited in memory and processing power. We recommend that you have set breakpoints before you start debug option.

6. We should now publish to test the remote file transfer and instance creation. To do this click **Publish** while newly instance is selected, which will open Publishing status in a separate window (see in Figure 3-23 on page 139).

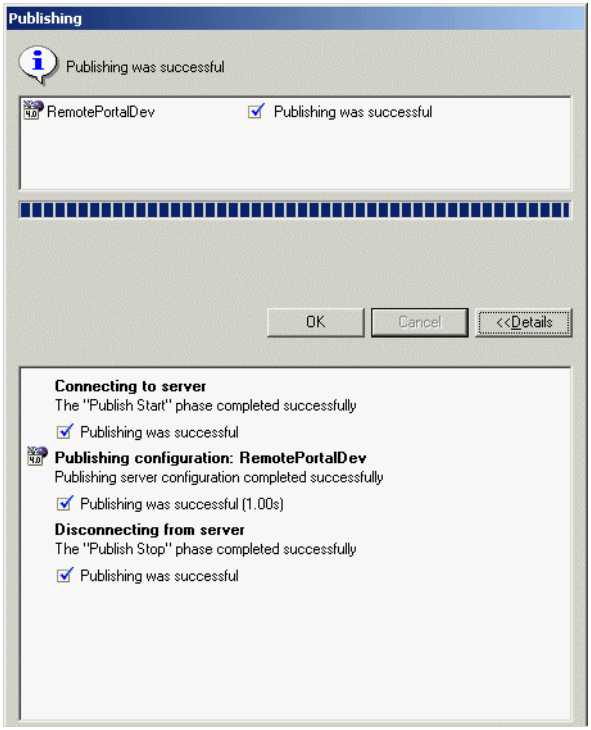


Figure 3-23 Publishing status

7. After **Publishing was successful** appears in the Publishing window, click **Details**. When you have done click **OK** to close the Publishing window. You will see in the Server pane the Server State has changed to **Server is synchronized** from **The server should be publish** (see Figure 3-24).

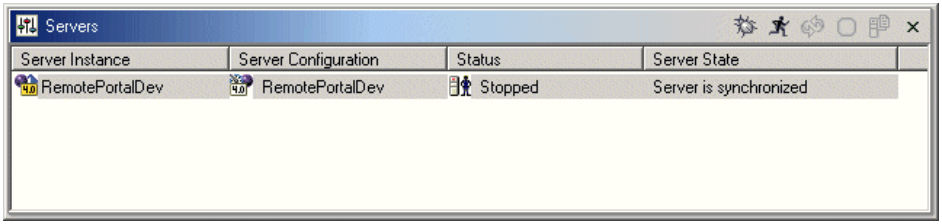


Figure 3-24 Server Pane after successful published

8. After you successfully published, right-click and select **Start** from your servers pane (make sure your server instance is selected). This will take some time, eventually the Console tab will appear in the server pane of the WebSphere Studio Application Developer workbench. You can maximize it by

double-clicking on the title bar if you want. Otherwise just wait until the message **Server WebSphere Portal open for e-business** appears in the console window as shown in Figure 3-25.

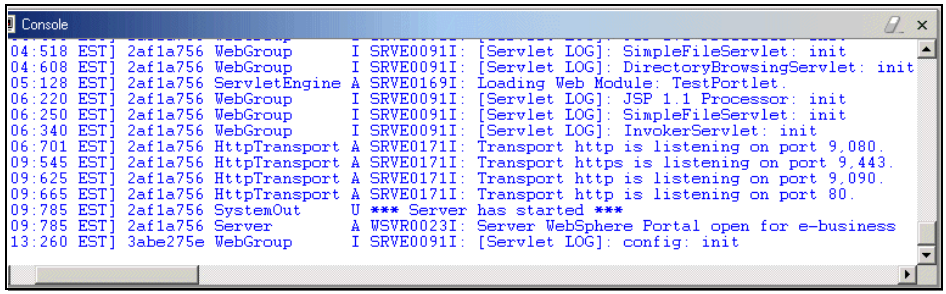


Figure 3-25 Console of started server instance

9. After successful server start, browse the Portal Server by entering the following URL:

   `http://<hostname>:9080/wps/portal`

   Replace the <hostname> with your Portal Server machine host name. For example, `http://portaldev.itso.ral.ibm.com:9080/wps/portal`. You should see something like Figure 3-26 on page 141.

   Note: If your server does not start successfully search the console for error message.

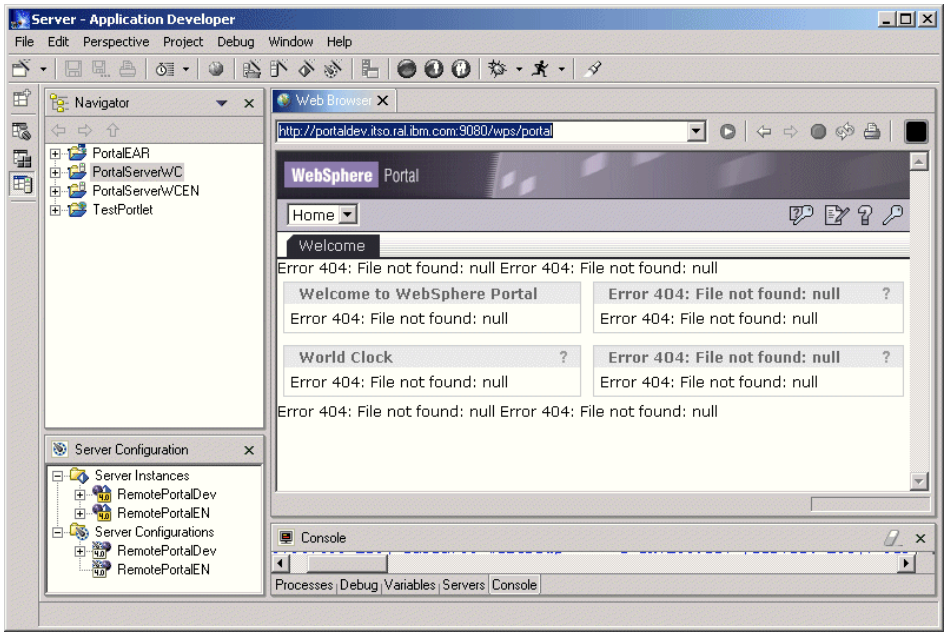


Figure 3-26 Portal test via browser

Only the portal skeleton is displayed. All the portlet will through 404 error message. At this point WebSphere Portal Toolkit did not deploy and portlet and that is reasons you getting all 404 Error messages. To see this portlet contents you have deploy manually.

Nothing to worry about these 404 error messages. As test is completed, **Stop** the server from your servers pane and continue to create a portlet application see details in, “Create a portlet application for testing and debugging” on page 141.

Create a portlet application for testing and debugging

This section is not mandatory. As the portal server is up and running, it is a good idea to create a sample project and a test portlet to deploy and test using WebSphere Portal Toolkit with WebSphere Studio Application Developer.

To create a portlet application do the following:

1. In WebSphere Studio Application Developer, click **File -> New-> Other** from the menu. Then select **Portlet development** on the left panel and **Portlet application project** on the right panel (see Figure 3-25 on page 140). When you done click **Next**.

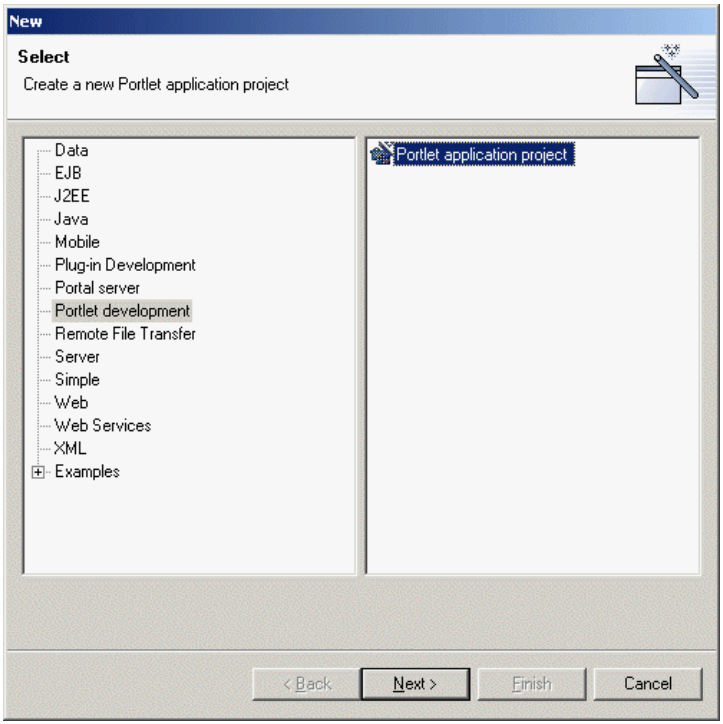


Figure 3-27 Create portlet application

2. On the Define the Portlet project screen, enter the following and then click **Next**:
   - Project name: `ITSOPortlet`

     You can choose any name for your Studio project; the name must be unique.
   - Select the **Use default location** check box.

     If you want to place the files in a separate location for management purposes, then unchecked this box and specify a new location into Location field.
   - Enterprise Application project name: `ITSOPortalEAR` or select from the existing list.

     Enter the name of the enterprise application (EAR) that you want to use. You can place multiple portlet applications (WARs) in a single enterprise application. The enterprise application is the file deployed by the Portal Toolkit in the test environment. In a full installation, the WAR file would be deployed.

– Context root: `ITSOPortlet`

  This will be the context root of the portlet application. This must be unique for all of the portlet applications that you want to deploy on this server. On a full installation, the context root is assigned a value during installation. This field is only used for the Portal Toolkit and will not be used if you deploy your application on a full Portal installation.

– Do not select **Create CSS file** checkbox.

  Only select this box when you have CSS file.

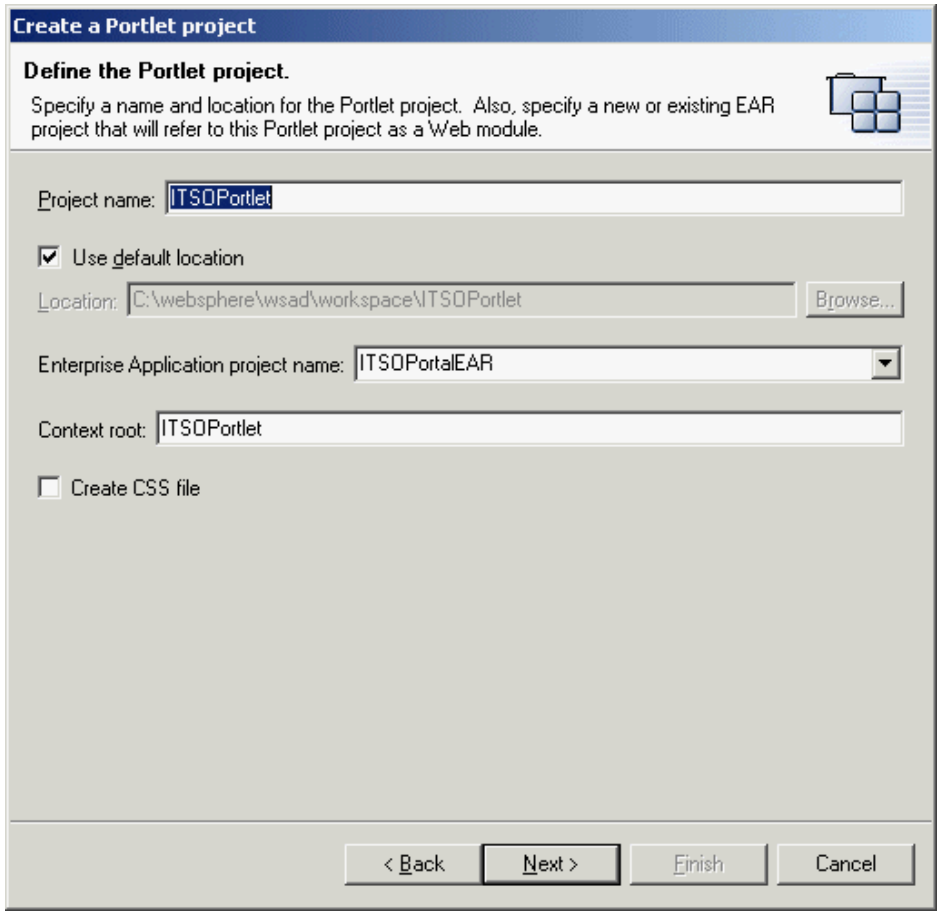


Figure 3-28 Define the Portlet project

3. In the portlet selection screen select **MVC portlet** options and click **Next**.

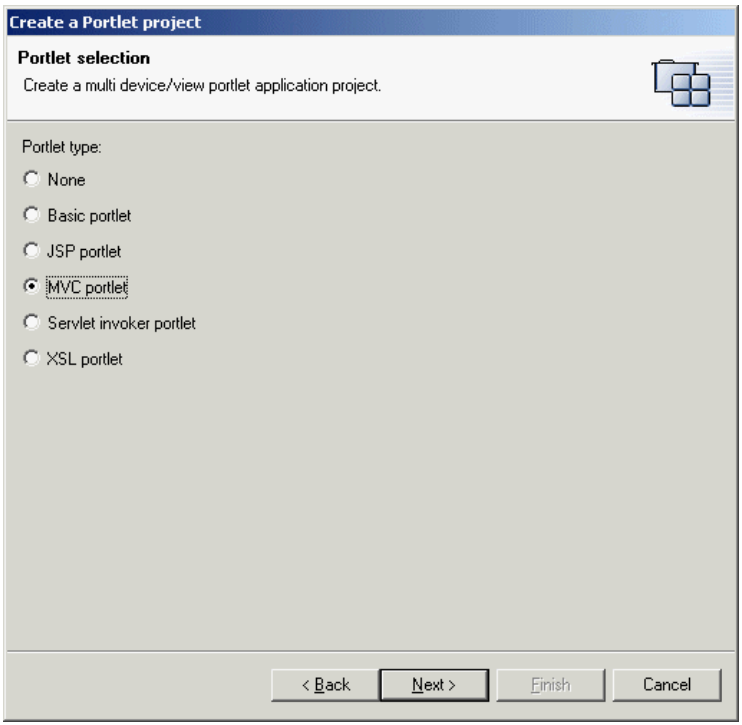


Figure 3-29 Portlet type selection

The MVC portlet is a Model-View-Controller portlet that lets WebSphere Portal control the type of response to provide, which is based on the client. For example, you could write a WML controller that Portal would use if it detected a WML browser. With the MVC portlet, you do not need to worry about the client because Portal takes care of this type of functionality.

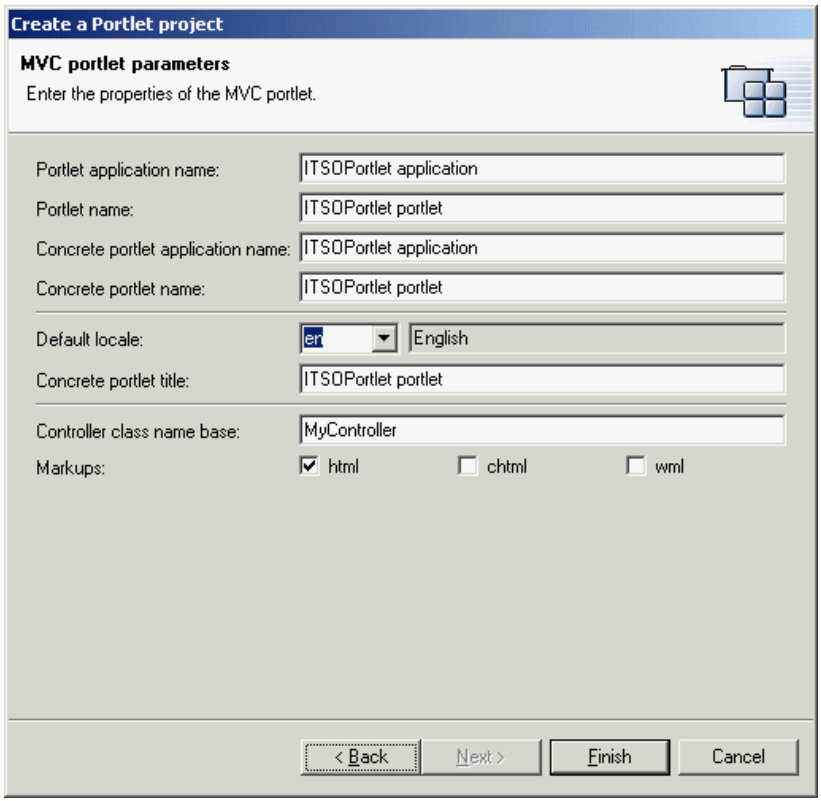


Figure 3-30 MVC Portlet parameters

4. Accept the values entered automatically in MVC Portlet parameters screen, and click **Finish** to create the portal application (Figure 3-30).

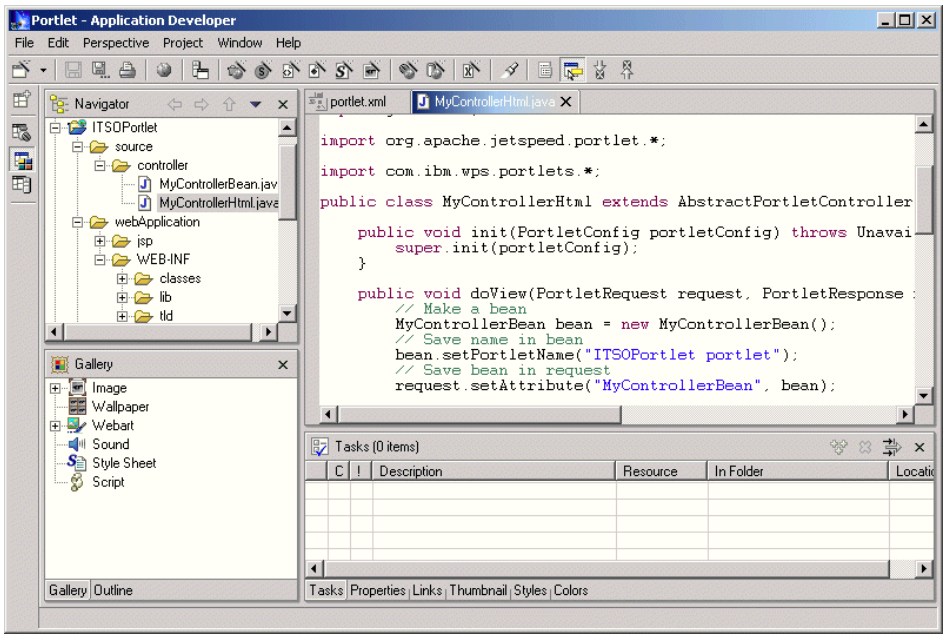


Figure 3-31 Portlet perspective

5. Portlet Toolkit will generate project and all assets folders, java classes, configuration files, HTML files, and JSPs that are required. It will also open the Portlet Perspective. Browse through all the folders and files in the Navigation panel to get and idea of what is created (see Figure 3-31).

Deploy and test the portlet application

Now that the portlet application has been created, deploy your portlet in debug mode and see what it looks like running in WebSphere Portal.

To deploy and debug do the folPowing:

1. Make sure WebSphere Application Server is not running. If still running stop it from the Server panel on Server perspective.
2. Open the project you have created in previous section. For example, `ITSOPortlet` project, which will open the Portlet perspective.
3. Open the MyControllerHtml.java under **ITSOPortlet -> source -> controller**. When you opened it, your WebSphere Studio Application Developer workbench should look similar to Figure 3-31, “Portlet perspective” on page 146. If you like to modify some of the code, then do so. Make sure the you know what you are doing.

4. Set a breakpoint right at the beginning of the **doView()** method. Right click the grey border close to the statement you want processing to stop to display the context menu. Then click on **Add Breakpoint**. For example, we set it on the first line of code in doView() method see in Figure 3-32, “Add breakpoint” on page 147. A blue bullet point will appear on the grey border across the code line. You also can do this by double click on the grey border across the code line you would like to set the break point. By double clicking on the blue bullet point it will remove the break point. Also can be removed by selecting **Remove Breakpoint** option by right-clicking.

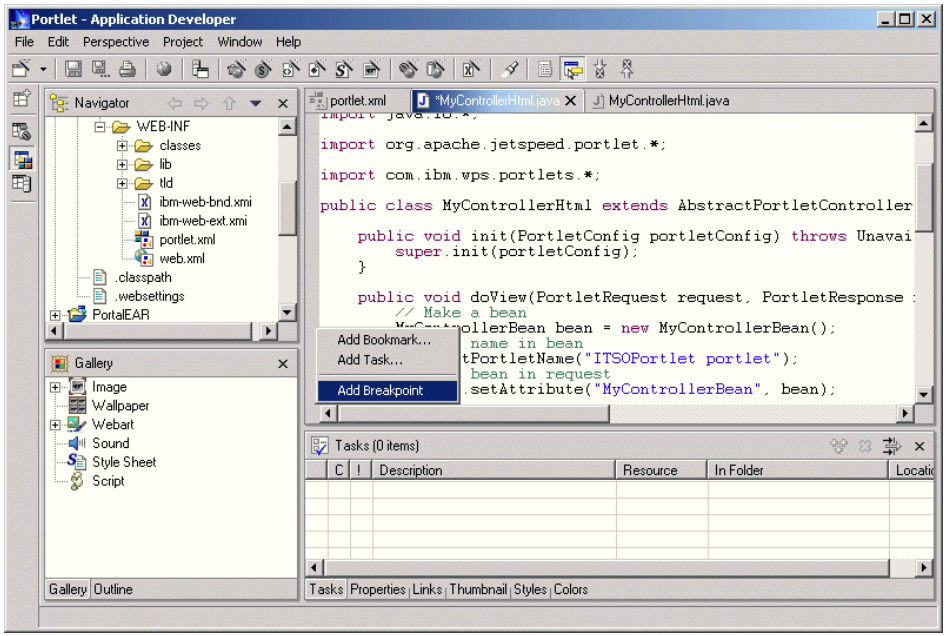


Figure 3-32 Add breakpoint

5. When you have completed adding the breakpoint, you need to go back to the Server perspective to select the correct server for deployment by clicking on the Server perspective icon or select **Perspective -> Open -> Server** as you did previously when verifying the server configuration.
6. Stop the server (RemotePortalDev) if it is running.
7. Right-click on the configuration that you created earlier (RemotePortalDev), and select **Add Project -> ITSOPortletEAR** (or the enterprise application project your portlets reside). This tells Application Developer to use this particular server configuration to run your portlet application (see Figure 3-33 on page 148).

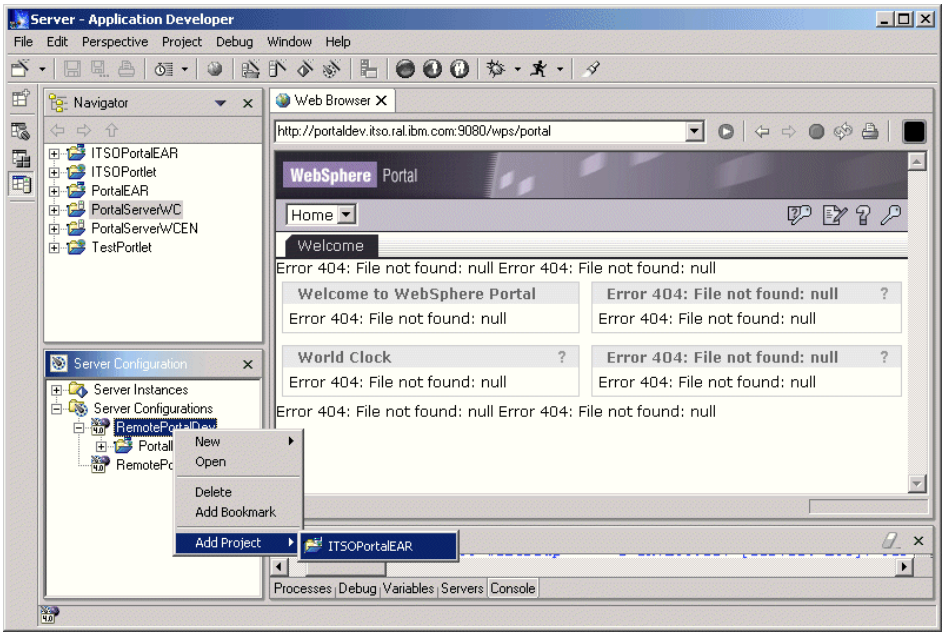


Figure 3-33 Add portlet application to server

8. After you add the breakpoint into the server configuration, the Server pane in Server perspective indicates that `The server should be republished`.

   Now start the server in debugging mode. Select the server instance in the Servers pane, right-click and select **Debug** as you can see in Figure 3-34.

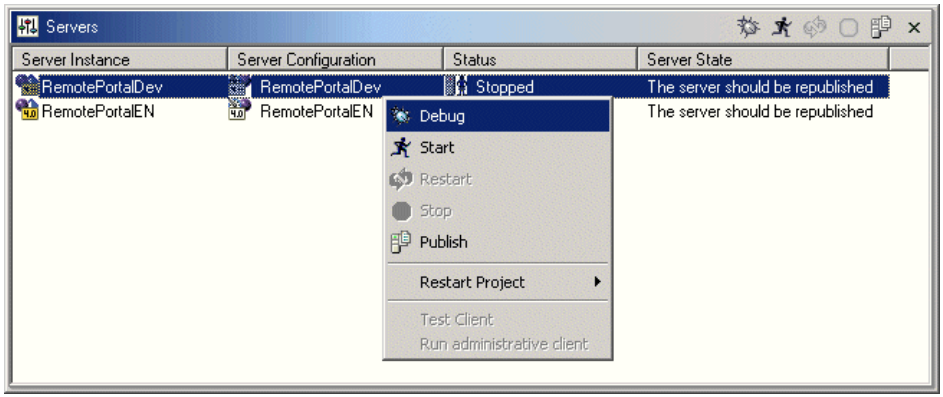


Figure 3-34 Start server in Debug mode

9. This action will also publish the portlet application to the server and the server will be synchronized again. Wait until the server has completely started and the message Server WebSphere Portal `open for e-business` is displayed in Console pane.

> **Note:** This process might take several minutes depending on your hardware.

10. When the server is started, switch to the Portlet perspective from the Server perspective. Right click on your portlet project in the Navigation pane, (for example, `ITSoPortlet`), and select **Run on Server**.

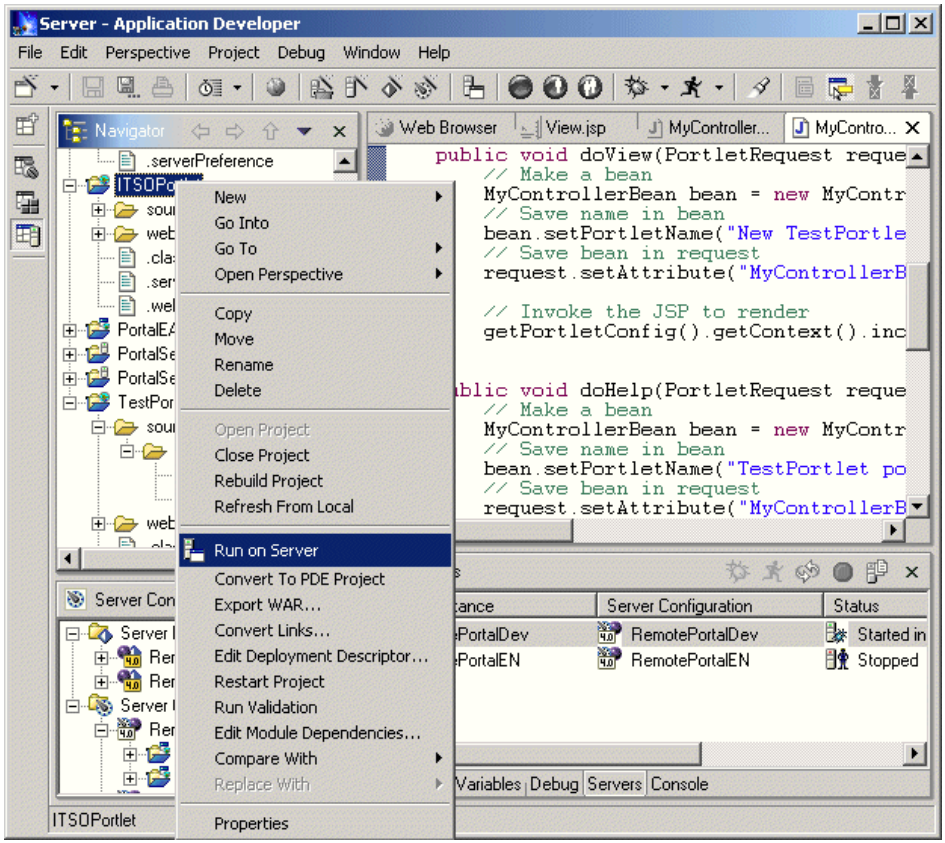


Figure 3-35 Run portlet through Run on Server option

11. You will notice that Debugger window has opened. After a while you will see MYControlHTML.java source opens in the context pane and highlighted the line we set the break point.

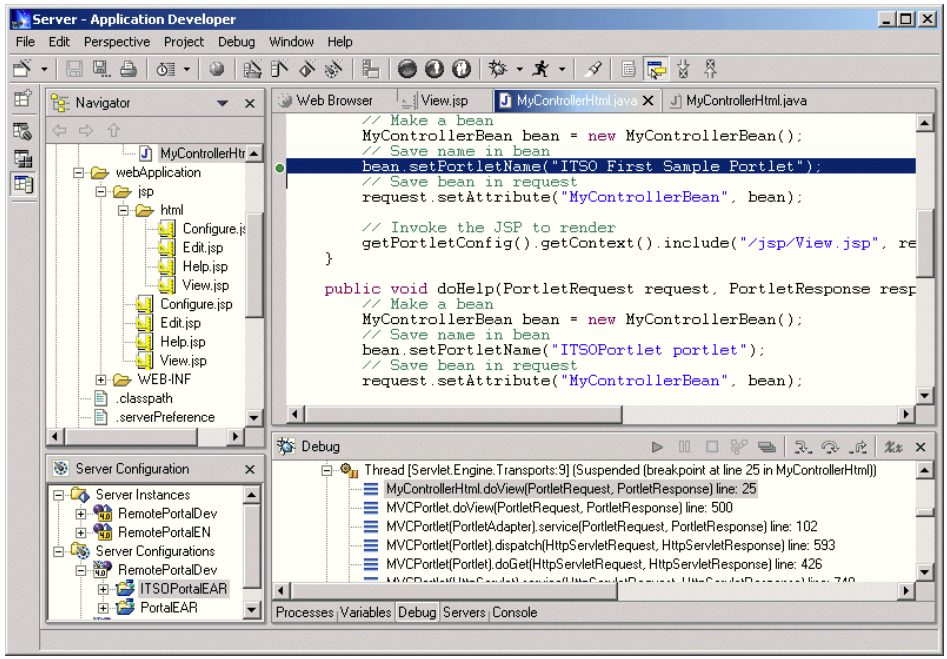


Figure 3-36 Debug window

12. Now can choose from several options to debug example Step in or Step Over. When debug is completed, the portlet will display in the browser. Note the user ID and the password in the URL. This lets the Portal Toolkit log in automatically without having to request the user information. If you get a login prompt or another page, there may be an issue with a previous connection. Click the **Refresh** or the **Logoff** button, and try again. Also, verify that the hostname and port are correct.

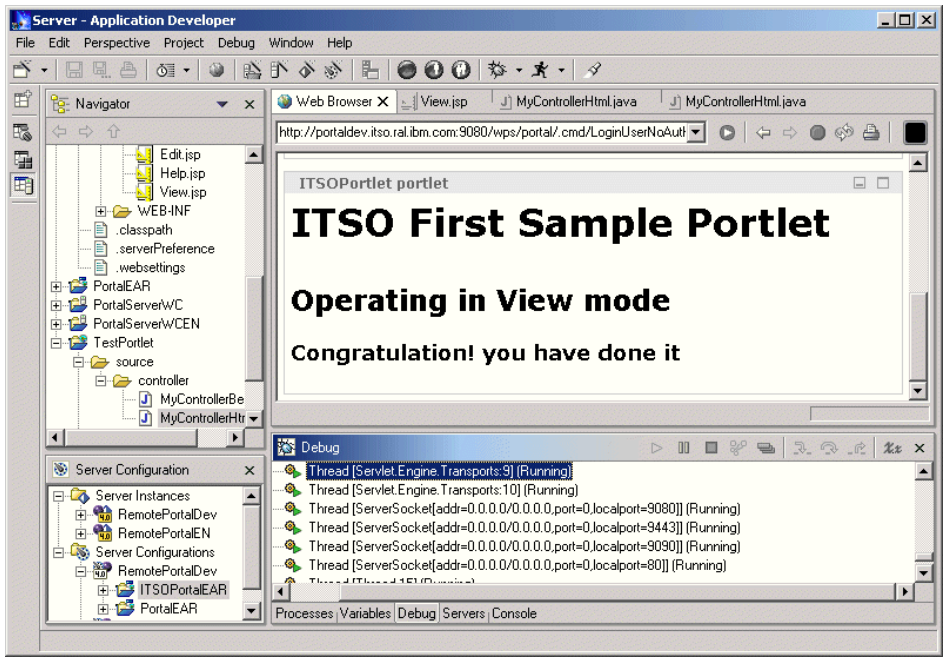


Figure 3-37 ITSO sample portlet

3.3.7 DB2 installation

In our example we used DB2 Universal Database™ as your database for WebSphere Commerce Studio. If you install your WebSphere Commerce development and test tools in single node configuration, then you do not need to install client separately. But in multi-node or remote database development environment you do need to install DB2 Client on your node where VisualAge for Java has been installed.

To setup a proper commerce development environment, you need to create WebSphere Commerce instance and WebSphere Commerce instance database (Mall). During the WebSphere Commerce Studio installation you can choose these options to create WebSphere Commerce instance and Mall database, but if you select remote database option in that case, those options will be grayed out.

To solve this problem, you can choose any options mentioned below:

- Install WebSphere Commerce Studio on DB2 server machine with local database and create WebSphere Commerce instance and Mall database options selected. On the DB2 Client machine also install WebSphere Commerce Studio on DB2 server machine with remote database options.

- Install DB2 Universal Database Server on both machines. Ultimately one will act as Server and other one as Client. For the client machine also select DB2 Application Development Client options during DB2 Server installation process. Now install WebSphere Commerce Studio on DB2 client machine with local database and create WebSphere Commerce instance and Mall database options selected. Note that at this point you have installed DB2 Server on both machine so you can select WebSphere Commerce instance and Mall database options. Once WebSphere Commerce Studio installation completed. Create backup the Mall database and restore in other machine where you also installed DB2 server. Now you can remove the mall data base from the machine where you have installed WebSphere Commerce Studio as from now on this machine will act as a DB2 Client.

  Note: Our example documents this solution.

In both cases you do have to configure the DB2 Client machine to get access to the DB2 server and databases. The second option is much cleaner but you may confuse why do you have to install DB2 Server on both machine. Eventually one will act as a Server and other will act as a Client. For the first option you have to install WebSphere Commerce Studio on both machines. Even though you will not use any of the applications installed by WebSphere Commerce Studio installation process except WebSphere Commerce instance database (Mall database). This book describes the second option in detail. If you like to choose the first option, you can get the fair amount of idea how to install and configure from this sections.

Note: For more information on installing WebSphere Commerce Studio, refer to *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*.

For normal DB2 Client installation refer to the *WebSphere Commerce V5.4 Handbook*, SG24-6567 or the *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*.

We need to install DB2 Server on client machine to create WebSphere Commerce instance and Mall database. For details refer to 2.2.2, “DB2 Server installation” on page 15.

3.3.8 WebSphere Commerce Studio installation

The standard development tools set for WebSphere Commerce V5.4, Business Edition, is the IBM WebSphere Commerce Studio V5.4, Business Developer

Edition for Windows NT and Windows 2000. This section highlights the WebSphere Commerce Studio, installation and configuration.

In this section first we will describe the steps to install the WebSphere Commerce Studio in a stand-alone development environment, which uses a local database. After installation, we will move the WebSphere Commerce instance database to a remote database machine. Then we will configure WebSphere Commerce Studio remote database. At the end of this section, our environment will match the, “Multi-node development environment separate portal runtime” on page 99.

The section is organized as follows:

- WebSphere Commerce Studio installation
- Move Mall database to remote DB2 server
- Configure WebSphere Commerce Studio remote database
- Configure VisualAge for Java
- Configure and start the EJB, PNS and Servlet Engine
- WebSphere Commerce Studio verification

Note: More detailed information can be found in the *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000* and in the *WebSphere Commerce V5.4 Developers Handbook*, SG24-6190

For a team environment we recommend the following high level procedure:

- Install and configure a local development environment
- Configure a repository server. Copy your repository to the repository server.
- Connect from other VisualAge for Java development workstation to the repository server.
- Configure and share databases for the persistent name service and your WebSphere Commerce instance database

WebSphere Commerce Studio installation

The primary components of WebSphere Commerce Studio V5.4 are as follows:

- IBM WebSphere Studio
- WebSphere Commerce Studio extensions
- IBM VisualAge for Java
- WebSphere Commerce instance database

The high level installation and configuration procedure is as follows:

1. Windows 2000 Server installation
   - Ensure Windows 2000 Server and service pack 3 has been installed

- Ensure an administrator user is logged in for installation the WebSphere Commerce Studio and supporting software with the following user rights: act as part of the operating, create a token object, increase quotas, replace a process level token
- Ensure Internet Explorer 5.5 and service pack or higher has been installed

2. Make sure that you have installed DB2 UDB V7.2, Enterprise Edition, DB2 V7 FixPak 7 (7.1.0.68) and updated the JDBC level to JDBC2.

> **Note:** We choose the installation procedure to develop store-front assets, and back-office business logic. For other installation procedures we refer to the *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*.

3. Log on to a Windows user ID that has Administrator authority.
4. Insert the WebSphere Commerce Studio, Version 5.4 CD into your CD-ROM drive.
5. From the root directory of the WebSphere Commerce Studio CD, run `setup.exe`.
6. The Choose Language window opens. Select your language and click **OK** to continue.
7. The Welcome window opens. Click **Next** to continue.
8. The License Agreement window opens. Review the terms of the license agreement and click **Accept**.
9. The Select Components window opens. Select the following:
   - **Develop store-front assets using WebSphere Studio** check box
   - **Develop Store Backend logic using Visual Age for Java** check box
   - The **Select Database** drop-down list allows you to select which database will be used for development.

   When you have made your selections, click **Next** to continue.
10. The WebSphere Commerce Instance Information window allows you to enter the following information about your WebSphere Commerce instance:
    - Instance name: `demo`
    - Mall database: `mall`

> **Note:** Use the same settings as in the runtime environment. We use `demo` for the instance name and `MALL` for the database.

- Select the **Create Database** check box to create your WebSphere Commerce database.
- Select the **Include Sample Store** check box to create a sample store in the database automatically to verify your installation.

When you have done click **Next** to continue

11. Enter the database information in the Database Information window. Click **Next** to continue.
12. Customize your installation path in the Choose Destination window. Click **Next** to continue. For example, we entered the following:
    - Store Archive Tools + Blaze: `c:\ibm\wcstudio`
    - WebSphere Studio: `c:\ibm\wstudio`
    - VisualAge for Java: `c:\ibm\vaj`
    - WebSphere Commerce: `c:\ibm\wc`
13. The Summary window displays a summary of the selections you have made. Click **Next** to continue.
14. Depending on your software packages already installed on your system, you are prompted to insert the appropriate CDs.
15. After WebSphere Commerce Studio has been installed, you are prompted to restart your machine. Click **Finish** to restart.

Move Mall database to remote DB2 server

In our example, our WebSphere Commerce instance database is on a remote node (WebSphere Portal development test node). To avoid configuration problems within our environment, we created the instance database on the same node as the development tools. This is not a required step, but desirable when shared the instance database with other team members.

To move Mall database from development machine to DB2 Server machine do the following on the Development Tools node, do the following:

1. Get the Database territory and code set by typing the following command in DB2 command window:

   ```
   db2 connect to Mall
   db2 get db cfg for Mall
   ```

   Record the Database Territory and Code Set, which will be found at the beginning of the above command output. This will be used when you create Mall database in DB2 Server machine.

2. Create a backup of Mall database by using following command in a DB2 Command window:

   ```
   db2 backup db Mall to c:\temp
   ```

We assume this the first time you have created back of Mall database in c:\temp. If not then you have to remember the backup taken datetime. To find this go to the folder C:\temp\MALL.0\DB2\NODE0000\ and will see the folder with backup date in format yyyymmdd. Remember this when you restore the Mall database in DB2 Server machine.

3. On the remote DB2 Server machine, copy c:\temp\Mall.0 folder from the Development Tools node to the c:\temp of the WebSphere Portal development test node (remote DB2 Server).
4. Create a Mall database by typing the following command in a DB2 Command window on the WebSphere Portal development test node (remote DB2 server):

   ```
   db2 create db Mall using codeset <codeset> territory <territory>
   ```

   Where <codeset> and <territory> will be replace with the value from step on of this section. For example, in ITSO:

   ```
   db2 create db Mall using codeset UTF-8 territory US
   ```

5. Restore Mall database from backup (c:\temp) by typing the following command in a DB2 command window:

   ```
   db2 restore db Mall from c:\temp into Mall
   ```

   If you have more than one back of Mall database in c:\temp folder then need to use taken attribute with the previous command.

   ```
   db2 restore db Mall from c:\temp taken at <date> into Mall
   ```

   The <date> can be obtained from the record values listed above (`db2 get db cfg for Mall`).

6. In this example, we now need to drop the Mall database on the Development Tools node. In a later configuration step we will configure the Development Tools node to communicate to the remote DB2 server and catalog the database. To drop the local Mall database on the Development Tools node enter the following from a DB2 command window:

   ```
   db2 force applications all
   db2 terminate
   db2 drop db Mall
   ```

Configure WebSphere Commerce Studio remote database

This section describes how to configure the connectivity of DB2 on the Development Tools node with the WebSphere Portal development test node where we have move the instance database.

1. On the WebSphere Portal development test node (remote DB2 Server), find the DB2 instance TCP/IP connection port in DB2 server machine.

For details how to find the connection port, see Chapter 2.7.3, “Verify the DB2 instance TCP/IP connection port” on page 88.

2. Catalog the TCP/IP node:

   On the Development Tools node, start a DB2 command window and enter the following to catalog the TCP/IP node:

   ```
   db2 catalog tcpip node <node_name> remote <database_server_hostname> server
   <port_number>
   ```

   Where the <node_name> is your WebSphere Commerce Studio node name. <data base_server_hostname> is the host name of your database server and <port_number> is the port being used by DB2 instance.

3. Catalog the Mall database.

   On the Development Tools node, catalog the Mall database run the following from a DB2 command window:

   ```
   db2 catalog db <remote_database_name> at <node_name>
   ```

   Where, <remore_database_name> is the name of your remote database and <node_name> is your WebSphere Commerce Studio node name.

4. Verify the database has been properly cataloged on the WebSphere Commerce Studio machine (client). To do this, run the following command in a DB2 command window:

   ```
   db2 list db directory
   ```

   Mall database entry should have a directory entry type of REMOTE and a catalog node number of -1.

5. Verify that you can connect to the mall database from WebSphere Commerce Studio machine. To check this, run the following command in a DB2 command window:

   ```
   db2 connect to <remote_database_name> user <db_user> using <db_password>
   ```

3.3.9 VisualAge for Java configuration

This section describes the high level steps to configure VisualAge for Java for WebSphere Commerce.

Note: For detailed installation and configuration instruction on WebSphere Commerce Studio V5.4, refer to the *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000* guide for details.

Configure VisualAge for Java

To configure VisualAge for Java, do the following:

1. Extract the WebSphere Commerce code repository (<wc_cd_2>\repository\WC_54.exe) to the node where WebSphere Commerce Studio is installaed (Development Tools node).
2. Apply the FixPak called `EJB-1.1-DeployedTool.zip`.

   This FixPak upgrades the version of the Export Tool for Enterprise Java Beans. This file is found on the WebSphere Commerce Studio CD in the `efixes/vaj` directory. Extract the zip to a temp directory and run Setup.
3. Open VisualAge for Java by clicking **Start > Programs > IBM VisualAge for Java for Windows > IBM VisualAge for Java**. You will be prompted to assign a the Windows user ID as the VisualAge for Java administrator.
4. Press F2 to launch the QuickStart panel.
5. Select **Features -> Add Feature** and select the following (use Ctrl key to select more than one feature); when done click **OK**.
   - Select **Export Tool for Enterprise Java Beans 1.1 4.1.5**
   - Select **IBM Common Connector Framework 3.5.3**
   - Select **IBM EJB Development Environment 3.5.3**
   - Select **IBM Java Record Library 3.5.3**
6. Install the e-Fix `readonly.zip`. This file is found in the efixes\VAJ directory of the WebSphere Commerce Studio CD.
   a. Unzip readonly.zip to the VisualAge for Java directory.
   b. Open VisualAge for Java.
   c. Select **Workspace -> Tools -> Fix Manager**.
   d. Select the following:

      `readonly - Prevent invalid deletion of EJB read-only attributes`
   e. Click **>>** to add to Fixes to load and then click **OK**.
   f. Confirm that an * (asterick) appears next to `readonly - Prevent invalid deletion of EJB read-only attributes`.
   g. Click **Close**.
   h. Exit VisualAge for Java.
   i. Restart VisualAge for Java for changes to take effect.
7. Prior to importing the WebSphere Commerce repository, import the PQ50159.jar and `ivjfix35.jar files`.

   To import PQ50159.jar, do the following:
   a. From the VisualAge for Java, select **File -> Import**.

b. Select **Jar file** and click **Next**.

c. Enter the path to the PQ50159.jar file in the filename field. The PQ50159.jar can be found in the WebSphere Commerce Studio CD \efixes\VAJ directory. Click **Browse** next to project, select **IBM WebSphere Test Environment**.

d. Click **Finish**.

e. When prompted to create edition for package, click **Yes to All**.

To import ivjfix35.jar, do the following:

a. From the VisualAge for Java, select **File -> Import**.

b. Select **Jar file** and click **Next**.

c. Enter the path to the ivjfix35.jar file in the filename field. The ivjfix35.jar can be found in the WebSphere Commerce Studio CD \efixes\VAJ directory. Click **Browse** next to project, select **IBM WebSphere Test Environment**.

d. Click **Finish**.

e. When prompted to create edition for package, click **Yes to All**.

8. Import the WebSphere Commerce repository as follows:

a. From the VisualAge for Java, select **File -> Import**.

b. Select **Repository** and click **Next**.

c. Select **Local repository**.

d. In the repository name field, enter the location of the unzipped repository. For example, c:\temp\wc_54.dat.

e. Select Projects and click Details. Select the following projects:

- IBM WC Commerce Server
- IBM WC Enterprise Beans

f. In the right-hand pane, select **WC 5.4** for version available and then click **OK.**

g. Ensure the **Add most recent project edition to workspace** is checked.

h. Click **Finish** to begin importing. This process takes several hours.

i. When prompted to create edition for package, click **Yes to All**.

9. After the import is complete, change the Workspace Owner to WCS Developer by selecting Workspace.

10. Save your workspace by selecting **File -> Save Workspace**.

Configure and start the EJB, PNS and Servlet Engine

To configure the EJB and PNS server for VisualAge for Java, do the following:

> **Note:** For details see, *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*.

1. Click **EJB** tab in VisualAge for Java workspace.
2. Add all WebSphere Commerce EJBs to Server configuration (start with WCS).
3. Set properties by selecting the server, right-click **Properties**.
4. We entered the following on the Properties page and then clicked **OK**:
   - Data Source: `WebSphere Commerce DB2 DataSource demo`

     Where `demo` is the name of the WebSphere Commerce instance.
   - Connection Type: select **<DataSource>**

     > **Note:** In this example, the < > brackets surrounding <DataSource> are part of the selection.

   - Transaction Timeout: `1200`
   - Transaction Inactivity Timeout: `600000`

   > **Note:** In the **Transaction timeout** field of the EJB server properties, enter `1200`. In the **Transaction inactivity timeout** field, enter `600000`. If the values are to small CORBA exception messages might occur.

5. Save the Workspace.
6. Create PNS database. From the Development Tools node, enter the following:

   ```
   db2 attach to <node> user <db2_user> using <password>
   db2 create db pns
   ```

7. Start the WebSphere Test Environment.
   a. From VisualAge for Java, select **Workspace -> Tools -> WebSphere Test Environment**.
   b. Select the **Persistent Name Server**.
   c. Enter the following for the Persistent Name Server and then click **Apply**:
      - Database URL: `jdbc:db2:pns`
      - Database driver, select **COM.ibm.db2.jdbc.DB2Driver** from the pull-down list.
      - Database ID: `<db2_user>`

- Database password: `<password>`

d. Click **Start Name Server**.

8. Set the properties for the Persistent Name Server.
   a. Select the **DataSource Configuration** and click **Add**.
   b. Enter the following on the Add DataSource window and then click **OK**:
      - DataSource Name: WebSphere Commerce DB2 DataSource demo

        Where `demo` is the WebSphere Commerce instance name.
      - Database Driver: select **COM.ibm.db2.jdbc.app.DB2Driver** (default)
      - Database URL: `jdbc:db2:mall`
9. Stop and start the Persistent Name Server.
10. Start the EJB server.
    a. From the EJB tab, in the Enterprise Beans pane, right-click **Open To -> Server Configuration**.
    b. Select **EJB Server**, right-click **Start**. The startup of the EJB Server takes several minutes.
11. Configure the Servlet Engine.
    a. From VisualAge for Java, select **Workspace -> Tools -> WebSphere Test Environment**.
    b. Select **Servlet Engine**.
    c. Click **Edit Classpath**.
    d. Click **Select All** and click **OK**.

    Note: If you want to display trace messages, check the Display trace messages checkbox. We checked this option.

    e. Click **Apply**.
12. Click **Start Server Engine**.
13. Verify the Servlet Engine has started by checking the console for “Servlet Engine is started”.
14. Save your workspace.

WebSphere Commerce Studio verification

To verify the configuration of WebSphere Commerce Studio, do the following:

1. Make sure PNS server, EJB server and Servlet engine are running in VisualAge for Java.

Note: In case of CORBA exceptions you should increase the EJB server properties for the **Transaction timeout** field and **Transaction inactivity timeout** field. Restart your EJB server and the servlet engine.

2. Open a Web browser and enter the following URL:

   ```
   http://localhost:8080/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=adminconsole.AdminConsoleLogon
   ```

3. Log in as user wcsadmin with password wcsadmin. You will be request to change this password.
4. Select **Site** from the Administration Console Site/Store Selection Page. Click **OK** to continue.
5. Publish contract data by running the following URL command in the original browser window:

   ```
   http://localhost:8080/wcs/contractPublish.html?storeId=10001
   ```

   Note: You must enter the URL in the original browser window opened when launching the AdminConsole (passed credentials - wscadmin).

6. To access your sample store enter the following URL:

   ```
   http://localhost:8080/webapp/wcs/stores/servlet/StoreCatalogDisplay?storeId=10001&catalogId=10001&langId=-1
   ```

7. Refer to the *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000* for WebSphere Payment Manager (WebSphere Commerce Payments) configuration.

3.3.10 WebSphere Commerce FixPak V5.4.0.3 installation

This section describes the high level steps to install the WebSphere Commerce FixPak V5.4.0.3 within the WebSphere Commerce Studio environment.

Note: More detail information can be found in the *Code Transition Guide, IBM WebSphere Commerce FixPak for Windows NT and Windows 2000 Version 5.4.0.3.*

1. We strongly recommend that you backup the following assets:
   - WebSphere Commerce directory tree
   - WebSphere Commerce instance database
   - VisualAge for Java assets
   - Studio assets

2. Download the WebSphere Commerce FixPak V5.4.0.3 from the IBM Web site at:

   http://www-1.ibm.com/support/docview.wss?rs=497&uid=swg24001839

3. Extract the files into a temporary directory on your WebSphere Commerce Studio machine. This directory must be writable.
4. Ensure that your are logged in with a Windows user ID that is a member of the Administrator group.
5. Shutdown the VisualAge for Java application.

   Important: We strongly recommend that you save your Workspace prior to stopping all the servers.

   At the time of writing this Redpaper, VisualAge for Java crashes when stopping the Persistent Name Server.

6. To start the FixPak install, type install_wc.bat.
7. You will be prompted with the following questions:
   - IBM WebSphere Commerce Version 5.4.0.3 FixPack install

     Are your applying this Fixpack to IBM WebSphere Commerce Studio? (y)es (n)o. We entered `y`.
   - Enter the directory where IBM WebSphere Commerce Studio is installed. We entered `c:\ibm\wc`

     Note: The install directory is really the WebSphere Commerce Server directory, not WebSphere Commerce Studio.

   - Enter the directory where java.exe is located. We entered `c:\ibm\wstudio\bin` (WebSphere Studio bin directory).
   - Enter the installed Edition of the IBM WebSphere Commerce Server. We entered `Business`.
8. Check the following log file of the WebSphere Commerce 5.4.0.3 FixPak installation was successful:

   `<WCSTUDIO_HOME>\services\fixpack\5403\WCfixpack<edition>5403_wc_WIN.log`

9. Add new bootstrap data by running the following scripts from the `<WC_HOME>/bin` directory:

   `updatedb.bat <db_name> <db2user> <db2_password> <schemaowner> <infile> <dbtype>`

   For example, for DB2:

```
updatedb.bat MALL admin <password> admin wcs.updateFP.bootstrap.xml DB2
```

10. Confirm that updatedb.bat command was successful by checking the file `<WC_HOME>/logs/updatedb.log`. The file should not contain error messages.
11. After the FixPak 5.4.0.3 install, perform the following copy:

    `<WC_HOME>/instances/default/xml/search.xml` to your `<WC_HOME>/instance/<instance_name>/xml` directory.

> **Note:** The FixPak 5.4.0.3 repository can not be imported into VisualAge for Java. We recommend that you install the Commerce Enhancement Pack instead.

3.3.11 Commerce Enhancement Pack installation

This section describes the high level steps to install the IBM Commerce Enhancement Pack - October 2002 Edition on the WebSphere Commerce Studio environment.

> **Note:** For detailed information on installing IBM Commerce Enhancement Pack - October 2002 Edition in a WebSphere Commerce Studio environment, refer to *Getting Started, IBM Commerce Enhancement Pack*.

The high level steps to install the Commerce Enhancement Pack for WebSphere Commerce Studio, are as following:

1. Check the following prerequisites:
   - Ensure WebSphere Commerce Version 5.4 FixPack 3 is installed
   - Ensure IBM DB2 Universal Database Version 7.2 FixPack 7 is installed
2. Install the Commerce Enhancement Pack.
3. Run the Commerce Enhancement Pack database update scripts.
4. Run the Commerce Enhancement Pack redeploy script against your instances.
5. Enable the appropriate services within your `instance_name.xml` file.
6. Disable SSL to run a store within the VisualAge for Java WebSphere Test Environment.

   From a DB2 command prompt, enter the following:

```
db2 connect to <db_name>
db2 update viewreg set https=0
db2 update urlreg set https=0
```

3.3.12 Import the IBM Commerce Enhancement Pack repository

To import the IBM Commerce Enhancement Pack - October 2002 Edition repository, do the following:

> **Note:** For details on importing the repository for the Commerce Enhancement Pack, refer to the *Code Transition Guide, IBM WebSphere Commerce FixPak for Windows NT and Windows 2000 Version 5.4.0.3.*

1. Start VisualAge for Java.
2. Ensure WC Developer is the Workspace owner.
3. Version all open edition projects.
4. Remove the **IBM WC Commerce Server** and **IBM WC Enterprise Beans** projects from the workspace. The delete operation takes several minutes.

   > **Note:** These are deleted from the workspace only. The import of the Commerce Enhancement Pack level of the repository will create these in the workspace.

5. Remove the original EJB server configuration.
6. Save your repository file (ivj.dat, unless you have altered) as well as the IDE workspace file ide.icx to a new location outside your VisualAge for Java directory tree. In the event something goes wrong, you can copy the files back to VisualAge for Java.
7. Download the IBM Commerce Enhancement Pack repository from:

   `http://www.ibm.com/software/commerce/epacks`

8. Unzip the downloaded file into temp direcotry. You will see a dat file and project resource folder for that dat file. For example:

   ```
   CommerceEnhancementPack_OCTOBER2002.dat file
   CommerceEnhancementPack_OCTOBER2002.dat.pr folder
   ```

9. Import the dat file to update the repository, which you have extracted in above steps. To import do the following in VisualAge for Java:
   a. From the File menu, select **Import**.
   b. On the Import SmartGuide window select the **Repository** radio button and click **Next**.
   c. Select the **Local repository** radio button, then in the Repository name field, enter the path name to where you have extracted the CommerceEnhancementPack_OCTOBER2002.dat code repository on your machine. For example:

<temp>\repository\CommerceEnhancementPack_OCTOBER2002.dat

d. Select the **Projects** radio button and click **Details**. It will open a separate window for Project Import.

e. On the Project import window, select the **IBM WC Commerce Server**, **IBM WC Enterprise Beans**, and **IBM Commerce Enhancement Pack** projects from the project pane and select version from the version available pane for each project. When you have done click **OK**. It will close the window and set the focus back to the SmartGuide.

f. Ensure that the **Add most recent project edition to workspace** check box is selected. Then click **Finish**.

Note: The import on our development system, took 1.5 hours.

10. When the import is complete, ensure that your workspace owner is set to **WC Developer**.

11. Add `com.ibm.commerce.collaboration.livehelp.beans` into the workspace manually by doing the following:

    a. Select the **IBM WC Enterprise Beans** projects from the Project pane. Right-click **Manage -> Create Open Edition**.

    b. Select the **IBM WC Enterprise Beans** projects from the Project pane. Right-click and **Add -> Package**. It will open Add Package SmartGuide.

    c. On Add Package SmartGuide select **Add package for the repository** options. Select **com.ibm.commerce.collaboration.livehelp.beans** package from Available package name pane and select the latest edition checkbox from Available editions. When you have done click **Next**.

    d. When it is complete save your workspace.

12. Configure and run EJB, PNS and Servlet Engine. For details see, “Configure and start the EJB, PNS and Servlet Engine” on page 160.

13. Save your workspace.

3.3.13 WebSphere Commerce Portal Tooling Framework

This section describes the steps to install the WebSphere Commerce Portal Tooling Framework (WebSphere Studio Application Developer plug-in). In addition, this section provides instructions on how to generate new WebSphere Commerce portlets.

With the Commerce Tooling Framework, users can generate new WebSphere Commerce portlets by adding the generated portlet entries to the WebSphere

Commerce Portlet Application deployment descriptor and WAR packaging file descriptor.

Note: For more information on the WebSphere Commerce Portal Tooling Framework, refer to the *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*.

Commerce Tooling Framework installation

To install the Commerce Tooling Framework, do the following:

1. Ensure that WebSphere Studio Application Developer is properly installed.
2. Ensure that the WebSphere Portal Toolkit t is installed.
3. Unzip the `WebSphereCommerceEnabledPortal.zip` file to a temporary directory on your WebSphere Studio Application Developer machine.
4. Go to the `<temp>/Base/WebSphereCommerceEnabledPortalAssemblyTool/` directory.
5. Unzip the `com.ibm.commerce.portal.tooling.zip` to the `<WSAD_HOME>/plugins` directory of your WebSphere Studio Application Developer machine.

   Note: The Portlet Application Assembly Tool relies on the following WebSphere Studio Application Developer resources to function properly:

   ```
   org.eclipse.ui_1.0.2/workbench.jar
   org.apache.xerces/xerces.jar
   org.eclipse.core.resources_1.0.2/resources.jar
   org.eclipse.help/help.jar
   org.eclipse.core.runtime/runtime.jar
   org.ecplise.swt_1.0.2/swt.jar
   org.eclipse.ui/wokbench.jar
   org.eclipse.core.resources/resources.jar
   org.eclipse.swt/swt.jar
   org.ecplise.core.boot/boot.jar
   ```

6. Copy the following file:

   `<temp_dir>/Base/WebSphereCommerceEnabledPortal/bin/WebSphereCommerceBasePortlet.jar`

   To the following location on your WebSphere Studio Application Developer machine:

   `<WSAD_HOME>/plugins/com.ibm.commerce.portal.tooling/data/`

7. Depending on your database software, copy either `db2java.zip` or the `classes12.zip` file to the following directory on the WebSphere Studio Application Developer machine.

   In our example we are using DB2 and copied the db2java.zip from:

   <DB2_HOME>`\java\db2java.zip`

   To the following directory on the WebSphere Studio Application Developer machine:

   <WSAD_HOME>`/plugins/com.ibm.commerce.portal.tooling/`

   Note: You find the database software on your WebSphere Commerce database machine at one of the following locations:

   - *<DB2_HOME>*`/java/`
   - *<Oracle_HOME>*`/ora81/jdbc/lib`

Commerce Tooling Framework configuration

You need to ensure that the Commerce Portal Framework plug-in recognizes the WebSphere Commerce command database in the WebSphere Commerce Server. The plug-ins uses the WebSphere Commerce commands from the database to generate WebSphere commerce portlets.

1. Go to the *<WSAD_HOME>*`/plugins/com.ibm.commerce.portal.tooling/` directory.
2. Open the `plugin.properties` file in a text editor.
3. Change the following information for DB2:
   a. Change the value of the dbdriver:
      - If WebSphere Studio Application Developer is installed on the same machine as WebSphere Commerce, set the value of the dbdriver to:

        ```
        dbdriver=COM.ibm.db2.jdbc.app.DB2Driver
        ```

        Note: We used the `COM.ibm.db2.jdbc.app.DB2Driver` setting for our environment since the database was cataloged.

      - If WebSphere Studio Application Developer is installed remotely from WebSphere Commerce, set the value of the dbdriver to:

        ```
        dbdriver=COM.ibm.db2.jdbc.net.DB2Driver
        ```

   b. Change the value of the `dbconnectionurl` to the following:
      - Local database:

        ```
        dbconnectionurl=jdbc:db2:mall
        ```

Note: In our example, we have a remote database, but since the database is cataloged it should be treated as local. We use the value listed above for dbconnectionurl.

- Remote database:

  ```
  dbconnectionurl=jdbc:db2://<hostname>:<port_number>/<db_name>
  ```

Note: The `<port_number>` is the database JDBC listener port of the database machine. For example, for DB2 the JDBC listener port is `6789`. This is used in the full runtime with remote database.

c. Change the value of `dblogin` to your WebSphere Commerce instance database login name. The user name must have read authority. For example, `db2admin`.

d. Change the value of `dbpassword` to your WebSphere Commerce instance database password.

4. Change the `servletroot` to point to your WebSphere Commerce servlet root. For example, the servletroot is as follows:

   ```
   servletroot=http://<hostname>/webapp/wcs/stores/servlet/
   ```

Note: There are two possibilities for the hostname used in the servletroot:

- Hostname of the node where VisualAge for Java WebSphere Test Environment and WebSphere Commerce are running.
- Hostname of the WebSphere Commerce runtime environment.

In our example, we used the hostname of the node where VisualAge for Java WebSphere Test Environment and WebSphere Commerce are running.

In our example, we did the following:

```
servletroot=http://a23vnx58:8080/webapp/wcs/stores/servlet/
```

Notice the port number `8080` for the VisualAge for Java environment.

5. Change the portletclass to the class that the generated portlets should extend. This is needed for web.xml creation. For example:

   ```
   portletclass=com.ibm.commerce.portal.wpsportlets.WCSServletInvokerPortlet
   ```

6. Change the `LOCALE_PROPERTIES_FILE` to the locale that the plugin-in use to display locale specific settings. For example, `en_us`. You must ensure that the `properties_`*`locale`*`.properties` resource bundle exists for the following file:

<*WSAD_HOME*>/plugins/com.ibm.commerce.portal.tooling/commerceportalaat.jar

Note: The locale settings is case sensitive. In our example, en_us is lower case.

7. Save your changes.

Note: The information you entered in Step 2 to Step 6 depends on your environment. For example, we used the, “Multi-node development environment separate portal runtime” on page 99 approach. Where we have installed Commerce Studio and WebSphere Commerce Studio in Development Tools node and DB2 V7.2 Server, WebSphere Application Server Advanced Single Server Edition and WebSphere Portal in WebSphere Portal development test node (see in Figure 3-5 on page 101). We catalog our Mall database in Development Tools node. See in Example 3-2 for details of our `plugin.properties file after modification.`

Example 3-2 Sample plugin.properties file

```
pluginName=Commerce Portal Tooling
dbdriver=COM.ibm.db2.jdbc.app.DB2Driver
dbconnectionurl=jdbc:db2:mall
dblogin=db2admin
dbpassword=db2admin
servletroot=http://<hostname>:8080/webapp/wcs/stores/servlet/
portletclass=com.ibm.commerce.portal.wpsportlets.WCSServletInvokerPortlet
listenerclass=com.ibm.commerce.portal.wpsportlets.RemoteServlerInvokerPortletAc
tionListener
LOCALE_PROPERTIES_FILE=en_us
```

8. Restart WebSphere Studio Application Developer.
9. After WebSphere Application Developer restarts, click **Perspective > Show View > Other** menu. The Show View dialog opens.
10. Expand **Commerce Portal Tooling Views** node. Select **WCS Commands** node and click **OK**. The WebSphere Commerce commands view opens as seen in Figure 3-38.

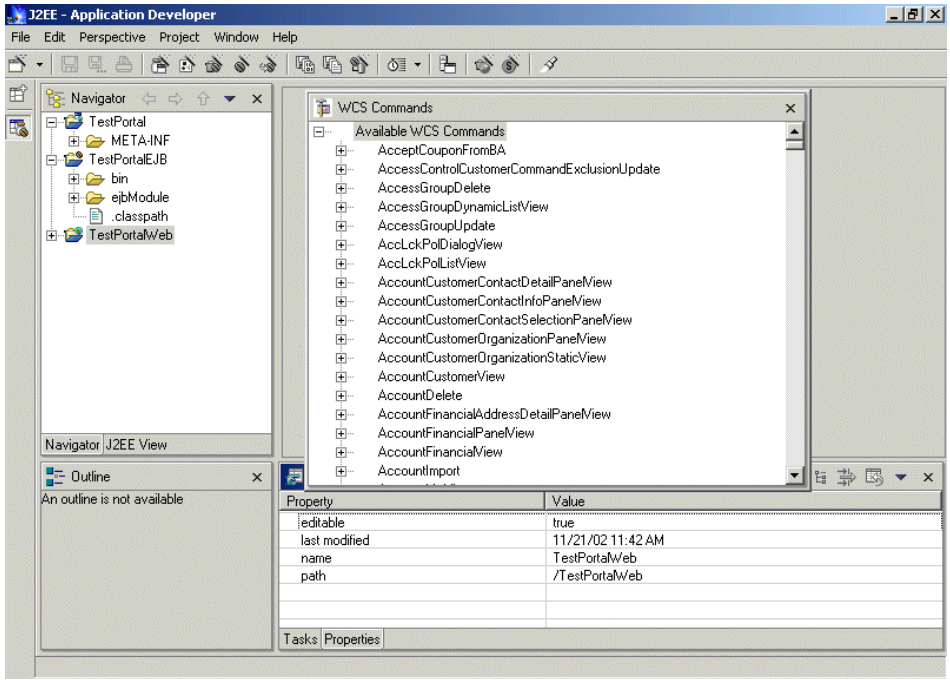


Figure 3-38 Commerce Portal Tooling View for WebSphere Commerce commands

3.3.14 CEP installation on WebSphere Application Server AEs

This section describes how to install the Commerce Enabled Portal included with the IBM Commerce Enhancement Pack - October 2002 Edition, on the WebSphere Application Server V4.0.2, Advanced Single Server Edition. This procedure was devised by the ITSO to facilitate the debug of commerce enabled portlets within WebSphere Studio Application Developer.

To install Commerce Enhancement Pack on WebSphere Application Server AEs, perform the following high level steps on the WebSphere Portal development test node:

1. Enable SSL and import the SSL certificate.

> **Note:** If you are using the VisualAge for Java WebSphere Test Environment and you turned off your SSL, then this step is not required.

To enable SSL and import the certificate, do the following:

– Extract the certificate from the commerce node:

Refer to, “Extract certificate from WebSphere Commerce Web server” on page 76.

– Import the certificate:

Refer to, “Certificate trust security configuration” on page 76.

2. Extract the `WebSphereCommerceEnabledPortal.zip` file provided with Commerce Enhancement Pack to c:\temp\CEP directory.
3. Install the Commerce Enabled Portal Enterprise Application.
   a. Open a command prompt and change to the following directory:

      `C:\temp\CEP\Base\PersonalizationUserHomePageBaseFolder\scripts`

   b. Create the CPS database by running the `createCPSDB.bat` file at the command prompt. You will be prompted with the following questions:

      - Enter DB2 Admin UID [db2admin]: `<db2admin_user>`

        If your database user ID is same as `db2admin` then press Enter key. Otherwise type the correct user id for DB2 admin and press Enter key.

      - Enter Your Host Name [host]: `<wasaes_hostname>`

        If your `host` name is host then press enter. Otherwise type your hostname and press Enter key.

      - Enter password for <host>\<db2admin_user>: `<password>`

        Now it will ask for DB2 Admin user’s password. Type the password and press Enter key. This will launch a DB2 command line program (CLP) session to create the CPS database.

   c. When process complete, review the createPSDB.log file found in the following directory:

      `c:\temp\CEP\Base\PersonalizationUserHomePageBaseFolder\logs`

4. Create CPS datasource in WebSphere Application Server for the Commerce Enabled Portal Enterprise Application.

Note: To create a datasource, the IBM Commerce Enhancement Pack - October 2002 Edition provides the createCPSDS.db2.bat script file, which only works if you installed WebSphere Application Server Advanced Edition and uses the XML loader.

In our environment we use the WebSphere Application Server Advanced Single Server Edition (required for debug) does not include the XML loader.

The following procedure describes how to create a JDBC driver datasource for the WebSphere Application Server Advanced Single Server Edition using the Administrator’s Console.

a. Click **Start -> Programs -> IBM WebSphere -> Application Server V4.0 AES -> Start Application Server**.

b. When Server start process complete, then open admin console from **Start -> Programs -> IBM WebSphere -> Application Server V4.0 AES -> Administrator's Console**. Console will open in a Web browser window. We entered `wasadmin` and then clicked **Submit** (this user can be any user ID).

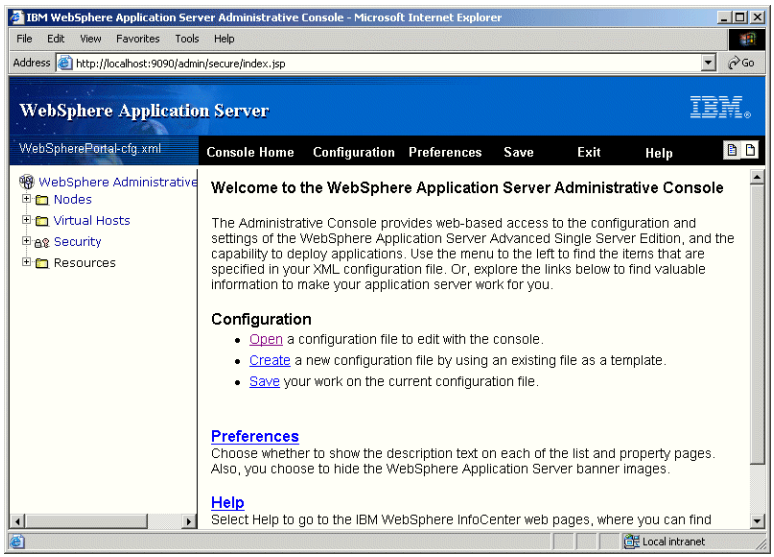


Figure 3-39 WebSphere Application Server Administrator's Console

c. Click on **Open** link under **Configuration** in main window. Do the following and when done click **OK** (see Figure 3-40).

- Open configuration File: select **Select from files in "Config" Directory** radio button
- From the pulldown, select **WebSpherePortal-cfg.xml**.

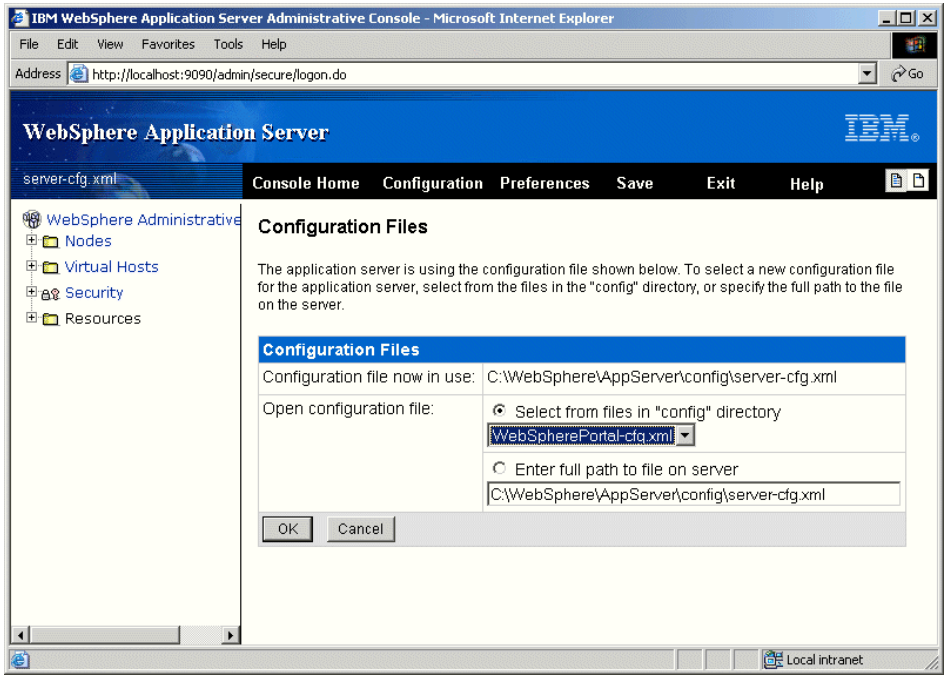


Figure 3-40 Configuration files

d. Click **Save** to save your current configuration file.

e. Expand the **Resources** from the tree in left panel. Click on **JDBC Drivers** link to create a new driver. Click **New** in main window.

f. Select **DB2 JDBC Driver** from the pull-down list for Resource Provider Type and Click **Next**.

g. Enter the followings and when done click **OK** (see Figure 3-41):

- Server Class path: `<DB2_HOME>\java\db2java.zip`
- Name: `CPSJDBCDriver`
- Description: `CPS DB2 JDBC2-compliant Driver`
- Implementation Class Name: `COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource` (default)

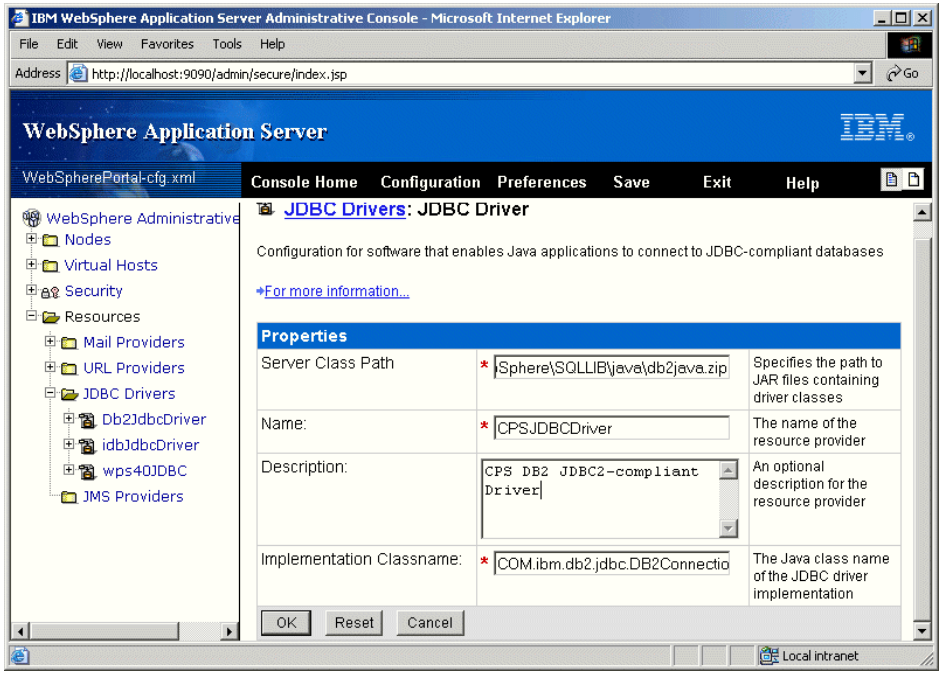


Figure 3-41 Configuration for new JDBC driver

h. Expand **Resources -> JDBC Drivers -> CPSJDBCDriver** and click on **Data Sources** link. Where `CPSJDBCDriver` is the name you have entered on previous screen. Then click on **New** in main window.

i. Enter the following and click **OK** when you have done (see Figure 3-42):

- Name: `cpsDS`
- JNDI Name: `jdbc/cps`
- Description: `CPS DB2 Data source`
- Category: (optional, we left this field blank)
- Database Name: `CPS`
- Default User ID: `db2admin`
- Default Password: `db2admin`
- Minimum Pool Size: 1
- Maximum Pool Size: 10
- Connection Timeout: `120000`
- Idle Timeout: `180000`
- Orphan Timeout: `1800000`
- Statement Cache Size: `100`
- Disable Auto Connection Cleanup: (do not check)

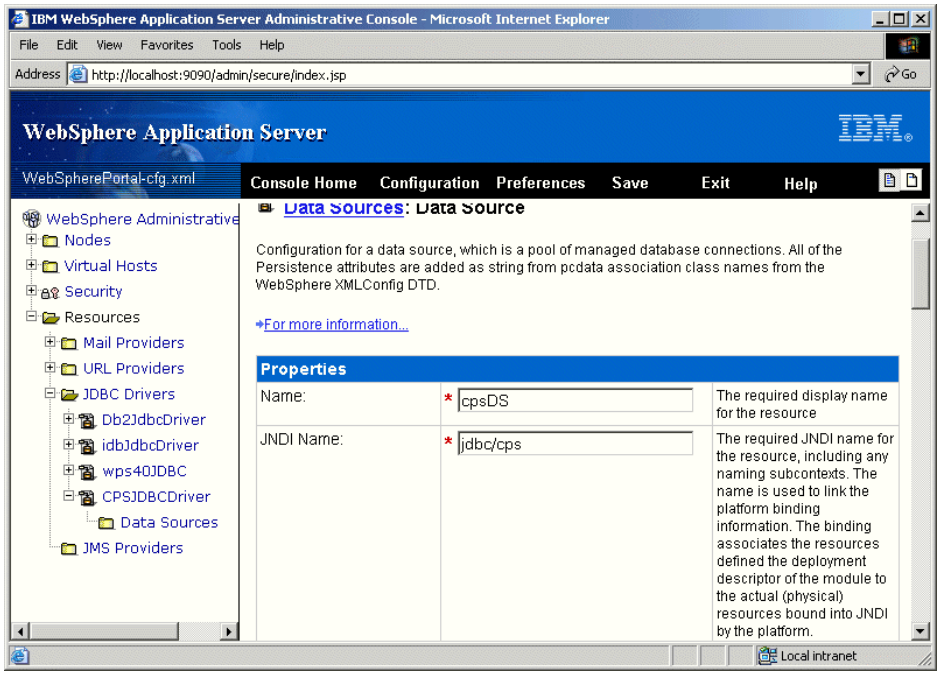


Figure 3-42 Data source configuration

j. Click on **Configuration needs to be saved** link or **Save** menu from the top panel.

k. On Save Configuration page, accept the default setup and click **OK**.

5. Install the application on the WebSphere Application Server AEs.

   a. Create a `createCPSEA_AEs.bat` script file to install the CPS application in WebSphere Application Server (see Example 3-3 on page 177).

      You update the values of the following variables in the createCPSEA_AEs.bat:

      - NODE
      - INSDIR
      - WASPATH
      - INSPATH

      Note: The NODE value is case sensative. For example, if your WebSphere Application Server Administrative Server node name is all upper case, the NODE value must be upper case.

Example 3-3 ITSO sample createCPSEA_AEs.bat

```
@echo off
REM WAS node name
set NODE=portaldev
REM The path where CPS will install.
set INSDIR=C:\WebSphere\CPS\app\cps.ear
REM The path where WebSphere Application Server Advanced Single Server Edition
was installed
set WASPATH=C:\WebSphere\AppServer
REM The CPS installation archive location.
set INSPATH=..\bin\CPSEAP.ear
echo Installing Commerce Portal Server...
call %WASPATH%\bin\SEAppInstall.bat -install %INSPATH% -nodeName %NODE%
-expandDir %INSDIR% -configFile %WASPATH%\config\WebSpherePortal-cfg.xml

set INSPATH=
set WASPATH=
set INSDIR=
set NODE=
```

b. Copy the createCPSEA_AEs.bat file created in the previous step to the following directory:

   `C:\temp\cep\Base\PersonalizationUserHomePageBaseFolder\scripts`

c. Open a command prompt and change to the following directory:

   `C:\temp\cep\Base\PersonalizationUserHomePageBaseFolder\scripts`

d. Set the TEMP and TMP location to the <temp> directory where you have unpacked the `WebSphereCommerceEnabledPortal.zip`. For example:

   ```
   set TEMP=C:\temp
   set TMP=C:\temp
   ```

e. From the command prompt type `createCPSEA_AEs.bat` to run the batch file.

f. When prompted, **"**`Do you wish to deploy all of the EJBs in this application ([Y]es/[n]o)?`**"** type `Y` and press Enter key (see Figure 3-43).

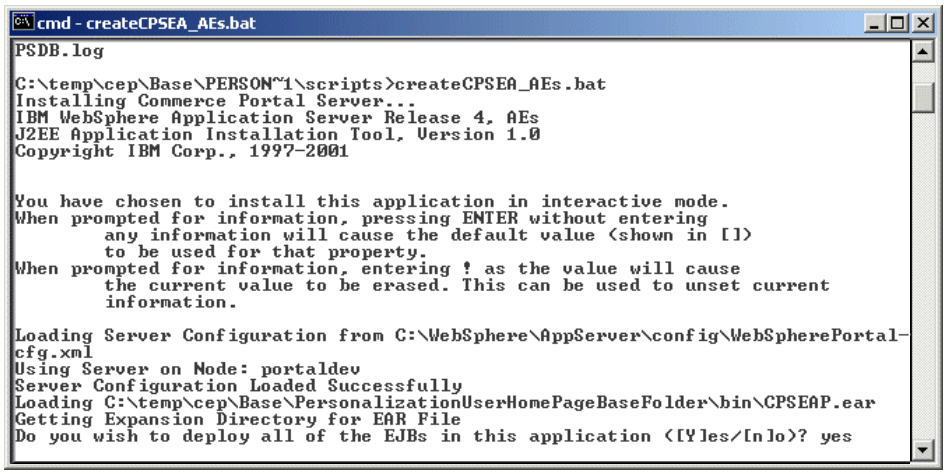


Figure 3-43 Running the createCPSEA_AEs.bat file and EJB deployment option

g. Type 1 (IBM DB2) and press Enter key when asking “Which type of database are you using?”. Press Enter key.

h. When prompted with, “What DB Schema name do you want to use for this application?” press Enter (In our example, we do have a schema).

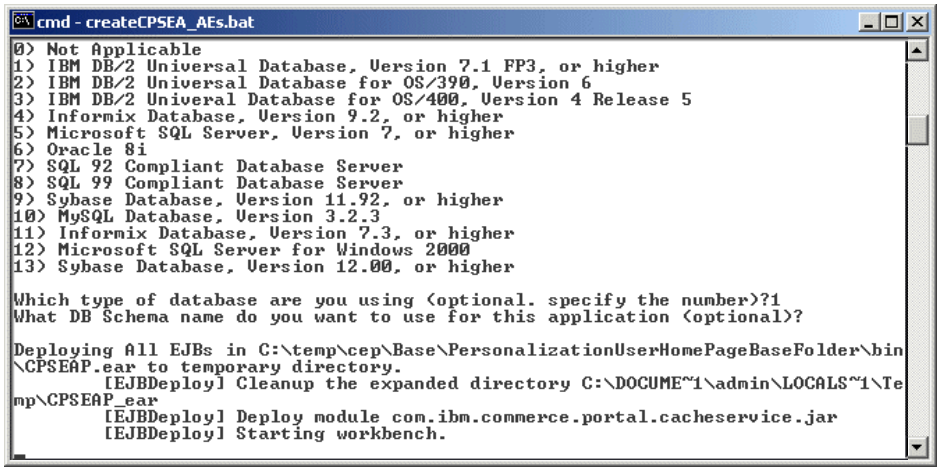


Figure 3-44 Database type and schema

Note: If you did not use IBM DB/2 Universal Database, Version 7.2 FP7 then follow the screen for other options.

i. Now it will start to deploy all required EJBs. When done, it will bind the following EJB jar files:

- com.ibm.commerce.portal.personalization.eapEJB.jar
- com.ibm.commerce.portal.personalization.beans.CommercePortalPersonalizationContentgrpBean
- com.ibm.commerce.portal.personalization.beans.RoleacBean
- com.ibm.commerce.portal.cacheservice.jar
- com.ibm.commerce.portal.cacheservice.beans.CacheBean

For each item, you will be required to enter the following:

- JNDI Name: `jdbc/cps`
- Datasource JNDI name: `jdbc/cps`
- User ID: `db2admin`
- Password: `<db2admin_password>`

For each question type appropriate value and press Enter key. For example in ITSO we have enter the values according to Figure 3-45.

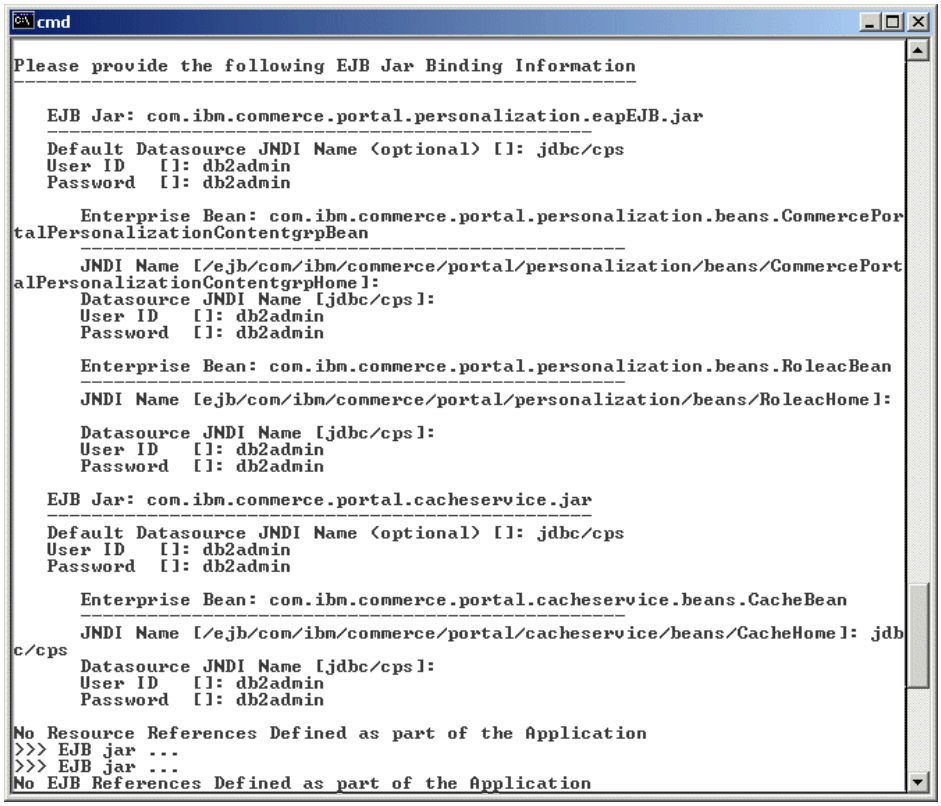


Figure 3-45 EJB bindings

j. For successful installation you will see the information as in Figure 3-46.

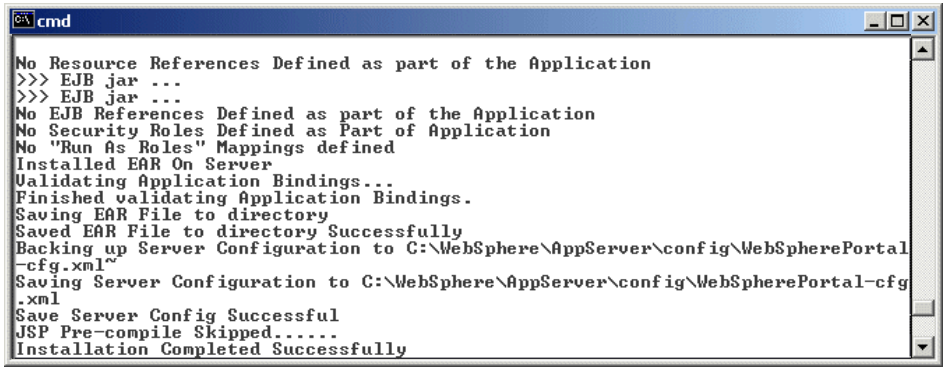


Figure 3-46 Successful installation message

6. Run the `modifyWAS.bat` script from the command line.

   When prompted "`Enter WAS Path`" type appropriate WebSphere Application Server Advanced Single Server Edition installation path (if default value is not correct) and press Enter key. It will copy three files as shown in Figure 3-47.

7. Run the `installNewWPSAttributes.bat (optional)` from the command line.

   a. When prompted `Enter Path to WebSphere Application Server enter the` appropriate WebSphere Application Server AEs install path and press Enter.

   b. When prompted `Enter Path to WebSphere Portal Server` enter the appropriate WebSphere Portal Server instal path and press Enter . It will extract JSP files as displayed in Figure 3-47.

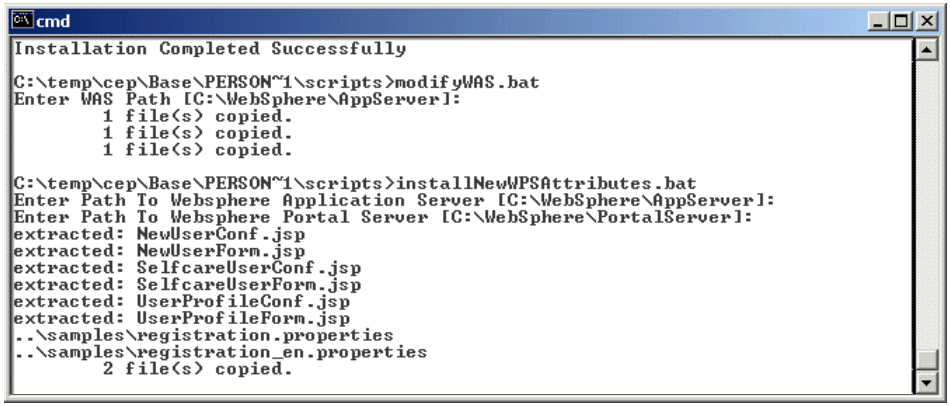


Figure 3-47 Scripts output

8. Verify that WebSphere Portal is functioning properly.
   a. Open a command line session and change to the <WAS_HOME>\bin directory.
   b. Stop WebSphere Application Server if it is running:

      ```
      stopServer
      ```

   c. Start the server with the portal config:

      ```
      startServer -configFile ..\config\WebSpherePortal-cfg.xml
      ```

   d. Access the WebSphere Portal Server by entering the following URL in a Web browser:

      ```
      http://<hostname>:9080/wps/portal
      ```

   e. Verify that you can log in as `wpsadmin`.

Note: At this stage you will not be able to log on with user IDs registered through WebSphere Commerce, because there is no single sign-on capability in the development environment.

3.3.15 Deploy the ITSO CEP B2B sample

After completing the configuration of the WebSphere Portal for WebSphere Commerce, we recommend that you publish the ITSO B2B CEP store to the development environment. Refer to 5.2, “Deploy the ITSO B2B CEP store to the development test environment” on page 229 for details.

Note: For more information, refer to *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*.

4

Create a commerce enabled portal store

The chapter describes how to use WebSphere Studio Application Developer to import a store template, customize, package, and publish the ITSO B2B CEP store.

The chapter is organized as follows:

- ITSO sample code
- WebSphere Studio Application Developer setup
- Create the ITSO B2B CEP store template
- Customize the ITSO B2B CEP store
- Package SAR and publish Web assets

4.1 ITSO sample code

This Redpaper provides a sample code zip file containing the artifacts developed for the Redpaper. These samples will be referenced throughout the remaining example chapters.

4.1.1 Procedure to download ITSO sample code

To download the redp3684.zip file, complete the following steps:

1. Enter the following URL in a Web browser:

   `ftp://www.redbooks.ibm.com/redbooks/redp3684`

2. Download the redp3684.zip to a directory on your system (for example, c:\temp).
3. Unpack the redp3684.zip (for example, c:\redp3684-code).

Note: We will refer to this base unpack directory throughout the working example as c:\redp3684-code.

4.1.2 Description of ITSO sample code

The sample code zip file includes a WebSphere Studio Application Developer workspace containing all artifacts developed, ITSO modified SAR file, configuration files, and an ITSO provided portlet jar file for debug purposes.

Table 4-1 describes the contents of the ITSO sample code zip file (redp3684.zip).

Table 4-1 ITSO sample code

Directory/Filename	Description
c:\redp3684-code	ITSO sample code root directory
c:\redp3684-code\wsad	ITSO WebSphere Studio Application Developer workspace containing the ITSO B2B CEP Web assets, XML data files, and Ant scripts for packaging a SAR and publishing.
c:\redp3684-code\sar	ITSO B2B CEP store SAR file containing ITSO modified assets.
c:\redp3684-code\ldif	WebSphere Commerce and WebSphere Portal sample ldif files

Directory/Filename	Description
c:\redp3684-code\wasaes_config	Script file used to deploy CPS application within the WebSphere Application Server AEs development environment.
c:\redp3684-code\debug	* ITSO modified version of the WebSphereCommerceBasePortlet.jar to include logon support without single sign-on for debug purposes. * WCSLogon.properties file containing debug WCSPassword and WCSLogonURL. **Note**: This should only be used for debug and development, not production use. This file is not supported by IBM WebSphere Commerce.

4.2 WebSphere Studio Application Developer setup

This section describes the steps to setup WebSphere Studio Application Developer for developing WebSphere Commerce web assets.

4.2.1 WebSphere Studio Application Developer installation

When installing the WebSphere Studio Application Developer V4.0.3, we entered the following:

- Changed the target installation path to `c:\ibm\wsad`
- Primary User Role: **J2EE Developer**

Note: The procedure listed in this section requires that you have installed the WebSphere Commerce Studio environment or WebSphere Commerce runtime environment. We will use files included as part of the runtime environment in the configuration of the development environment.

4.2.2 WebSphere Studio Application Developer configuration

This section describes how we created the WebSphere Studio Application Developer workspace for developing WebSphere Commerce assets. The assets modified are as follows:

- Images
- JSPs for native WebSphere Commerce stores
- JSPs for commerce enabled portlets including HTML and WML
- Properties
- XML files
- Extending WebSphere Commerce commands

At the time of writing this Redpaper, WebSphere Commerce did not officially support WebSphere Studio Application Developer. WebSphere Studio Application Developer can be used to develop front-end assets such as JSPs or XML, but it does not provide a WebSphere Commerce Test Environment. In our example, we use WebSphere Studio Application Developer to develop, compile, package any deploy custom developed assets as part of a store development effort.

The WebSphere Studio Application Developer workspace included in the c:\redp3684-code contains one Web project and one Enterprise Application project for the ITSO B2B CEP store. The ITSO workspace contains the modified JSPs, XML and ant scripts described in this chapter.

Modify launch.bat

Provided you have unpacked the redp3684.zip to c:\redp3684-code, as described in 4.1, “ITSO sample code” on page 184, modify the launch.bat.

Modify the `launch.bat` file found in the c:\redp3684-code\wsad\cep directory to include the install path of WebSphere Studio Application Developer as seen in Example 4-1.

Example 4-1 Sample WebSphere Studio Application Developer launch.bat

```
rem set the variable WSAD to the WSAD install path
set WSAD=C:\ibm\wsad
start %wsad%\wsappdev.exe -jvmargs -Xmx512m -Xms128m
```

Configure WebSphere Studio Application Developer

This section describes how to configure WebSphere Studio Application Developer to create the workspace as done by the ITSO for this Redpaper.

1. Double-click on c:\redp3684-code\wsad\cep\launch.bat.
2. Select **File -> New -> Web Project**. Click **Next**.

> **Note:** The procedure outlined is optional for creating a new Web project. In our example workspace, we have provided a completed version.

3. In the Create a Web Project window, enter the following and then click **Finish**.
   - Project Name: `b2bCepWeb`
   - Check **default location**
   - Enterprise Application project name: `b2bCepEAR`
   - Context root: `b2bCepWeb`

4. Configure the Java build path for the Web application to include the needed WebSphere Commerce libraries. These are the <WC_HOME>\wc.ear\lib directory. This assumes that these files are installed on the same node as WebSphere Studio Application Developer.
5. Set up the b2bCepWeb projects to use the WebSphere Commerce tag library `flow.tld`.
   a. Copy the `flow.tld` from `<WC_HOME>\wc.ear\wcstores.war\WEB-INF` directory to `c:\cep\b2bCepWeb\webApplication\WEB-INF` directory.
   b. In WebSphere Studio Application Developer right-click on the **b2bCepWeb** project and from the context menu, select **Refresh From Local**.
   c. In the WebSphere Studio Application Developer Navigator window, double-click on the **web.xml** file under: `b2bCepWeb\webApplication\WEB-INF`.
   This should bring up the Web.xml editor.
   d. On the References tab, select **JSP tag libraries** and select **Add**.
   e. In the URI column, replace (New URI) with `flow.tld` and for the Location, select the drop down arrow on the right side of the column and from the dialog, select **flow.tld**.
   f. Save web.xml.
6. Add the WebSphere Commerce EJBs to the WebSphere Studio Application Developer Build Path.
   a. From the Window drop down menu, select **Preferences**.
   b. Expand the Java element and select Classpath Variables.
   c. Select **New...** and enter `WC` as the Name and select the **Folder...** button and navigate to the `<WC_HOME>\wc.ear` and select **OK** to close the Folder Selection dialog.
   d. Select **OK** to close the Edit Variable Entry dialog.
   e. Select **OK** again to close the Preferences window.
   f. Right-click on the **b2bCepWeb** project and from the context menu select **Properties**.
   g. Select the **Java Build Path** element and then select the **Libraries** tab.
   h. Select **Add Variable** and for the Variable Name enter `WC`.
   i. Then select **Browse** next to the Path Extension text box and select **WCSCatalog-ejb.jar**.
   j. Click **Open**, then select **OK** to add the JAR to the build path.
   k. Repeat the previous step for all EJB JARs and all JARs in the lib directory.

l. After all of the JARs have been added click **OK** to close the properties window.

The list of JARs above reflects all JARs that we need during the development of the store. This allows us to compile and package the assets and deploy them to the wcstores Web application. It is possible that you may need additional JARs added to the build path for your development effort. To do this follow the steps outlined above for the JARs that you need.

4.2.3 Procedure for importing other sample stores (optional)

If you would like to use one of the other sample stores such as ToolTech from the IBM Commerce Enhancement Pack - October 2002 Edition, complete the following steps.

A folder structure was created to accommodate a repeatable build process. This included a sar folder under the b2bCepWeb project for only those artifacts that are included in the archive.

1. Create an `ant` folder under the `b2bCepWeb` folder.
2. Copy all files from the CEP sample `ant` directory to `c:\cep\b2bCepWeb\ant directory.`
3. Update all drive letters and directories in the `cep.config.properties` file, `cep.vaj.properties` file, and `cep.wcs.properties` file of the `c:\cep\b2bCepWeb\ant` directory.
4. Extract all contents of ToolTech_en_US_es_ES.sar found in the <WC_HOME>\samplestores\tooltech directory to the c:`\cep\b2bCepWeb\sar` directory.

   Note: The Commerce Enhancement Pack comes with a new ToolTech example store. You must copy the new version to WebSphere Commerce. Follow the instructions given by the Commerce Enhancement Pack.

5. Open the `webapp.zip` under c:`\cep\b2bCepWeb\sar` and extract the contents of this zip file to the `c:\cep\b2bCepWeb\webApplication` directory.
6. Create a new folder named `properties` under the `c:\cep\b2bCepWeb\webApplication directory.`
7. Open the `properties.zip` under c:`\cep\b2bCepWeb\sar` and extract the contents of this zip file to the `c:\cep\b2bCepWeb\webApplication\properties` directory.
8. Open the `runtime_xml.zip` under c:`\cep\b2bCepWeb\sar` and extract the contents of this zip file to the `c:\cep\b2bCepWeb\sar` directory.

9. Open the `tools_properties.zip` under c:\cep\b2bCepWeb\sar and extract the contents of this zip file to the `c:\cep\b2bCepWeb\sar` directory.
10. Open the `tools_xml.zip` under c:\cep\b2bCepWeb\sar and extract the contents of this zip file to the `c:\cep\b2bCepWeb\sar` directory.
11. In WebSphere Studio Application Developer, right-click on the **b2bCepWeb** project and from the context menu select **Refresh from Local**.

> **Note:** After importing the store archive into WebSphere Studio Application Developer, you may see errors in the task window. If you wish to turn of validation so that these errors are not displayed, do the following:
>
> - Right-click on the **b2bCepWeb** project and from the context menu.
> - Select **Properties**.
> - In the properties window, select **Validation** and deselect all of the Validation options.

12. Import the DTD files to be used to validate the store XML files. This step is optional, but highly recommended if you are modifying the XML files. The DTDs can be retrieved from the `<WC_HOME>/xml/sar` directory and should be copied to `c:/cep/b2bCepWeb/sar/data` directory.

Configuring WSAD for editing large XML files

While writing this Redpaper, we discovered that opening a large WebSphere Commerce catalog XML data file with the XML editor included with the WebSphere Studio Application Developer V4.0.3 and V5 causes WebSphere Studio Application Developer to terminate. The problem seems to be with the XML Editor design view.

This problem has been reported and the following work-around has been provided.

1. From the main menu in WebSphere Studio Application Developer, select **Windows -> Preferences**.
2. Select **Workbench -> File Associations**.
3. Select the ***.xml** file.
4. From the Associated editors list, click **Add** an the **Source Editor**.
5. Click **Default** to make the Source Editor the default editor.

Now when opening an xml file, the Source Editor will be used by default. The Design View and several toolbar icons will be absent, but the editor will still provide an outline and source view content assist.

4.2.4 Jakarta Ant overview

Another benefit to using WebSphere Studio Application Developer is it's ability to work with 3rd party components like Jakarta's Ant. Ant is powerful build tool that assists users in the creation, packaging and deployment of J2EE Enterprise Applications. We used Ant in our sample scenario to automate the creation and deployment of the sample store archive. We modified the necessary XML configuration files for the store archive and then configured Ant to package and deploy the end result. Ant version 1.3 is shipped with WebSphere Studio Application Developer and already configured and ready to use when you start up your workspace.

For information on Ant, refer to the following URL:

`http://Jakarta.apache.org/ant`

4.3 Create the ITSO B2B CEP store template

This section describes how we created the ITSO B2B CEP store template. The workspace provided in the redp3684-code.zip contains a the b2bCepWeb project with these changes already completed.

4.3.1 Update core data

The core data that was modified to create the ITSO B2B CEP store includes the following:

- Modify store.xml
- Modify the distinguished name (DN)

Note: When creating a template, we recommend that you update the core data files that contain information that will be common to store built from the template.

Refer to the *Store Developer's Guide, IBM WebSphere Commerce V5.4* or the *WebSphere Commerce V5.4 online documentation* for details on all core data assets. In addition, refer to the *WebSphere Commerce V5.4 Catalog Design and Content Management*, SG24-6585.

Modify store.xml

The core data that was updated for the ITSO B2B CEP store is located in the `store.xml` file. We modified the storeent_id and the store directory.

What uniquely identifies a store is a combinitation of the store identifier and the store owner. We modified the store identifier attribute to ensure that our store is

unique (see Example 4-2 for the storeent entry). For a complete listing, please refer to the store.xml that is available in the downloaded ITSO sample workspace, found in the `b2bCepWeb\sar\data` directory.

Example 4-2 XML element <storeent> in store.xml (identifier=b2bCep)

```
<storeent storeent_id="@storeent_id_1"
    member_id="&MEMBER_ID;"
    type="S" identifier="b2bCep"
    setccurr="USD"/>
```

Update the directory of the store (for example, directory=b2bCep as seen in Example 4-3).

Example 4-3 XML element <store> in store.xml (directory=b2bCep)

```
<store store_id="@storeent_id_1" directory="b2bCep"
ffmcenter_id="@ffmcenter_id_1" language_id="&en_US;" storegrp_id="-1"
allocationgoodfor="43200" bopmpadfactor="0" defaultbooffset="2592000"
ffmcselectionflags="0" maxbooffset="7776000" rejectedordexpiry="259200"
rtnffmctr_id="@ffmcenter_id_1" pricerefflags="0" storetype="B2B"/>
```

Note: You must edit all occurences of the *identifier* and *directory* in the SAR directory and language dependant sub directories. For example, the en_US and es_ES must be updated too.

Modify the distinguished name (DN)

There are several data files included in the ITSO B2B CEP store SAR (b2bCep_en_US_es_ES.sar) or the WebSphereB2CCommerceEnabledPortal.sar included with the IBM Commerce Enhancement Pack - October 2002 Edition, that require modifications for the distinguished name, including contract.xml, businessaccount.xml, and organization.xml.

The ITSO provided WebSphere Studio Application Developer workspace (c:\redp3684\wsad\cep) contains a contract.xml.LDAP, businessacount.xml.LDAP, and organization.xml.LDAP file with the distinguished name as needed for LDAP.

Note: For more detailed information, reference Technote #1083258 at:

http://www-1.ibm.com/support/docview.wss?rs=494&context=SSZLC2&q=publish&uid=swg21083258&loc=en_US&cs=utf-8&lang=en

To address this problem, do the following:

1. Check the Distinguished Name (DN) for the wcsadmin user in the USERS database table as follows:
   a. Start a DB2 command window.
   b. Connect to the WebSphere Commerce instance database.

      ```
      db2 connect to wc1db
      ```

   c. Query the USERs table.

      ```
      db2 select * from USERS
      ```

   d. Record the DN for wcsadmin (for example, `uid=wcsadmin,dc=ibm,dc=com`).
2. Ensure that the contract.xml file in the SAR has the same DN. If the DN is different, change the DN in the contract.xml to reflect that of the wcsadmin in the database.

   In our case, are using LDAP and we have a different DN for wcsadmin than the default contract.xml.

   For example, you may have uid=wcsadmin,dc=ibm,dc=com. However, the contract.xml has <User distinguishName ="uid=wcsadmin,o=Root Organization" />. You need to change the contract.xml to <User distinguishName ="uid=wcsadmin,dc=ibm,dc=com" /> in order to successfully publish the Sample Commerce Portal. The contract.xml file is included within the SAR file.

Note: The b2bCepWeb\sar\data\contract.xml.LDAP in the ITSO provided workspace contains the LDAP distinguished name. Copy contract.xml.LDAP to contract.xml as a starting point and modify the value of the distinguished name as described.

4.3.2 Update configuration data

Note: For the ITSO B2B CEP store SAR, the contract.xml can be easily modified in the WebSphere Studio Application Developer project, and then repackaged using the Ant script provided called ant_create_sar.xml.

In our example scenario, we are focused on the data assets and do not need to modify the command registry, view registry, URL registry for our sample store.

When developing a customized store, it is very common that you will create your own JSPs and commands. When this is done, you will need to update the following configuration data:

- Command registry (CMDREG table)
- View registry (VIEWREG table)
- URL registry (URLREG table)

For more information on how to update the configuratoin data, refer to the following:

- *WebSphere Commerce V5.4 online documentation*
- Product Guide *Programmer's Guide, IBM WebSphere Commerce V5.4*
- *WebSphere Commerce V5.4 Developers Handbook*, SG24-6190

4.4 Customize the ITSO B2B CEP store

This section outlines the steps to enable existing JSPs for commerce enabled portlets in the ITSO B2B CEP store.

Contraints

The JSPs of a commerce enabled portal must be modified to work properly with the WebSphere Portal. There are several restrictions and constraints you have to address within the JSPs:

- JavaScript naming convention
- Form naming conventions
- Stripping HTML tags
- Enable Portal URI
- Developing commerce enabed portlets

4.4.1 JavaScript naming convention

There is a one-to-many relationship between the WebSphere Commerce Portal portlets JSPs and the commerce enabled portlets. For example, several Product Display portlets can be aggregated, each accessing the `ProductDisplay.jsp`. Within the JSP a set of JavaScript elements can be defined. This leads to the circumstance that the JavaScript code is included multiple times within the generated HTML output of the portal page, which can create problems for a Web browser client.

If you are using JavaScript elements in the JSP, ensure that during the generation process the elements are uniquely defined. This can be solved as follows:

- Timestamp approach

  Concatenate the JavaScript with a timestamp object to uniquely identify the Script for the generated portlet.

Note: You can call the `now` Method from the `com.ibm.commerce.utils.TimestampHelper` class to retrieve a timestamp value.

- Additional key/value pairs

  Pass additional key/value pairs to the JSP (for example, portlet ID). Concatenate the JavaScript name with the value of the key object to uniquely identify the JavaScript for the generated portlet.

Timestamp approach

The following section describes the modifications to enable JavaScript elements across commerce enabled portlets as follows:

1. Open the `portal_jsp/include/GetResource.jsp` file in a text editor.
2. Add the following statement:

   ```
   int lTimeStamp = com.ibm.commerce.utils.TimestampHelper.now().getNanos();
   ```

   Note: In order to use the `lTimeStamp` variable each portal JSP must include the `include/GetResource.jsp` file.

3. Save the file.
4. For each JSP, do the following:
   a. Open your JSP which contains the JavaScript function
   b. Rename each JavaScript function as seen in Example 4-4.

Example 4-4 Modified submitSearch JavaScript function

```
function submitSearch<%=lTimeStamp%>(form)
{
   <% /* Show products & items if a SKU number is entered.
       * (A SKU number is unique in a store)
       * Only list items on the result page if 'manufacturer name' or
       * 'manufacturer part number' fields are input.
       */ %>
   if (form.sku.value !="")
   {
      form.resultType.value = "3"
   }
   else if (form.manufacturer.value != "" ||
      form.manufacturerPartNum.value != "")
   {
      form.resultType.value = "1"
   }
```

```
        form.submit()
    }
```

c. Save the JavaServer Page.

4.4.2 Form naming conventions

There is a one-to-many relationship between the WebSphere Commerce Portal portlet JSPs and commerce enabled portlets. For example, several Search portlets can be aggregated, each accessing the `AdvancedSearch.jsp`. Within the JSP a set of form elements can be defined. This leads to the circumstance that the generated code includes mutliple forms of the same name within the generated HTML output of the portal page, which can create problems for a Web browser client.

If you are using form elements in the JSP make sure that during the generation process the form elements are uniquely defined. This can be solved as follows:

- Timestamp approach

  Concatenate the form name with a timestamp object to uniquely identify the Script for the generated portlet.

  Note: You can call the `now` Method from the `com.ibm.commerce.utils.TimestampHelper` class to retrieve a timestamp value.

- Additional key/value pairs

  Pass additional key/value pairs to the JSP (for example, portlet ID). Concatenate the form name with the value of the key object to uniquely identify the JavaScript for the generated portlet.

Timestamp approach

The following section describes the modifications to enable HTML FORM tags across commerce enabled portlets as follows:

1. Open the `portal_jsp/include/GetResource.jsp` file in a text editor.
2. Add the following statement:

   ```
   int lTimeStamp = com.ibm.commerce.utils.TimestampHelper.now().getNanos();
   ```

   Note: In order to use the `lTimeStamp` variable each portal JavaServer Page must include the `include/GetResource.jsp` file.

3. Save the file.

4. For each JavaServer Page do the following:
   a. Open your JavaServer Page which contains the JavaScript function.
   b. Rename each FORM as follows:

Example 4-5 Modified AdvancedSearchForm HTML FORM tag

```
<form method="post" action="<%=sPortalURI%>"
name="AdvancedSearchForm<%=lTimeStamp%>">
```

   c. Save the JavaServer Page.

4.4.3 Stripping HTML tags

Since the WebSphere Portal generates the HTML/WML output for the browser you have to strip the following HTML tags:

- <HEAD> and </HEAD> tags
- <DOCTYPE> tag
- <BODY> and </BODY> tags
- <META> tags

For more information, refer to the Appendix C. “Converting Existing JSPs” of the *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*. Ironically the B2B CEP sample provided with the IBM Commerce Enhancement Pack - October 2002 Edition does not follow the guidelines.

4.4.4 Enable Portal URI

To enable the portal URI within the commerce portlet JSP, do the following:

1. To support commerce enabled portal functionality each portlet JSP must have the following preamble before any output is produced by the JSP file:

Example 4-6 Sample for enabling portal URI within commerce portlet JSP

```
<%
String sURLPrefix=request.getScheme()+"://"+request.getServerName()+ “:” +
request.getServerPort() + request.getContextPath()+"/servlet/";
String sURLImgPrefix=request.getScheme()+"://"+request.getServerName()+ “:”
+
request.getServerPort();

String
sPortalURIParam=com.ibm.commerce.portal.exports.CommercePortalConstants.S_P
ORTLET_URI_PARAM;
```

```
String
sRemoteServletURIParam=com.ibm.commerce.portal.exports.CommercePortalConsta
nts.S_REMOTE_SERVLET_URL_PARAM;

String sPortalURI =null;
JSPHelper jsphelperForPrefixCode =new JSPHelper(request );
Object oPortalURI =jsphelperForPrefixCode.getParameter(sPortalURIParam );

if (oPortalURI !=null )
{
   sPortalURI =(String )oPortalURI;
   sPortalURI =sPortalURI.substring(0,sPortalURI.lastIndexOf("#"));
}

boolean fModifyLinks =false;
if (sPortalURI !=null )
   fModifyLinks =true;
%>
```

Note: This code can be maintained in a JSP include file to be included by each portal_jsp JSP.

The parameters obtained in the above piece of code are used for proper link encoding by the portlet JSP and mentioned in the remaining guidelines.

2. All links in the portlet JSP file must adhere to the specific encoding convention. If the links do not follow the specified mechanism, these links will not work properly.

   a. Image links:
   If images in the JSP files are specified by absolute URLs, no changes are need. If images in the JSP file are referenced by relative URLs, the encoding must be in the following form:

   ```
   <img src="<%=sURLImgPrefix%>/relative_image_path.jpg">
   ```

   b. URLs:
   All the URLs in the JSP file must follow specific encoding schema defined for Commerce Enabled portlets. The following is the example of such a modification. If you had the following URL in your regular JSP file:

   ```
   <a href="PrivacyView?langId=<%=languageId%>&
   storeID=<%=storeId%>&catalogId=<%=catalogId%>">PrivacyPolicy</a>
   ```

   In your Commerce Enabled Portlet JSP file the same link will be encoded as follows:

   ```
   <a href="<%=sPortalURI%>?<%=sRemoteServletURIParam%>=
   <%=java.net.URLEncoder. encode(sURLPrefix+"PrivacyView?langId="+
   ```

```
languageId +"&storeId="+storeId +"&catalogId="+
catalogId)%>">PrivacyPolicy</a>
```

As you can see the difference is in the fact that now request has to go to <%=sPortalURI%> URL and your old URL becomes a value for the <%=sRemoteServletURIParam%> parameter.

3. Forms Modification: All the forms in the JSP file must have the same value for the action, given by <%=sPortalURI%>. Also, all forms must have an additional <%=sRemoteServletURIParam%> parameter, containing the older action value of the form. If the user must see default representation of the portlet after the form submission, the following parameter must be added to the form:

```
<input
type="hidden"name="<%=com.ibm.commerce.portal.exports.CommercePortalConstan
ts.S_TERMINAL_URI_PARAM%>"value="<%=com.ibm.commerce.portal.exports.Commerc
ePortalConstants.S_TERMINAL_URI_PARAM_TRUE%>">
```

For example, if your original form looked like:

```
<FORM METHOD="POST" NAME="Logon" action="Logon"><INPUT TYPE="hidden"
NAME="storeId" VALUE="<%=storeId%>"><INPUT TYPE="hidden"
NAME="catalogId"VALUE="<%=catalogId%>"></FORM>
```

Then it should become:

```
<FORM METHOD="POST" NAME="Logon" action="<%=sPortalURI%>"><INPUT
TYPE="hidden" name="<%=sRemoteServletURIParam %>" value="<%=sURLPrefix
+"Logon"%>"><INPUT TYPE="hidden" NAME="storeId" VALUE="<%=storeId%>"><INPUT
TYPE="hidden" NAME="catalogId" VALUE="<%=catalogId%>"></FORM>
```

4.4.5 Developing commerce enabed portlets

To develop portlets you can use the WebSphere Studio Application Developer wizards, subclass the provided classes of the WebSphere Commerce Portal provided with the Commerce Enhancement Pack, or write native WebSphere Portal portlets. In the event you develop native portlets you must ensure that the UserAgent parameter is set properly.

The Commerce Enhancement Pack portal adapters assume that the following string is added to the UserAgent information:

```
(WpsHTTPClient)
```

For example:

```
httpurlconnection.setRequestProperty("User-Agent", sAgentId +
"(WpsHTTPClient)");
```

4.5 Package SAR and publish Web assets

This section describes how to package and publish the CEP stores. In our CEP scenario, we have several CEP stores for different B2B customers.

This section includes the following tasks:

- Create as SAR using ant_create_sar.xml.
- Publish Web assets to VisualAge for Java WebSphere Test Environment using ant_publish_VAJ.xml.
- Publish Web assets to WebSphere Commerce using ant_publish_WCS.xml.

To fully take advantage of the scripts provided for publishing, it is required that the WebSphere Commerce runtime and WebSphere Commerce Studio development environments are installed and configured.

In order to use the scripts provided in WebSphere Studio Application Developer providied in c:\redp3684-code\wsad, we recommend that you create a shared directory as follows:

1. Share the drive of your installed WebSphere Commerce Studio environment node or WebSphere Commerce runtime environment node using Windows file sharing.
2. Map the drive of the WebSphere Commerce Studio environment node or WebSphere Commerce runtime environment node to your client machine and assign the drive letter, for example, to W.

Note: You must update the provided ant scripts for the creation of SAR files and the publishing process. Update the drive information in the configuration files where necessary.

4.5.1 Create as SAR using ant_create_sar.xml

This section describes how the ant_create_sar.xml script found in the b2bCepWeb\ant directory of the workspace, can be used to repackage a SAR file.

Note: You must modify the `cep.config.properties` file for your needs to work properly. It contains a set of directory settings.

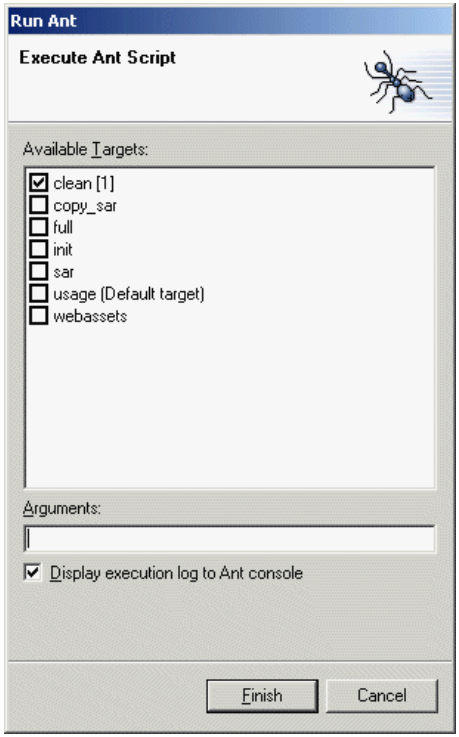


Figure 4-1 ant_create_sar.xml targets

To run the ant_create_sar.xml script within the WebSphere Studio Application Developer do the following:

1. Open the `ant` directory under your `b2bCepWeb` folder.
2. Right-click on the `ant_create_sar.xml` and from the context menu select `Run Ant.`
3. Select one of the following targets to run as seen in Figure 4-1.
   - full

     To generate all assets including the SAR file and copying the SAR file to the source directory.
   - copy_sar

     To copy the SAR file to the source directory.
   - sar

     To generate the SAR file including all necessary assets.
   - init

– clean

To delete all generated assets.

– webassets

To generate all assets for the SAR file.

4. By default, the SAR file is packaged in the c:\redp3684-code\wsad\cep\b2bCepWeb\sar directory.

The SAR file packaged using this script can be deployed to the runtime or development test environment using Store Services as described in Chapter 5, “Deploy a commerce enabled portal store” on page 207.

4.5.2 Publish Web assets to VisualAge for Java WebSphere Test Environment using ant_publish_VAJ.xml

The ant_publish_VAJ.xml script can be used if you want to publish Web assets to the VisualAge for Java WebSphere Test Environment. The publish process updates the following directories:

- Store properties

  `C:\IBM\VAJava\ide\project_resources\IBM WebSphere Test Environment\hosts\default_host\default_app\web\WEB-INF\classes\b2bCep`

- Portal assets

  `C:\IBM\VAJava\ide\project_resources\IBM WebSphere Test Environment\hosts\default_host\default_app\web\b2bCep`

- Web assets

  `C:\IBM\VAJava\ide\project_resources\IBM WebSphere Test Environment\hosts\default_host\default_app\web\b2bCep`

- Tools runtime xml

  `C:\IBM\CommerceServerDev\wc.ear\wcstores.war\xml\b2bCep`

- Tools properties

  `C:\IBM\CommerceServerDev\wc.ear\properties\tools\stores\b2bCep`

- Tools assets

  `C:\IBM\CommerceServerDev\wc.ear\wctools.war\xml\tools\stores\b2bCep`

Note: The `instance name` and `directories` are defined in the `cep.config.properties` file and `cep.vaj.properties` file.

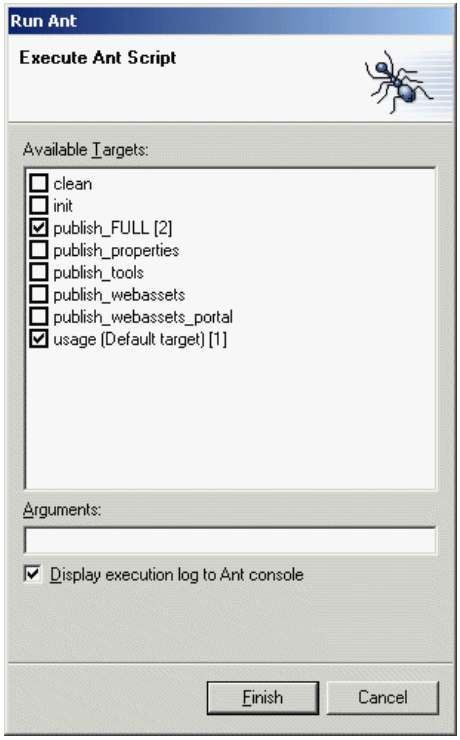


Figure 4-2 ant_publish_VAJ.xml targets

To run the script within the WebSphere Studio Application Developer do the following:

1. Open the `ant` directory under your `b2bCepWeb` folder.
2. Right click on the `ant_publish_VAJ.xml` and from the context menu select `Run Ant.`
3. Select one of the following targets to run:
   - publish_FULL

     To publish all assets to the destination directories for VisualAge for Java.
   - publish_properties

     To publish only the properties to the destination directories for VisualAge for Java.
   - publish_tools

     To publish all tool assets to the destination directories for VisualAge for Java.

– publish_webassets

  To publish all web assets to the destination directories for VisualAge for Java.

– publish_webassets_portal

  To publish all portal web assets to the destination directories for VisualAge for Java.

– init
– clean

4.5.3 Publish Web assets to WebSphere Commerce using ant_publish_WCS.xml

The ant_publish_WCS.xml script can be used if you want to publish web assets to the WebSphere Commerce runtime environment. The publish process updates the following directories:

- Store properties

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\wcstores.war\WEB-INF\classes\b2bCep`

- Portal assets

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\wcstores.war\b2bCep`

- Web assets

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\wcstores.war\b2bCep`

- Tools runtime xml

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\wcstores.war\xml\b2bCep`

- Tools properties

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\properties\tools\stores\b2bCep`

- Tools assets

  `C:\IBM\was\installedApps\WC_Enterprise_App_demo.ear\wctools.war\xml\tools\stores\b2bCep`

Note: The `instance name` and `directories` are defined in the `cep.config.properties` file and `cep.wcs.properties` file.

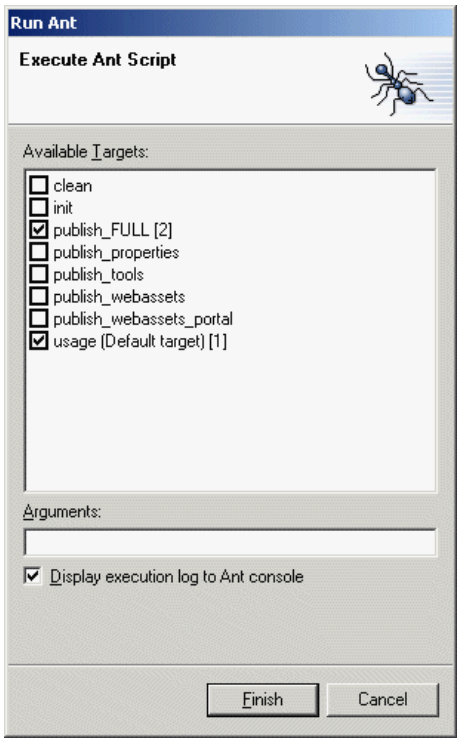


Figure 4-3 ant_publish_WCS.xml targets

To run the script within the WebSphere Studio Application Developer do the following:

1. Open the `ant` directory under your `b2bCepWeb` folder.
2. Right click on the `ant_publish_WCS.xml` and from the context menu select `Run Ant`.
3. Select one of the following targets to run:
   - publish_FULL

     To publish all assets to the destination directories for WebSphere Commerce.
   - publish_properties

     To publish only the properties to the destination directories for WebSphere Commerce.
   - publish_tools

     To publish all tool assets to the destination directories for WebSphere Commerce.

– publish_webassets

  To publish all web assets to the destination directories for WebSphere Commerce.

– publish_webassets_portal

  To publish all portal web assets to the destination directories for WebSphere Commerce.

– init
– clean

5

Deploy a commerce enabled portal store

This chapter includes procedures for deploying the ITSO B2B CEP store working example to the runtime environment and the development test environment.

The chapter is organized as follows:

- Deploy ITSO B2B CEP store to runtime environment

  This section describes how to deploy the ITSO B2B CEP store to the WebSphere Commerce Portal runtime environment. This procedure is useful for verifying the runtime environment and for experimenting with the sample.

- Deploy the ITSO B2B CEP store to the development test environment

  This section describes how to deploy the ITSO B2B CEP store to development test environment for development and debug purposes.

5.1 Deploy ITSO B2B CEP store to runtime environment

After completing the configuration of the runtime environment as described in Chapter 2, “Implement the runtime environment” on page 7, the ITSO B2B CEP store can be deployed.

This section includes the following:

1. ITSO sample code
2. Database backup
3. Server startup
4. Create a new organization for the ITSO B2B CEP store
5. Create a new user and add to CEP organization
6. Assign roles to an organization
7. Assign roles to a user of an organization
8. Prepare the ITSO B2B CEP store SAR for publishing
9. Create a store archive from the ITSO B2B CEP store template
10. Publish the store from Store Services
11. Verify the ITSO B2B CEP store
12. Deploy the commerce enabled portlets
13. Set commerce portlet permissions
14. Verify the ITSO B2B CEP store portal functionality

Disable WebSphere security before publishing

If WebSphere Portal and WebSphere Commerce are installed on the same node, WebSphere security should be disabled before publishing a store. This is not necessary if they are on separate nodes.

1. Ensure the WebSphere Portal application server is stopped before disabling WebSphere security. Do not start the WebSphere Portal application server until WebSphere security has been enabled after publishing.
2. Start the WebSphere Application Server Administration Console.
3. Click **Console -> Security Center**.
4. From the General tab, clear the **Enable Security** checkbox.
5. Click **Apply** and then **OK**.
6. Stop and start the WebSphere Administrative Server (IBM WS AdminServer 4.0) for changes to take effect.

After publishing the store, WebSphere security can be enabled.

5.1.1 ITSO sample code

For details on the ITSO sample code refer to 4.1, “ITSO sample code” on page 184.

5.1.2 Database backup

Backup the WebSphere Commerce instance database and other supporting databases before publishing the ITSO B2B CEP store SAR file.

We recommend a backup of the following databases:

- WebSphere Commerce <instance> database
- WebSphere Application Server repository database
- WebSphere Commerce Payments database

We created a new database backup directory c:\ibm\dbbakprestore.

For details on how to backup a DB2 database refer to, “Backing up a DB2 database” on page 259.

5.1.3 Server startup

Ensure the following are started on each of the nodes:

- WebSphere Portal node:
  - DB2
  - IBM HTTP Server
  - IBM WS AdminServer 4.0
  - WebSphere Portal application server
- Directory Server node:
  - DB2
  - IBM HTTP Server
  - IBM SecureWay Directory V3.2.2
- WebSphere Commerce node:
  - DB2
  - IBM HTTP Server
  - WebSphere Administrative Server (adminserver.bat)
  - WebSphere Commerce Payments application server + IBMPayServer.cmd
  - WebSphere Commerce <instance> application server

5.1.4 Create a new organization for the ITSO B2B CEP store

If you have already published a store for the existing WebSphere Commerce instance, you will need to create a new organization for the ITSO B2B CEP store. If you have not published an a store previously, you can use the predefined *Default Organization* as the store owner.

To create a new organization for the ITSO from the WebSphere Commerce Administration Console do the following:

1. Enter the following URL in a Web browser to start the WebSphere Commerce Administration Console:

   `https://<hostname>:8000/adminconsole`

2. Log in as wcsadmin, select **Site**, and click **OK**.
3. Select **Access Management -> Organizations** from the menu bar.
4. In the Organization window, click **New**.
5. Enter the following in the Details window and then click **Next**:
   - Short name (required): `CEP`
   - Distinguished name: *Leave this field blank*

Note: When attempting to create an organization with the DN supplied, we received a generic system error. For this reason, we left the distinguished name field blank when creating the organization. It will default the DN configured in LDAP.

- Description: `CEP`
- Business category: `CEP`
- Organization type: select **Organization**

6. Enter the following in the Address window and then click **Finish**:
   - Street address (required)
   - City (required)
   - State/Province (required)
   - Zip/Postal code (required)
   - Country/Region (required)
7. When done, you should see the newly created organization listed. Notice, the parent organization to the CEP organization is Root Organization. Double-click on the CEP organization. Notice the Distinguished Name is `o=CEP,dc=ibm,dc=com`. Figure 5-1 on page 212 displays the CEP organization from the Administration Console.

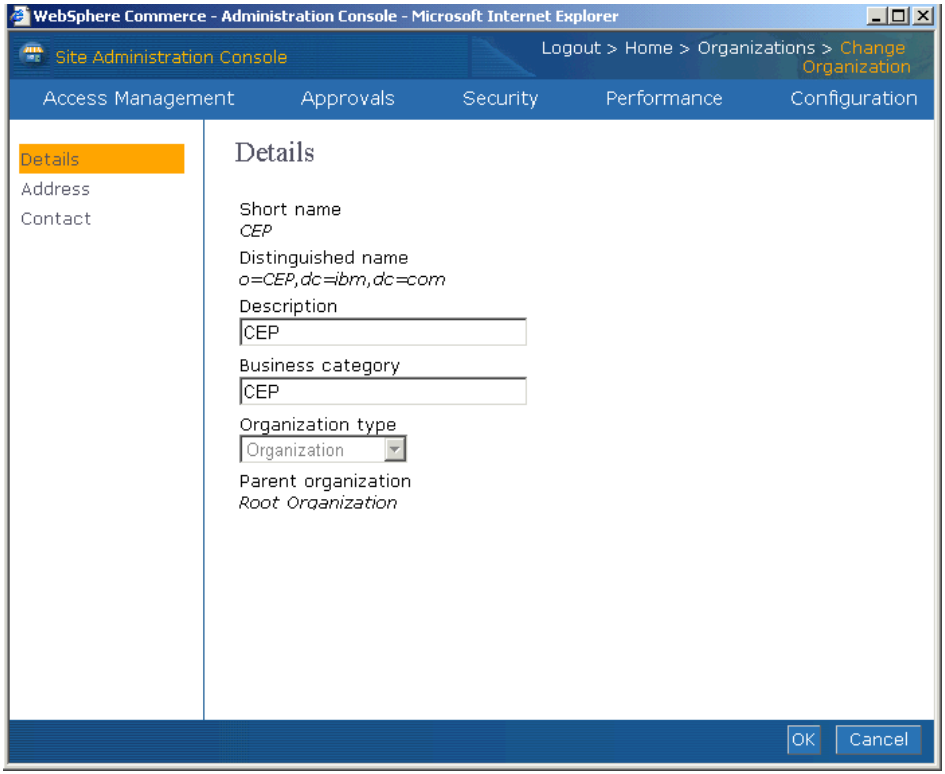


Figure 5-1 CEP organization

8. Logout and close the WebSphere Commerce Administration Console.

5.1.5 Create a new user and add to CEP organization

To create a new user as part of the CEP organization, do the following from the WebSphere Commerce Administration Console:

1. Start the WebSphere Commerce Administration Console:

   `https://<wc_hostname>:8000/adminconsole`

2. Select **Access Management -> Users**.
3. Click **New**.
4. When the Details window appears, we entered the following and then click **Next**:
   - Title: (optional)
   - First Name: (optional)
   - Middle Name: (optional)

- Last Name: `buycep1` (required)
- Logon ID: `buycep1` (required)
- Password: `<password>` (required)
- Password confirmation: `<password>` (required)
- Challenge question: (optional)
- Answer to challenge question: (optional)
- Account policy: select **Shoppers**
- Account Status: select **Enabled**

5. When the Business Profile window appears, we entered the following and then clicked **Next**:
   - Employee ID: (optional)
   - Employee type: (optional)
   - Department number: (optional)
   - Manager's name: (optional)
   - Administrative assistant's name: (optional)
   - Preferred language: we selected **United States English**
   - Parent organization: click **Find** (required)
6. A listing of the organizations will appear. We checked the **CEP** organization, clicked **Select** and then clicked **Next**.
7. When the Address window appears, we entered the following and then clicked **Next**:
   - Street address: `700 Park Office` (required)
   - City: `RTP` (required)
   - State/Province: `NC` (required)
   - Zip/Postal code: `27709` (required)
   - Country/Region: `USA` (required)
8. When the Contact window appears, we did not enter any information as these fields are optional (for test purposes). Click **Finish**.
9. Notice the user we created, `uid=buycep1,o=CEP,dc=ibm,dc=com` is displayed in the Administration Console as seen in Figure 5-2 on page 214.

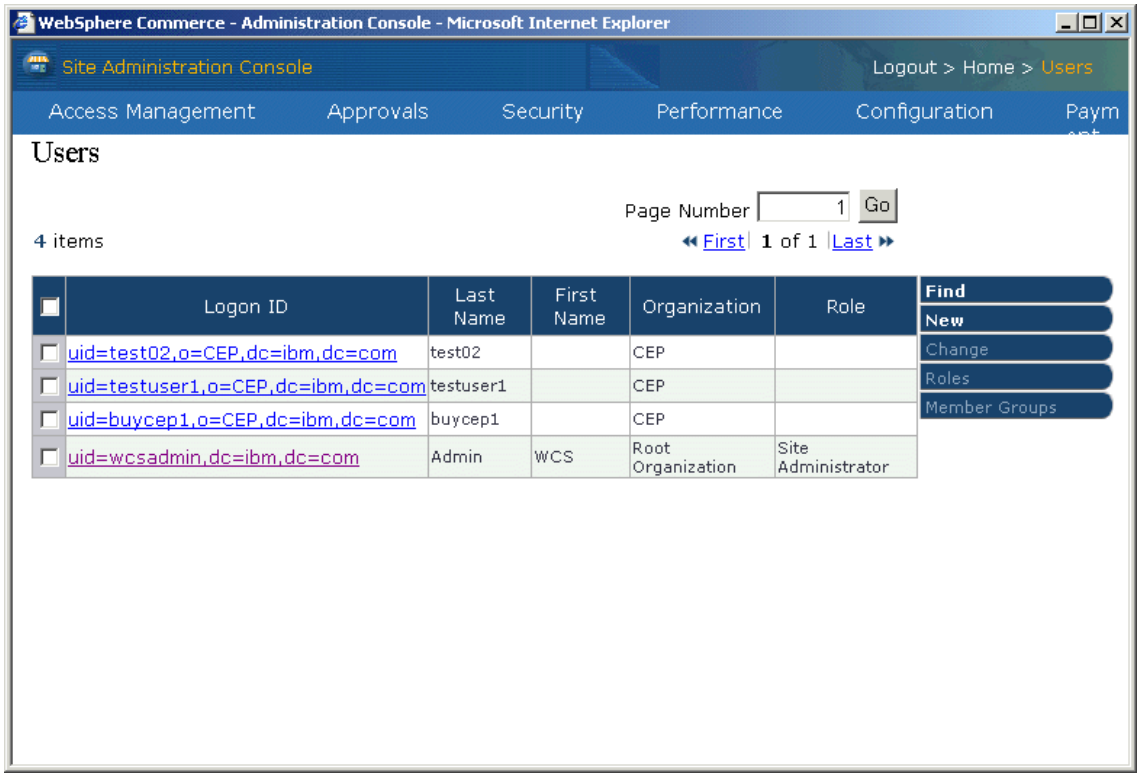


Figure 5-2 WebSphere Commerce Administration Console - testuser1

5.1.6 Assign roles to an organization

In order to assign roles to a user, the organization must first be assigned roles. There are many predefined roles or roles can be created. In our example, we assigned all possible roles to the CEP organization we created.

To assign the WebSphere Commerce roles to the CEP organization, do the following:

1. From the WebSphere Commerce Administration Console, select **Access Management -> Organizations**.
2. Check the **CEP** organization, and then click **Roles**.
3. Select the desired role and click Add. In our example, we clicked **Add All** roles for the CEP organization and then clicked **OK**.

5.1.7 Assign roles to a user of an organization

In our example, we assigned the buyer roles, subset of all roles available, to the buycep1 user of the CEP organization created in 5.1.5, “Create a new user and add to CEP organization” on page 212.

To assign the buycep1 user of the CEP organization buyer roles, do the following:

1. From the WebSphere Commerce Administration Console, select **Access Management -> Users**.
2. Check the **uid=buycep1,o=CEP,dc=ibm,dc=com** user.
3. Click **Roles** from the right-hand controls.
4. When the Roles window appears, we selected the following and clicked **Add** for each:
   - Organization: select **CEP**
   - Role: <role> click **Add**

     Where <role> is all of the following:

     - Procurement Buyer
     - Procurement Buyer Administrator
     - Buyer (buy-side)
     - Buyer (sell-side)
     - Buyer Administrator
     - Buyer Approver

   Note: The roles assigned are for example purposes and may be different for your business needs.

5. When done adding roles, you should see something like Figure 5-3, “Assign roles to buycep1 user” on page 216.

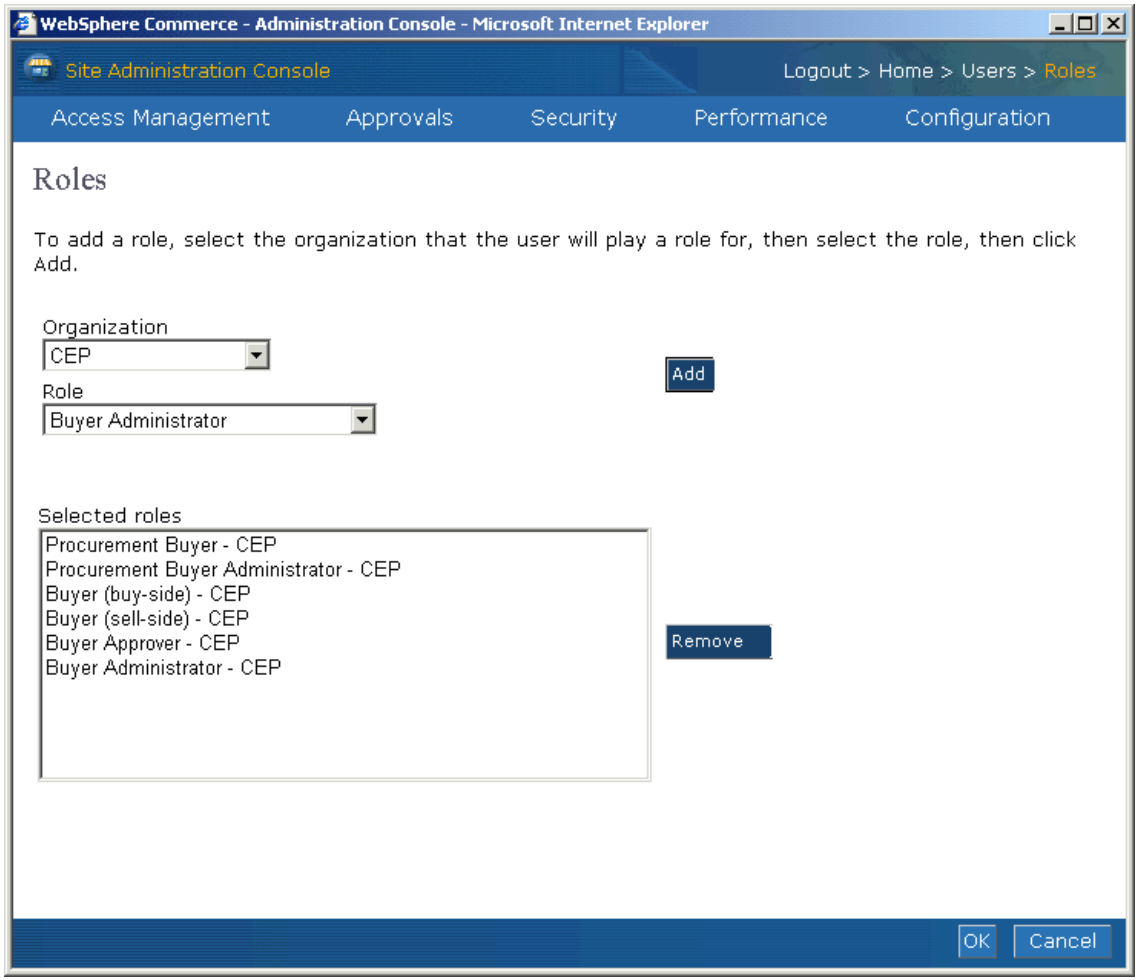


Figure 5-3 Assign roles to buycep1 user

6. After clicking **OK**, you will see the roles assigned to the buycep1 user as displayed in Figure 5-4.

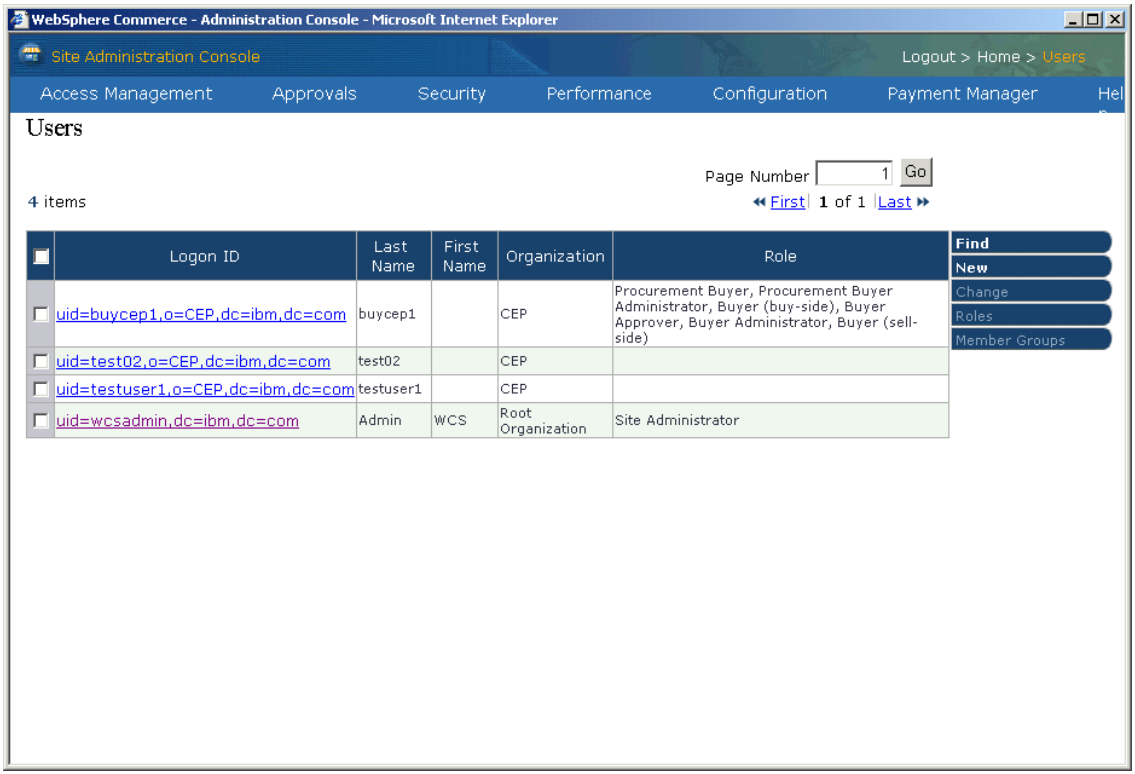


Figure 5-4 Roles displayed for the buycep1 user

5.1.8 Prepare the ITSO B2B CEP store SAR for publishing

Before the ITSO B2B CEP store SAR can be published, the WebSphere Commerce node must be prepared as follows:

1. Create a directory in the `<WC_HOME>\samplestores` directory for the ITSO B2B CEP store (for example, `c:\ibm\wc\samplestores\b2bcep`).
2. Copy the `b2bCep_en_US_es_ES.sar` and `Feature_en_US.html` files found in the c:\redp3684-code\sar directory to the `<WC_HOME>\samplestores\b2bcep` directory.
3. Modify the `<WC_HOME>\xml\tools\devtools\SARRegistry.xml` file with a text editor to include the entries listed in Example 5-1 on page 217.

Example 5-1 Sample SARRegistry.xml update for ITSO B2B CEP store

```
<SampleSAR fileName="b2bCep_en_US_es_ES.sar" relativePath="b2bcep">
       <html locale="en_US" featureFile="b2bcep/Feature_en_US.html"/>
```

```
</SampleSAR>
```

4. After you have saved the SARRegistry.xml, do the following for the changes made to take effect:
   - Close Store Services.
   - Restart the WebSphere Commerce <instance> application server.

   Note: We noticed that the SARRegistry.xml file was locked even after closing Store Services. In order to save the SARRegistry.xml file without getting a locked file error, we had to stop the application server, save the SARRegistry.xml and then start the application server.

5.1.9 Create a store archive from the ITSO B2B CEP store template

To create a store based on the ITSO B2B CEP store template, do the following:

1. Ensure the following are started:
   - IBM DB2 services
   - IBM HTTP Server (Windows service)
   - IBM WS Admin Server 4.0 (Windows service)

     Or, adminserver.bat if you have modified this file to include startup parameters.
   - WebSphere Commerce - <instance_name> (WebSphere application server)
   - WebSphere Payment Manager (WebSphere application server)
   - IBMPayServer (command executed and running)
2. Start Store Services by entering the following URL from a Web browser:

   `https://<hostname>:8000/storeservices`
3. Log on as `wcsadmin`.
4. Select the sample store that you want your new store to be based on. For example, we selected the following for ITSO and then clicked **OK**:
   - Select **b2bCep_en_US_es_ES.sar**
   - Store archive: `b2bcep1`
   - Store directory: `b2bcep1`
   - Store owner: select **CEP**
5. You should see the message `b2bcep1.sar created successfully`. Click **OK**.

   You have now created a new store archive named b2bcep1 based on the b2bCep_en_US_es_ES.sar template store.

5.1.10 Publish the store from Store Services

Now that the store archive b2bcep1.sar has been created, do the following to publish the ITSO B2B CEP store from Store Services:

1. To publish the store, check the store you created (for example, we checked b2bcep1.sar) and then click **Publish**.
2. We accepted the defaults for the remaining options.
3. We monitored the <WC_HOME>\instances\<instance>\logs\wcs.log and used the Windows task manager to track publishing progress. Periodically, click the Refresh option when CPU activity has decline in the Windows Task Manager. When publishing status changes to `Publishing Completed Successfully`, continue to the next step.

   If the publishing fails refer to, “Troubleshooting a store publishing failure” on page 257.

5.1.11 Verify the ITSO B2B CEP store

After publishing the store, we recommend that you verify the functionality of the standard WebSphere Commerce JSPs by directly accessing the WebSphere Commerce node, and the functionality of the commerce enabled portal store.

1. To perform a basic verification test of the store after publishing is finished, click **Publish Summary -> Launch Store**. Add the store to your favorites or bookmarks list.

   Alternatively, enter the store URL such as:

   `http://<wc_hostname>/webapp/wcs/stores/servlet/b2bcep1/index.jsp`
2. Register a new user.
3. Logon with the new user.
4. Browse the store catalog.
5. Select and item to purchase and add to shopping cart.
6. Complete an order by selecting a valid payment type. For example, we selected Visa and entered the 16 zeroes (0) as the card number for test purposes (or 4111111111111111, 15 zeros).
7. Verify the order from WebSphere Commerce Payments.
   a. Logon to the WebSphere Commerce Payments:

      `http://<wcpay_hostname>/webapp/PaymentManager`

   b. Click **Approve**. You should see the order that was just placed. Select the order and click **Approve**.

c. You should see a message indicating the order was approved and that there are no more pending order to approve.

5.1.12 Deploy the commerce enabled portlets

This section describes how to deploy the commerce enabled portlets supplied with the B2B Direct sample of the IBM Commerce Enhancement Pack - October 2002 Edition as a base for the ITSO B2B CEP store sample.

Note: For more information, refer to *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*.

To deploy and configure the commerce enabled portal runtime for the B2B Direct sample, do the following on WebSphere Portal node:

1. Ensure you have published one of the following commerce enabled portal store SAR files.

   Note: This step will have already been completed if you followed the procedure documented 5.1, “Deploy ITSO B2B CEP store to runtime environment” on page 208).

   – ITSO B2B CEP store SAR, by doing the following:
   - 5.1.2, “Database backup” on page 209
   - 5.1.4, “Create a new organization for the ITSO B2B CEP store” on page 210
   - 5.1.8, “Prepare the ITSO B2B CEP store SAR for publishing” on page 217
   - 5.1.9, “Create a store archive from the ITSO B2B CEP store template” on page 218
   - 5.1.10, “Publish the store from Store Services” on page 219
   - 5.1.14, “Verify the ITSO B2B CEP store portal functionality” on page 227

   Or,

   – B2B Direct sample SAR (refer to *Commerce Enabled Portal Integration Guide, IBM Commerce Enhancement Pack October 2002 Edition*)

2. Unpack the `WebSphere CommerceEnabledPortal.zip` included with the IBM Commerce Enhancement Pack - October 2002 Edition (for example, C:\temp\CEP). We will refer to the unpack directory as `<CEP_HOME>`.

3. Change to the <CEP_HOME>\B2BDirectPortal\CommerceRefApp\bin directory.
4. Rename the WebsphereCommerceEnabledPortalB2BDirect.war file to WebsphereCommerceEnabledPortalB2BDirect_original.war.

   This has the effect of creating a backup, since we will modify files and repackage the war file as WebsphereCommerceEnabledPortalB2BDirect.war.
5. Set PATH to include Java:

   ```
   set PATH=%PATH%;c:\ibm\was\java\bin
   ```
6. Unpack the WebsphereCommerceEnabledPortalB2BDirect_original.war file:

   ```
   jar -xvf WebsphereCommerceEnabledPortalB2BDirect_original.war
   ```
7. Edit the WEB-INF/portlet.xml file:
   a. Search for all hostname.domain entries and replace them with your WebSphere Commerce node hostname and domain (for example, wcserv1.itso.ral.ibm.com)
   b. In our scenario, our environment is configured to use SSL. Change all portlet URLs in the portlet.xml accessing port 8000 from http:// to https:// in the portlet.xml file. For example, in our environment we replaced all occurrences as follows:

      ```
      http://wcserv1.itso.ral.ibm.com:8000/
      https://wcserv1.itso.ral.ibm.com:8000/
      ```
   c. Update the values of for the following parameters in the portlet.xml file to match your published store.
      - storeId
      - catalogId
      - langId
      - categoryId
      - productId
      - parent_category_rn

      Table 5-1 on page 222 includes the values entered for ITSO B2B CEP store. The values for you store may be different depending on the if you have published stores prior to the ITSO B2B CEP store store. Also, as you add/delete products and categories, the reference ID numbers will change.

Note: When updating the information in the portlet.xml file, we have the following recommendations for updating these values:

- We suggest that you update the parameters in the order listed.
- The `storeId`, `catalogId` and `langId` are the easiest to find and update.
- Ensure you update the value only.
- When updating the categoryId and productId, please note that these values may change as catalog data is resolved and loaded. You may have to update the portlet.xml file in the future as the resolved ID numbers change. Find the categoryId values for the following used in the B2B commerce enabled portal sample. For example:
  - Woodworking Saws
  - Sanders
  - Lathes
  - Drills
  - Grinders
  - Screwdrivers
- For test/development purposes, we chose a couple categoryId's and updated one productId value for them and the parent_category_rn.
- The highest level category is the catalogId (parent_category_rn for high level categories).

There are a couple of methods than can be used to obtain the values of the parameters (database columns).

- Record values from WebSphere Commerce URL when navigating the site categories and products.
- Perform DB2 sql queries on the WebSphere Commerce instance database.

  For example:

```
db2 connect to <wc_database>
db2 select * from CATGROUP
```

Table 5-1 ITSO B2B CEP store portlet.xml values

Parameter/element	Value
hostname.domain	wcserv1.itso.ral.ibm.com
langId	-1
storeId	10001

Parameter/element	Value
catalogId	10001
Category portlet updates	
Woodworking saws categoryId * parent_category_rn **Note**: The parent category for the highest level categories is the catologId.	10005 10001
Sanders categoryId * parent_category_rn	10006 10001
Lathes categoryId * parent_category_rn	10007 10001
Drills categoryId * parent_category_rn	10008 10001
Grinders categoryId * parent_category_rn	10009 10001
Screwdrivers categoryId * parent_category_rn	10010 10001
Airtools categoryId * parent_category_rn	10011 10001
Cordless saws categoryId * parent_category_rn	10012 10001
Handsaws categoryId * parent_category_rn	10015 10001
Drill bits categoryId * parent_category_rn	10013 10001
Chargers categoryId * parent_category_rn	10014 10001
Product portlet updates	
Circular saw productId * Woodworking parent_category_rn	10001 10005 (Woodworking saw categoryId)
Electric Sander productId Sanders parent_category_rn	10037 10006 (Sander categoryId)

Parameter/element	Value
Second Operation Lathe productId * Lathes parent_category_rn	10073 10007 (Lathe categoryId)
Hammer Drill productId * Drills parent_category_rn	10109 10008 (Drill categoryId)
Small Angle Grinder productId * Grinders parent_category_rn	10154 10009 (Grinder categoryId)
Power Screwdrivers productId * Drywall Screwdrivers parent_category_rn	10190 10010 (Screwdriver categoryId)

8. Now that we have completed the modifications to the `portlet.xml` file, the `WebsphereCommerceEnabledPortalB2BDirect.war` file needs to be repackaged.
   a. Change to the following directory:

      ```
      <CEP_HOME>\B2BDirectPortal\CommerceRefApp\bin
      ```

   b. Set PATH to include Java:

      ```
      set PATH=%PATH%;c:\ibm\was\java\bin
      ```

   c. Repackage the war file to include the updates as follows:

      ```
      jar -cvf WebsphereCommerceEnabledPortalB2BDirect.war WEB-INF
      ```

9. Deploy the updated B2B Direct sample portlets.
   a. Change to the following directory:

      ```
      <CEP_HOME>\B2BDirectPortal\CommerceRefApp\scripts
      ```

   b. Run the following command to deploy the B2B direct sample portlets.

      ```
      installb2bdirectcommerceportal.bat
      ```

      You will be prompted with the following (for our example, we entered the following values):

      - Enter WPS Root: `c:\ibm\PortalServer`
      - Enter WPS Admin UID: `wpsadmin`
      - Enter WPS Admin Password: `wpsadmin`
      - Enter WPS Access URL: `http://wcportal1.itso.ral.ibm.com/wps`

Note: After the `WebsphereCommerceEnabledPortalB2BDirect.war` is deployed, the deployed version of the portlet.xml file can be found in the `<WAS_HOME>\InstalledApps\WebSphereCommerceEnabledPortalB2BDirect<stamp>\WebSphereCommerceEnabledPortalB2BDirect.war\WEB-INF` directory. For test purposes, you may edit the portlet.xml file from this location. For production deployment and proper testing, we recommend that you edit portlet.xml, repackage the war file and then redeploy the war.

10. Install B2B Direct sample store personalization workspace:
    a. Change to the following directory:

       `<CEP_HOME>\B2BDirectPortal\PersonalizationUserHomePageBaseFolder\scripts`

    b. Run the following command to deploy the personalization workspace:

       `createB2BDirectPZNWorkspace.bat`

       You will be prompted with the following (for example, we entered the following values):

       - Enter Path to WebSphere Application Server: `c:\ibm\was`
       - Enter Path to Commerce Portal Server: `c:\ibm\cps`

5.1.13 Set commerce portlet permissions

After the WebSphere Portal installation and configuration, set the commerce portlet permissions within the WebSphere Portal Administration Console. Login to the portal as `wpsadmin` and add at least View permission for Commerce Enhancement Pack portlet application, portlets, places and pages for portal users/groups.

1. From a Web browser, enter the following URL:

   `http://<portalserver_fullyqualified_hostname>/wps/portal`

2. In the upper right, click on the **Key** symbol to login. You will be prompted for the following, and then click **Log in**.
   – User ID: `wpsadmin`
   – Password: `wpsadmin`
3. From the pull-down in the upper-left, select **Portal Administration**.
4. From the menu bar, click **Security**.
5. Set permission for portlets.

Note: For test purposes, we set permissions for all authenticated users to have access. In a production environment, you will need to set permissions for specific groups or users.

a. Select the **Special Groups** radio button. From pull-down selection **All Authenticated Users**.

b. Select the objects for permissions. In this case, we selected **portlets**.

c. Select the **Show All** radio button.

d. Click **Go** to display the list of portlets.

e. Check **view permission** for the following commerce enabled portlets; when done click **Save**:

- Account List
- Advanced Search
- Catalog
- Contract List
- Current Orders
- Drills
- Drills SKUs
- Grinders
- Grinder SKUs
- Invoice List
- Lathes
- Lathes SKUs
- My Account
- MyAlert
- Order Approval
- Order History
- Order List
- Product Catalog
- Product Search
- RFQ Approval
- RFQ List and Launch RFQ tool
- RFQ Response List
- Requisition List
- Sanders
- Sanders SKUs
- Screwdrivers
- Screwdrivers SKUs
- SendAlertMessage
- Woodworking Saws
- Woodworking Saws SKUs

- myCatalog
- quickOrder

6. Set permissions for the portlet application.
   a. From the Select the objects for the permissions pull-down, select **portlet applications**.
   b. Click **Go**.
   c. From the list of portlet applications, check view permission for the **Commerce Portal Application** and then click **Save**:
7. Set permissions for pages and places.

> **Note:** The WebSphere Portal permissions only need to be set manually if you are not using personalization. The replacement login command provided with CEP will override all portal access control lists according to personalization rules.

   a. From the Select the objects for the permissions pull-down, select **pages**.
   b. Click **Go**.
   c. From the list of following *places* and corresponding *pages*, check view or manage permission (as appropriate for you needs) and then click **Save**:
      - Buyer Organization
      - MyStore
      - Seller Organization

> **Attention:** We found that the WebSphere Portal set the pages/places to manage permission even when we selected view. We believe this is a problem with WebSphere Portal V4.1.4.

8. Logoff the WebSphere Portal admin user (wpsadmin).

5.1.14 Verify the ITSO B2B CEP store portal functionality

Now that the runtime is configured and the ITSO B2B CEP store sample has been deployed, we need to verify the functionality of the B2B commerce enabled portal application. If SSO is working a new token cookie is received by the browser after a successful connection. After logging on to the WebSphere Portal, we will access a WebSphere Commerce page and should not be prompted to logon.

To verify the B2B commerce enabled portal application, do the following:

1. Open a Web browser and type the URL:

   `http://<Portal_fully_qualified_hostname>/wps/portal`

   Note: Ensure you have entered the fully qualified hostname.

2. Click the Key symbol in the upper-right of the page. A login dialog is displayed. Enter the following and then click **Submit Login**.

   – Account: `wcsadmin`
   – Password: `<wcsadmin_password>`

   Note: WebSphere Commerce admin user wcsadmin.

   You should see the WebSphere Portal welcome page.

3. From the pull-down in the upper left, select **MyStore**. You should see a window similar to:

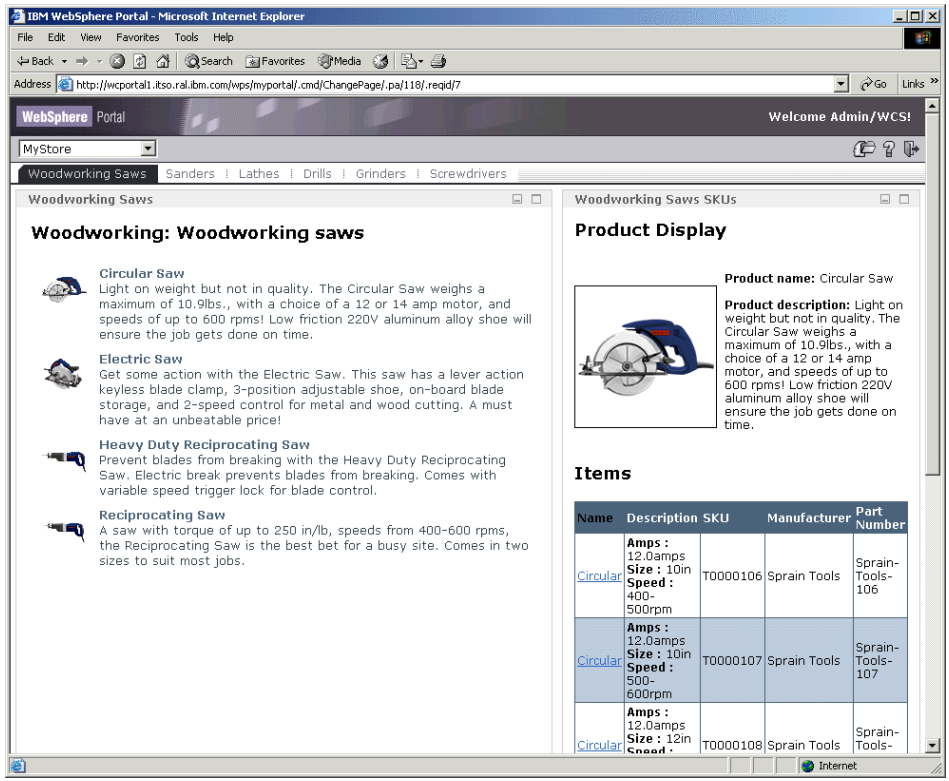


Figure 5-5 ITSO B2B CEP store MyStore home page

4. Experiment with the other options listed under the pull-down.

Congratulations! You have completed the ITSO B2B CEP store deployment to the WebSphere Commerce Portal runtime environment.

5.2 Deploy the ITSO B2B CEP store to the development test environment

After completing the configuration of development environment, as described in Chapter 3, “Implement the development environment” on page 93, the ITSO B2B CEP store can be deployed to the development environment.

The high level steps to deploy the ITSO B2B CEP store are as follows:

1. ITSO sample code
2. Database backup
3. Create organization
4. Prepare the ITSO B2B CEP store SAR for publishing
5. Create the CEP store from the ITSO B2B CEP store template
6. Publish the CEP store
7. Post-publish CEP store configuration
8. Deploy the commerce enabled portlets
9. Verify the CEP store

5.2.1 ITSO sample code

For details on the ITSO sample code refer to 4.1, “ITSO sample code” on page 184.

5.2.2 Database backup

Before you deploy the ITSO B2B CEP store in the development environment, we recommend that you backup the WebSphere Commerce instance database.

Refer to “Backing up a DB2 database” on page 259 for details.

5.2.3 Create organization

Create a new organization for a fictitious working example CEP store from the WebSphere Commerce Administration Console. Refer to the following for details:

- 5.1.4, “Create a new organization for the ITSO B2B CEP store” on page 210
- 5.1.5, “Create a new user and add to CEP organization” on page 212
- 5.1.6, “Assign roles to an organization” on page 214
- 5.1.7, “Assign roles to a user of an organization” on page 215

Enter the following URL in a Web browser to start the WebSphere Commerce Administration Console within the VisualAge for Java WebSphere Test Environment:

```
http://<hostname>:8080/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=adminconsole.AdminConsoleLogon
```

Where <hostname> is the hostname of the VisualAge for Java WebSphere Test Environment node.

5.2.4 Prepare the ITSO B2B CEP store SAR for publishing

Prior to publishing the ITSO B2B CEP store we need to create a new directory for the ITSO B2B CEP store template sar file, configure the SARRegistry.xml for the template to be visible from Store Services, and create a new store for the CEP organization.

1. Ensure the VisualAge for Java WebSphere Test Environment for WebSphere Commerce is started. If not started, start the PNS, EJB server and Servlet Engine.

   For details refer to “Configure and start the EJB, PNS and Servlet Engine” on page 160.

2. Create a directory for the ITSO B2B CEP store template on the Development Tools Node in the <WC_HOME>\samplestores directory. For example:

   ```
   <WC_HOME>\samplestores\cep
   ```

3. Copy the ITSO provided B2B CEP sample store template SAR file, `b2bCep_en_US_es_ES.sar` and `Features_en_US.html` from the following directory:

   ```
   c:\redp3684-code\sar\
   ```

   To the following directory on the Development Tools Node:

   ```
   <WC_HOME>\samplestores\cep\
   ```

4. Update the SARRegistry.xml to add the ITSO B2B CEP store template.

Edit the <WC_HOME>\xml\tools\devtools\SARRegistry.xml to include the ITSO B2B CEP store template SAR elements as seen in Example 5-2.

Example 5-2 Sample B2B CEP SAR elements for SARRegistry.xml

```
<SampleSAR fileName="b2bCep_en_US_es_ES.sar" relativePath="cep">
       <html locale="en_US" featureFile="cep/Feature_en_US.html" />
</SampleSAR>
```

Note: In the event that Store Services was started prior to updating the SARRegistry.xml, restart Store Services for the entry to become available. It is not necessary to stop/start the application server.

5.2.5 Create the CEP store from the ITSO B2B CEP store template

To create a store based on the ITSO provided B2B CEP store template from within Store Services, do the following:

1. Make sure the following are started within the VisualAge for Java WebSphere Test Environment:
   - IBM DB2 Windows services
   - VisualAge for Java
   - WebSphere Test Environment
   - Persistent Name Server
   - EJB Server
   - Servlet Engine
2. Start Store Services by entering the following URL from a Web browser:

   http://<hostname>:8080/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=devtools.Logon

   Where <hostname> is the hostname of the VisualAge for Java WebSphere Test Environment node.
3. Log on using the user ID wcsadmin and <password>.

   Tip: If your account becomes disabled refer, “Resetting a disabled account” on page 260.

4. From Store Services, select the sample store that you want your new store to be based on. For example, we selected the following for the CEP and then clicked **OK**:
   - Select **b2bCep_en_US_es_ES.sar**
   - Store archive: b2bCep1
   - Store directory: b2bCep1

– Store owner: select **CEP**

5. You should see the message `b2bCep1.sar created successfully`. Click **OK**.

You have now created a new store archive named b2bCep1 based on the b2bCep_en_US_es_ES.sar template store.

5.2.6 Publish the CEP store

Now that the store archive for the b2bCep1 has been created, do the following to publish the ITSO B2B CEP store from Store Services:

1. To publish the store, check the store you created (for example, we checked b2bCep1.sar) and then click **Publish**.
2. We accepted the defaults for the remaining options.
3. In the event of a publishing failure, do the following:

> **Note:** On more than one occasion, the status did not change from *publishing* to *published*. We have found using the Windows task manager to monitor CPU utilization useful.
>
> First, try closing Store Services after waiting and verifying from the task manager that CPU activity is low. Restart Store Services and check publishing status. Sometimes it really is completed, while at other times you will need to stop the application server and republish.
>
> If the publishing fails refer to, “Troubleshooting a store publishing failure” on page 257.

4. To perform a basic verification test of the store after publishing by entering the following URL:

   `http://<hostname>:8080/webapp/wcs/stores/servlet/Logoff?storeId=10051&langId=-1`

   – Where <hostname> is the hostname of the node where VisualAge for Java WebSphere Test Environment is running.
   – Enter the values of the storeId and langId for your store.

   Alternatively, from the Publish Summary page of Store Services, click Launch Store.

5.2.7 Post-publish CEP store configuration

After you have published the store using Store Services, you must manually copy the following files and directories:

1. Copy the files and sub directories from:

   ```
   <VAJ_HOME>/ide/project_resources/IBM WebSphere Test
   Environment/hosts/default_host/default_app/web/b2bCep1/CSA/wcstores/portal_
   jsp/
   ```

   To the following location:

   ```
   <WC_HOME>/wc.ear/wcstores.war/
   ```

2. Copy the files and sub directories from:

   ```
   <VAJ_HOME>/ide/project_resources/IBM WebSphere Test
   Environment/hosts/default_host/default_app/web/b2bCep1/CSA/wcstools/portal_
   jsp
   ```

   To the following location:

   ```
   <WC_HOME>/wc.ear/wcstools.war/
   ```

3. In a text editor, open the following two files:

   ```
   <VAJ_HOME>/ide/project_resources/IBM WebSphere Test
   Environment/hosts/default_host/default_app/web/<store>/CSA/wctools/Append_C
   ontractRB_en_US.properties

   <WC_HOME>/wc.ear/properties/com/ibm/commerce/tools/contract/properties/Cont
   ractRB_en_US.properties
   ```

   Add the contents of the `Append_ContractRB_en_US.properties` file to the end of the `ContractRB_en_US.properties` file. Save the file.

4. Save the `ContractRB_en_US.properties` file.

5. In a text editor, open the following two files:

   ```
   <VAJ_HOME>/ide/project_resources/IBM WebSphere Test
   Environment/hosts/default_host/default_app/store_name/CSA/wcstools/Append_O
   rderLabels_en_US.properties

   <WC_HOME>/wc.ear/properties/com/ibm/commerce/tools/order/properties/OrderLa
   bels_en_US.properties
   ```

   Add the contents of the `Append_OrderLabels_en_US.properties` to the end of the `OrderLabels_en_US.properties` file. Save the file.

6. Set the Portal adapter within the WebSphere Commerce instance database.

   From a DB2 command window, connect to your WebSphere Commerce instance database and run:

   ```
   insert into cntrdisply (devicefmt_id,cntrdisply_id,displaypagename) values
   (-4,1,'tools/contract/ContractSummary.jsp')
   ```

Note: The `-4` value above is the default value. Refer to 2.2.13, “Enable WebSphere Commerce portal adapter” on page 35 for more information on determining the proper values.

Additional information can be found in the redbook *WebSphere Commerce Portal V5.4 Solutions*, SG24-6890 (in the Create commerce enabled portals for mobile clients chapter).

5.2.8 Deploy the commerce enabled portlets

To configure the commerce enabled portal runtime for the B2B Direct sample, do the following:

1. Modify the <instance>.xml file
   a. Set the appropriate Http Adapter device Format Id for Portal. For details see 2.2.13, “Enable WebSphere Commerce portal adapter” on page 35.
   b. Set the appropriate Http Adapter device Format Id for WAP Portal. For details refer to the redbook *WebSphere Commerce Portal V5.4 Solutions*, SG24-6890.
   c. Set cache daemon to false (see Example 5-3).

Example 5-3 No cache setup

```
<component compClassName="com.ibm.commerce.cache.daemon.CacheDaemonComponent"
               enable="false"
               name="CacheDaemon" />
```

2. Create the `wpsdebug` user using the WebSphere Commerce Accelerator.

   For details, refer to the *WebSphere Commerce V5.4 online documentation*. Search on *create a user*. This procedures includes selecting the organization and user role.

   Note: The wpsdebug user is required for the ITSO provided debug capability between the portal and commerce.

   From now on the rest of the steps will be done on WebSphere Portal Server (Portal Test Node) machine.

3. Change to the following directory:

   `C:\temp\CEP\B2BDirectPortal\CommerceRefApp\bin`

4. Rename the WebsphereCommerceEnabledPortalB2BDirect.war file to WebsphereCommerceEnabledPortalB2BDirect_original.war.

This has the effect of creating a backup, since we will modify files and repackage as WebsphereCommerceEnabledPortalB2BDirect.war.

5. Set PATH to include Java:

   ```
   set PATH=%PATH%;c:\ibm\was\java\bin
   ```

6. Unpack the WebsphereCommerceEnabledPortalB2BDirect_original.war file:

   ```
   jar -xvf WebsphereCommerceEnabledPortalB2BDirect_original.war
   ```

7. Edit the WEB-INF/portlet.xml file. Refer to the 5.1.12, “Deploy the commerce enabled portlets” on page 220 for more information.

   a. Replace all `hostname.domain` entries with your WebSphere Commerce node hostname and domain. For example, in ITSO:

      ```
      <hostname>:8080
      ```

   b. Update the values of for the following parameters in the portlet.xml file to match your published store:

      - storeId
      - catalogId
      - langId
      - categoryId
      - productId
      - parent_category_rn

Note: When updating the information in the portlet.xml file, we have the following recommendations, although we are not aware of a painless method of updating these values.

- We suggest that you update the parameters in the order listed.
- The storeId, catalogId and langId are the easiest to find and update.
- Ensure you update the value only.
- When updating the categoryId and productId, please note that these values may change as catalog data is resolved and loaded. You may have to update the portlet.xml file in the future as the resolved ID numbers change. Find the categoryId values for the following used in the B2B commerce enabled portal sample:
  - Woodworking Saws
  - Sanders
  - Lathes
  - Drills
  - Grinders
  - Screwdrivers
- For development purposes, we chose a couple categories and updated one productId value for them and the parent_category_rn.
- The highest level category is the catalogId (parent_category_rn for high level categories).

There are a couple of methods that can be used to obtain the values of the parameters (database columns).

- Record values from WebSphere Commerce URL when navigating the site categories and products.
- Perform DB2 sql queries on the WebSphere Commerce instance database.

c. In development environment, we need to turn off SSOEnabled to access commerce from portlet. Search for <param-name>SSOEnabled</param-name> in the portlet.xml file and set **false** to param value. For example, in our environment we replaced all occurrences as seen in Example 5-4.

Example 5-4 Update SSOEnabled attributes in portlet.xml

```
<config-param>
   <param-name>SSOEnabled</param-name>
  <param-value>false</param-value>
```

```
</config-param>
```

d. In development environment, replace all 8000 port number with 8080 in the portlet.xml file. For example, in our environment we replaced all occurrences as follows:

```
hostname.domain:8000
<hostname>:8080
```

8. Delete the `commerceportal.jar` file from `C:\temp\cep\B2BDirectPortal\CommerceRefApp\bin\WEB-INF\lib\` as we are going to replace with ITSO modified version, so that without SSO, still can be access in Commerce from Portal.
9. Copy the `WebsphereCommerceBasePortlet.jar` file into `C:\temp\cep\B2BDirectPortal\CommerceRefApp\bin\WEB-INF\lib\` and rename to `commerceportal.jar`.
10. Configure the WebSphere Commerce logon URL and password.
    a. Create WCSLogon.properties files. For details see in Appendix B, “Debug logon for commerce portlets without single sign-on” on page 245.
    b. Create a dummy.html file on the Development Tools Node.
11. Now that we have completed the modifications to the portlet.xml file, the WebsphereCommerceEnabledPortalB2BDirect.war file needs to be repackaged.
    a. Change to the following directory:

    ```
    C:\temp\CEP\B2BDirectPortal\CommerceRefApp\bin
    ```

    b. Set PATH to include Java:

    ```
    set PATH=%PATH%;c:\ibm\was\java\bin
    ```

    c. Repackage the war file to include the updates as follows:

    ```
    jar -cvf WebsphereCommerceEnabledPortalB2BDirect.war WEB-INF
    ```

12. Deploy the updated B2B sample portlets.
    a. Change to the following directory:

    ```
    c:\temp\CEP\B2BDirectPortal\CommerceRefApp\scripts
    ```

    b. Run the following command to deploy the B2B direct sample portlets.

    ```
    installb2bdirectcommerceportal.bat
    ```

    c. You will be prompted with the following questions (for our example, we entered the following values):
       - Enter WPS Root: `c:\ibm\PortalServer`
       - Enter WPS Admin UID: `wpsadmin`
       - Enter WPS Admin Password: `wpsadmin`

- Enter WPS Access URL: `http://portaldev.itso.ral.ibm.com/wps`

13. Make sure portal server is running. If not open a command window and change to <WAS_HOME>\bin then type the following command:

    ```
    startServer -configFile ..\config\WebSpherePortal-cfg.xml
    ```

14. Install B2B Direct sample store personalization workspace:

    a. Change to the following directory:

    ```
    c:\temp\CEP\B2BDirectPortal\PersonalizationUserHomePageBaseFolder\scripts
    ```

    b. Run the following command to deploy the personalization workspace:

    createB2BDirectPZNWorkspace.bat

    c. You will be prompted with the following questions (for our example, we entered the following values):

    - Enter Path to WebSphere Application Server: `c:\ibm\was`
    - Enter Path to Commerce Portal Server: `c:\ibm\cps`

15. Set the commerce portlet permissions. For details see set permissions.

> **Note:** In a Web browser, enter the following URL:
>
> `http://<portalserver_fullyqualified_hostname>:9080/wps/portal`
>
> instead of `http://<portalserver_fullyqualified_hostname>/wps/portal`.

5.2.9 Verify the CEP store

Now that the B2B commerce enabled portal sample has been deployed, we need to verify the functionality of the B2B commerce enabled portal application. Once logged on to the WebSphere Portal, we will access a WebSphere Commerce page and should not have to logon.

To verify the B2B commerce enabled portal application, do the following:

1. Open a Web browser and type the URL:

   ```
   http://<Portal_fully_qualified_hostname>:9080/wps/portal
   ```

   > **Note:** Ensure you have entered the fully qualified hostname.

2. A login dialog is displayed. Enter the following and then click **Submit Login**.
   - Account: `wpsdebug`
   - Password: `<wpsdebug_password>`

Note: In this case we are using the WebSphere Commerce and Portal Debug user wpsdebug.

You should see the WebSphere Portal welcome page.

3. From the pull-down menu in the upper left, select **MyStore**.
4. Enter the URL for your WebSphere Commerce store. For example, tooltech:

   `http://<wc_hostname>:8080/webapp/wcs/stores/servlet/tooltech/index.jsp`
5. Click **Account** located in the upper part of the page.

Note: Depending on the store model being used, the links referred to in this example may vary.

The Account page should be shown. If the user was not registered in WebSphere Commerce, a different page will appear showing a form to enter a user ID and password.

A

WebSphere Commerce and WebSphere Portal sample LDIF files

This appendix includes sample LDIF files for WebSphere Portal and WebSphere Commerce used to import data into the IBM SecureWay Directory database.

The appendix includes the following sections:

- WebSphere Portal LDIF sample
- WebSphere Commerce LDIF sample

WebSphere Portal LDIF sample

Example A-1 contains a sample WebSphere Portal LDIF file used to import into the IBM SecureWay Directory database for the ITSO working example.

Note: The LDIF sample file wp-itso.ldif,listed in Example A-1 can be found in the `c:\redp3684-code\ldif\wp-itso.ldif`.

This file is based on the WebSphere Portal LDIF template included with WebSphere Portal V4.1.4.

Example: A-1 WebSphere Portal LDIF sample (wp-itso.ldif)

```
version: 1

# NOTE: you must edit this file before importing it and replace all
# occurrences of the default suffix "dc=ibm,dc=com" with the suffix
# that your LDAP server is configured for.

dn: dc=ibm,dc=com
objectclass: domain
objectclass: top
# Add lines according to this scheme that correspond to your suffix
dc: ibm,dc=com
dc: ibm

dn: cn=users,dc=ibm,dc=com
objectclass: container
objectclass: top
cn: users

dn: cn=groups,dc=ibm,dc=com
objectclass: top
objectclass: container
cn: groups

dn: uid=wpsadmin,cn=users,dc=ibm,dc=com
objectclass: organizationalPerson
objectclass: person
objectclass: top
objectclass: inetOrgPerson
uid: wpsadmin
userpassword: {iMASK}>16LcsthnrncZOVioPS+CLJI+WYQiLJ7QNwP1ymxBnPPfdak794823fyV
 9dHh8wk76EcRs9NSrNN9PIKGpqNSs/gnaMbPvTkAp9BJ9uqudufBMcUdpjGBdjpcpe/Ai8dOe3TM
 anELn9qKYXdyzan/rE/ksPyMvQ1Dv9<
sn: admin
givenName: wps
```

```
cn: wps admin


dn: uid=wpsbind,cn=users,dc=ibm,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
uid: wpsbind
userpassword: {iMASK}>1As64OBgqGSOYRnxNxa/VZbxYOH29yF9zM+ZqI4C53TGRvCko5DnYEHO
 8PC7jFc5i1OOnV1Fm54FE2Ftlc/1n3z4tUfNGYrklliuwksTTeU/xZMOOYfLQe+y7km8QsEWoZFp
 qrtsysnpjYvYeVodYZSD6i15iKL6H4<
sn: bind
givenName: wps
cn: wps bind

dn: cn=wpsadmins,cn=groups,dc=ibm,dc=com
objectclass: groupOfUniqueNames
objectclass: top
uniquemember: uid=wpsadmin,cn=users,dc=ibm,dc=com
cn: wpsadmins
```

WebSphere Commerce LDIF sample

Example A-2 contains a sample WebSphere Commerce LDIF file used to import into the IBM SecureWay Directory database for the ITSO working example.

Note: The LDIF sample file wp-itso.ldif,listed in Example A-2 can be found in the `c:\redp3684-code\ldif\wc-itso.ldif`.

This file is based on the WebSphere Commerce sample LDIF file included with the IBM Commerce Enhancement Pack - October 2002 Edition.

Example: A-2 WebSphere Commerce LDIF sample (wc-itso.ldif)

```
version: 1

dn: dc=ibm,dc=com
objectclass: domain
objectclass: top
objectclass: organization
dc: ibm,dc=com
dc: ibm
o: Root Organization
```

```
dn: cn=users,dc=ibm,dc=com
objectclass: container
objectclass: top
objectclass: organization
cn: users
o: Default Organization

dn: cn=groups,dc=ibm,dc=com
objectclass: top
objectclass: container
cn: groups

dn: uid=wcsadmin,dc=ibm,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: ePerson
objectclass: inetOrgPerson
givenname: WCS
uid: wcsadmin
userpassword:
sn: Admin
cn: Admin/WCS
```

B

Debug logon for commerce portlets without single sign-on

This appendix describes how to enable logon of commerce portlets without single sign-on within the commerce enabled portal development environment detailed in Chapter 3, “Implement the development environment” on page 93.

Note: You can find the ITSO provided `WebSphereCommerceBasePortlet.jar` and `WCSLogon.properties` in the `c:\redp3684-code\debug` directory. This code is only meant to be used in a development unit test environment and should not be used in a production runtime environment.

Architectural decisions

This section lists the ITSO architectural decisions for addressing the need for a debug development environment between WebSphere Portal and WebSphere Commerce by enabling logon of commerce portlets without single sign-on.

Table B-1 AD01: Modify commerce portlet base class

Architectural Decision ID	**AD01**
Architectural Decision	Modify commerce portlet base class to transparently call WebSphere Commerce logon command to create WebSphere Commerce session when single sign-on is not available.
Problem Statement	WebSphere Studio Application Developer test environment only runs on WebSphere Application Server Single Server Edition, which does not support LDAP or single sign-on. When running WebSphere Commerce in VisualAge for Java test environment, LDAP and single sign-on are also not supported. An alternative authentication mechanism is necessary to enable debugging of commerce enabled portlets and JSPs.
Assumptions	WebSphere Studio Application Developer is used for commerce portlet development and debugging, VisualAge for Java is used for WebSphere Commerce development and debugging, or both.
Motivation	Enable debugging for commerce enabled portlets and JSPs.
Alternatives	1. Modify commerce portlet base class to transparently call WebSphere Commerce logon command to create WebSphere Commerce session when single sign-on is not available. 2. Deploy portlets on WebSphere Application Server Advanced Edition with LDAP and single sign-on enabled. Use remote Java debugger.
Decision	The first alternative was selected due to full integration with WebSphere Studio Application Developer Portal Toolkit and VisualAge for Java test environment.

Table B-2 AD02: Use WebSphere Portal user id for WebSphere Commerce logon

Architectural Decision ID	**AD02**
Architectural Decision	Use WebSphere Portal user id for WebSphere Commerce logon.
Problem Statement	A user id is needed to call logon command in WebSphere Commerce
Assumptions	Single sign-on is not used.
Motivation	Allow debugging of portlets with several users having different roles.
Alternatives	1. Obtain user id from portal User object and use it when issuing WebSphere Commerce logon command. 2. Read WebSphere Commerce user id from configuration file. 3. Store WebSphere Commerce user ids and passwords in WebSphere Portal credential vault.
Decision	The first alternative was selected due to greater flexibility than second alternative, and smaller amount of code modifications compared to third alternative.

Table B-3 AD03: Store WebSphere Commerce logon URL in a property file

Architectural Decision ID	**AD03**
Architectural Decision	Store WebSphere Commerce logon URL in a property file together with other WebSphere Portal property files.
Problem Statement	The WebSphere Commerce logon URL needs to be known to commerce portlets.
Assumptions	All commerce portlets are accessing the same WebSphere Commerce server, so a single logon URL can be shared across all portlets.
Motivation	

Architectural Decision ID	AD03
Alternatives	1. Store WebSphere Commerce logon URL in a property file. 2. Provide the logon URL as portlet configuration parameter that can be read via PortletSettings.
Decision	The first alternative was chosen because the second scenario requires adding a configuration parameter for every commerce portlet.

Table B-4 AD04: Store WebSphere Commerce logon password in a property file

Architectural Decision ID	AD04
Architectural Decision	Store WebSphere Commerce logon password in a property file together with other WebSphere Portal property files.
Problem Statement	A password is needed to call logon command in WebSphere Commerce machine.
Assumptions	All users defined in WebSphere Commerce test system must have the same password.
Motivation	
Alternatives	1. Store WebSphere Commerce password in property file. 2. Store WebSphere Commerce password in WebSphere Portal credential vault.
Decision	The first alternative was chosen to minimize development effort. To use a similar logon mechanism in production environment, WebSphere Portal credential vault must be used.

Modifications to Commerce Enhancement Pack code

The RemoteServletInvokerPortlet.java was modified to invoke the WCS Logon command if SSO is disabled. The modified code will use the current portal user ID to logon to WebSphere Commerce. Since in development environment (WebSphere Application Server Advanced Single Server Edition) JAAS functionality is limited, it is not possible to get the PasswordCredential object for

the current user. To minimize code modifications, we decided that for development environment it is acceptable to have the same password for all WebSphere Commerce users. The password and WebSphere Commerce logon URL are stored in WCSLogon.properties file located in C:\WebSphere\AppServer\lib\app\config\services\WCSLogon.properties.

When a new HttpClient object is created and `SSOEnabled` flag is set to `false`, WebSphere Commerce logon command is called with the current portal user id and the password read from WCSLogon.properties file. The HTTP response from the logon command contains WebSphere Commerce session cookies. The cookies are automatically stored in HttpClient CookieTable object and included in subsequent requests to WebSphere Commerce. In this case WebSphere Commerce session is managed like a regular user browser session without the need for single sign-on capabilities.

Note: The getHttpClientForthisApplicationSession() method was static in the original code. It was nodified to be non-static to be able to access PortletContext object.

Example: B-1 Modified RemoteServletInvokerPortlet.java code

```
   private synchronized HttpClient
getHttpClientForThisApplicationSession(PortletRequest portletrequest) {
      com.ibm.commerce.portal.wpsapiextensions.ApplicationSession
applicationsession =
         ApplicationPortlet.getApplicationSession(portletrequest, true);
      Object obj = applicationsession.get("httpclientobjectkey");
      if (obj == null) {
         obj = new HttpClient();
         applicationsession.put("httpclientobjectkey", obj);

         if (!isSSOEnabled(portletrequest)) {
            try {
               PortletContext context = getPortletConfig().getContext();
               //Get the Logon URL and password from properties file
               String password =
                  context.getText(
                     "app.config.services.WCSLogon",
                     "WCSPassword",
                     portletrequest.getLocale());
               String logonURLTemplate =
                  context.getText(
                     "app.config.services.WCSLogon",
                     "WCSLogonURL",
                     portletrequest.getLocale());
               String userID = portletrequest.getUser().getUserID();
```

```
                StringBuffer logonURL = new StringBuffer(logonURLTemplate);
                logonURL = logonURL.append("&logonId=").append(userID);
                logonURL = logonURL.append("&logonPassword=").append(password);

                ((HttpClient) obj).getThisURLContents(
                   logonURL.toString(),
                   portletrequest.getClient().getUserAgent(),
                   new Hashtable());
             } catch (Exception e) {
                getPortletLog().error("Could not perform WCS Logon", e);
             }
          }
       }
       return (HttpClient) obj;
    }

    private boolean isSSOEnabled(PortletRequest portletrequest) {
       // The SSOEnabled flag in not visible from this class, so we
       // get it from PortletSettings again
       PortletSettings portletSettings = portletrequest.getPortletSettings();
       String ssoString = portletSettings.getAttribute("SSOEnabled");
       boolean ssoboolean = false;

       if (ssoString != null) {
          ssoboolean = (new Boolean(ssoString)).booleanValue();
       }
       return ssoboolean;
    }
```

Known limitations

The limitations of this solution have been identified as follows:

1. All WebSphere Commerce user IDs used for debugging should match WebSphere Portal user IDs.
2. All WebSphere Commerce user passwords should be the same and match the password defined in WCSLogon.properties file in WebSphere Portal machine.
3. All portlets should have the `SSOEnabled` flag set to either `true` or `false`, otherwise results may be unpredictable.
4. All portlets must point to the same WebSphere Commerce machine.
5. The currently implemented code is designed for use in development environment only, and has several security vulnerabilities. With some modifications it could be made secure for use in production environments

where single sign-on between WebSphere Portal and WebSphere Commerce systems is not possible.

Updating the WSAD plugin with the modified classes

To update the WebSphere Studio Application Developer plugin with the modified classes, do the following:

1. Export the Java project from WebSphere Studio Application Developer to `WebsphereCommerceBasePortlet.jar` file.
2. Copy the exported file to <WSAD_HOME>\plugins\com.ibm.commerce.portal.tooling\data.

   For example, in ITSO development environment the full path of the final file location was as follows:

   ```
   c:\ibm\wsad\plugins\com.ibm.commerce.portal.tooling\data\WebsphereCommerceBasePortlet.jar.
   ```
3. Restart WebSphere Studio Application Developer.

Updating B2B Direct sample with the modified classes

To update the B2B Direct sample with the modified classes (or other samples), do the following:

1. Copy the exported WebsphereCommerceBasePortlet.jar to commerceportal.jar file.
2. Go to C:\temp\CEP\B2BDirectPortal\CommerceRefApp\bin.
3. Extract WebsphereCommerceEnabledPortalB2BDirect.war if not already extracted:

   `jar -xf WebsphereCommerceEnabledPortalB2BDirect.war`
4. Replace WEB-INF/lib/commerceportal.jar file with the modified one.
5. If you will be using these portlets in development environment, edit portlet.xml file and change all `SSOEnabled` attributes to `false` (see Example B-2).

Example: B-2 Update SSOEnabled attributes in portlet.xml

```
<config-param>
   <param-name>SSOEnabled</param-name>
  <param-value>false</param-value>
</config-param>
```

6. Recreate the WebsphereCommerceEnabledPortalB2BDirect.war file:

```
jar -cf WebsphereCommerceEnabledPortalB2BDirect.war WEB-INF
```

7. Deploy the portlets from WebsphereCommerceEnabledPortalB2BDirect.war file.

Configure WebSphere Commerce logon URL, password

To configure the WebSphere Commerce logon URL and password, do the following:

1. In the <WAS_HOME>\lib\app\config directory, create a file called `WCSLogon.properties` as shown in Example B-3.

Example: B-3 Sample WCSLogon.properties

```
WCSPassword=password0
WCSLogonURL=http://ka6brmh:8080/webapp/wcs/stores/servlet/Logon?URL=http://ka6b
rmh:8080/webapp/wcs/stores/dummy.html&storeId=10051&reLogonURL=http://ka6brmh:8
080/webapp/wcs/stores/dummy.html
```

2. Set `WCSPassword` property to your WebSphere Commerce password.
3. Set `WCSLogonURL` property to your WebSphere Commerce logon URL without the `logonId` and `logonPassword` parameters (change hostname). `logonId` and `logonPassword` will be automatically appended to the URL.

C

Tips and troubleshooting for commerce enabled portals

This appendix includes tips and troubleshooting techniques for WebSphere Commerce and WebSphere Portal.

The appendix includes the following topics:

- Resolving issues during installation or testing
- Troubleshooting a store publishing failure
- Backing up a DB2 database
- Restoring a DB2 database
- Resetting a disabled account

Resolving issues during installation or testing

This section describes tips for resolving issues encountered during the installation of the commerce enabled portal runtime and development environment, and during testing.

Cannot start portal using Application Developer

There are several reasons you may not be able start the portal when using WebSphere Studio Application Developer:

- Depending on the error you receive, there could be numerous reasons for this error to occur. One reason could be that the IBM Agent Controller or WebSphere Studio Application Developer was installed prior to WebSphere Application Server or you have multiple installations of WebSphere Application Server on the machine. Depending on when WebSphere Studio Application Developer or the IBM Agent Controller were installed, WebSphere Application Server may be configured to use the WAS_HOME environment variable or it may be configured to point directly to the WebSphere installation directory. If it is configured to use the WAS_HOME environment variable, then WebSphere Application Server may be having difficulties locating this variable or the variable might be incorrect. A work-around is to modify the <AGENT_CONTROLLER_HOME>\config\serviceconfig.xml file (this is typically located in <WSAD_HOME>\IBM Agent Controller. If WebSphere Application Server in different node than WebSphere Studio Application Developer then this path will be where you have installed IBM Agent Controller), and replace all entries of %WAS_HOME% with the location of your WebSphere Application Server installation (for example, C:\WebSphere\AppServer). Then, restart the IBM Agent Controller Service for the changes to take effect. When the IBM Agent Controller is restarted, restart the server.

Example: C-1 Application tag sample for serviceconfig.xml

```
<Application executable="java.exe"
   path="C:\websphere\wsad\jre\bin\java.exe" location="C:\">
   <Variable name="CLASSPATH"
      value="C:\websphere\wsad\IBM Agent Controller\lib\picb.jar"
      position="append"/>
   <Variable name="CLASSPATH"
      value="C:\websphere\wsad\IBM Agent Controller\lib\logutil.jar"
      position="prepend"/>
</Application>
<Application executable="wteRemote.exe"
   path="%WAS_HOME%\java\bin\java.exe" location="%WAS_HOME%">
   <Variable name="CLASSPATH" value="
      C:\websphere\wsad\IBM Agent Controller\lib\wasTools.jar;
```

```
        C:\websphere\wsad\IBM Agent Controller\lib\wteServers.jar;
        C:\websphere\wsad\IBM Agent Controller\lib\logutil.jar;
        %WAS_HOME%\properties;
        %WAS_HOME%\lib\bootstrap.jar"
        position="replace"/>
    <Variable name="PATH" value="%WAS_HOME%\bin" position="append"/>
</Application>
```

- Another problem could be that the IBM Agent Controller is not started on the WebSphere Application Server machine. If you receive an error fairly quickly stating that the server could not be started, then you should check to see if the IBM Agent Controller had started. The IBM Agent Controller is available as a Windows service on the machine that hosts WebSphere Application Server if it was installed properly.
- If you receive an error stating that port `900` or `9000` is already in use, then this might mean that WebSphere Application Server is started outside WebSphere Studio Application Developer. Make sure that the `java.exe` process is not running by checking the Task Manager in Windows or trying to stop the server using the stopServer.bat file in `<WAS_HOME>\bin`.
- If you receive numerous exceptions in the console and the server simply will not start, then possibly the Portal Server instance was not properly configured or you did not reboot the machine after the installation, prior to starting this Portal instance. You should remove the Portal Server instance and configuration, and recreate it following the steps provided previously or reboot the machine and try again.

Portal cannot connect to the database during installation

Depending on the error, there could be multiple reasons for this to occur. Verify the following before attempting to retry the connection:

- Verify that the `<DB2_ROOT>\java12\usejdbc2.bat` file was executed properly so that Portal can locate the proper class files needed to connect to the database. Refer to the product installation help documentation.
- Verify that the user ID and password provided during the Portal installation is the same user ID and password used during the DB2 installation.
- Verify that the DB2 - DB2 service is started in the Windows services.

404 - File not found exception when accessing portal

This error would occur when the hostname or port number specified during the configuration cannot be resolved or was entered incorrectly. Also, if you are using the IBM HTTP Server, verify that it is started or restart it to verify that the WebSphere Application Server plug-in information is reloaded.

System could not log into account when running portlet on the server

When running your portlet on the server, you might get an error indicating that “the system could not log into your account”. The error message would read something like, "Reason: The system could not retrieve your user account information data store. Please try again later." The WPSDEBUG ID is required to log into WebSphere Portal. Verify that the user ID and password are correct when you configured the Portal Server instance, and that the user ID was created correctly in the Configuring the Debug ID in Portal steps provided in this article.

Error occurs when publishing the portlet application

Verify that the directory that you specified when configuring the Portal Server instance already exists. The Portal Toolkit will not create this directory for you.

Publish file, but I do not see my portlet on the page

This might occur when the deployment directory specified during the configuration does not match the location specified to copy the files. The Portal Toolkit would deploy the files, but would not be able to find the files when trying to display them. This might also occur if your debug ID or password does not match the one created earlier. Refer to the section, Configuring a Portal Server instance to verify the steps.

WebSphere Portal installation and configuration

When installing Portal on WebSphere Application Server Advanced Single Server Edition, it is important to note the following:

- To use the Portal Toolkit, you must install Portal on the Advanced Single Server Edition of WebSphere Application Server.
- An HTTP server is not required to use the Portal Toolkit.
- You must copy the personalization.jar and prCommon.jar files to the `<WAS_HOME>\lib\app` directory before installing Portal.
- You must apply WebSphere Application Server fixpack 2 and e-fix PQ56615 **before** installing Portal.
- Use the hostname format of http://yourserver.domain.com:9080 so that you do not need to regenerate the plug-in information while developing portlets.
- You must deploy the base portlets.
- Make sure to run the usejdbc2.bat file in the `<DB2_HOME>\java12` directory to enable the correct JDBC drivers for Portal.

Troubleshooting a store publishing failure

If you experience problems while publishing a store archive, review the troubleshooting techniques outlined in this section.

WebSphere Commerce log files

Review the following WebSphere Commerce log files found in the <WC_HOME>\instances\<instance_name>\logs directory:

- message.txt:
- trace.txt
- ecmsg_<hostname>_<timestamp>.log
- wcs.log

Unlocking a store archive

A store archive may be locked during publishing because someone else is using it, or if an error occurs while accessing the store archive. Before unlocking the file, ensure that no one else is using the file.

To unlock a store archive, delete the lock file in the following directory:

<WC_HOME>\temp\demo\tools\devtools\lock

Republish a contract or account

If a message in the ecmsg_<hostname>_<timestamp>.log reports that publishing has failed while attempting to republish the contract or account asset, do the following before attempting to republish again:

- For contracts:

  Ensure that you have changed the sequence number in the contracts.xml file to the next number in the sequence. The sequence number is a combination of the value for the majorVersionNumber and the minorVersionNumber.

  For example if the majorVersionNumber="1", and the minorVersionNumber="0" change the value for the minor version number to 1. The resulting sequence number will be 1.1.

  When you change the major version number and then republish the store archive, a new contract will be created in the database. This new contract replaces the previous contract.

  If you do not change the major version number, a "contract exists" error will be recorded in the logs when you republish the store archive, and the contract information will not be updated in the database

- For accounts:

  Remove the account.xml file from the store archive and the sarinfo.xml file.

  Publishing is successful but store does not display or does not function properly

  If publish is stated to be successful, but you can't launch the store, or the store isn't functioning properly, check ecmsg_hostname_timestamp.log or ecmsg_instancename_timestamp.log (IBM @server iSeries™) to ensure that the contract and account assets published properly. If not, see "Republishing contract or account" above.

Transaction log for the database is full

If the message, "Transaction log for the database is full," displays in the message.txt log, you have the following options:

Create secondary transaction logs using the DB2 Control Center. For more information, see the DB2 Administration Guide. Increase the transaction log file size by doing the following:

1. In a DB2 command window, type:

   ```
   db2 get db cfg for <wc_database>
   ```

2. Then, look for the log file size (logfilsiz). After finding the log file size, type:

   ```
   db2 update db cfg for <wc_database> using <logfilsiz>
   ```

   Where <logfilsiz> is a larger number then the previous number.

3. Restart DB2.

   or

   In a DB2 command window, type:

   ```
   db2 update db cfg for <wc_database> using LOGPRIMARY 20
   ```

   Where 20 is the number of primary logs (this number may be different for your site). Increasing the LOGPRIMARY increases your space requirement. For more information, see the DB2 Administration Guide.

4. If you are using DB2, you can also drop the following summary tables, using the DB2 command, "DB2 drop table xxx":

   ```
   PRODUCT
   STOREINV
   CATEGORY
   RICHATTR
   RICHATTRCATGP
   ```

Note: The PRODUCT table is most likely to cause the Transaction log to overflow, so delete it first and try to publish the store before deleting the other tables.

Create secondary transaction logs and attempt to publish the store before deleting summary tables.

java.lang.OutofMemory error

If the publishing status in the Store archive list page remains at Publishing, and there is no computer activity indicating that the publish is still in progress, check the wcs.log file. If the message, "java.lang.OutOfMemoryError," or a similar message, displays in the wcs.log file, increase the servlet JVM heap size in the WebSphere Application Server. For more information, see Configuring and tuning WebSphere Commerce.

Backing up a DB2 database

To backup a DB2 database, do the following:

1. Stop the applications that are connected to the DB2 database.
2. Disconnect all applications connected to the database:

   ```
   db2 force applications all
   db2 terminate
   ```

3. Create a database backup directory (for example, `c:\ibm\dbbackup`).
4. Open a DB2 command window.
5. Backup the database, by entering command:

   ```
   db2 backup db <dbname> to <path>
   ```

   For example:

   ```
   db2 backup db wc1db to c:\ibm\dbbackup
   ```

Restoring a DB2 database

As needed, to restore a DB2 database, do the following:

1. Stop the applications that are connected to the DB2 database.
2. If a database exists on the DB2 server that is no longer needed, drop the database as follows from a DB2 command window:

```
db2 drop db <db_name>
```

3. Copy the backup of the database to the DB2 server.
4. To restore the database, enter the following from a DB2 command window:

   ```
   db2 restore db <wc_dbname> from <path>
   ```

   For example:

   ```
   db2 restore db wc1db from c:\ibm\dbbackup
   ```

Resetting a disabled account

In the event that you incorrectly enter your password and the account becomes disabled, do the following:

1. Open a DB2 command window.
2. Connect to the WebSphere Commerce instance database:

   ```
   db2 connect to <db_name>
   ```

3. Enter the following DB2 command to reset the WebSphere Commerce `wcsadmin` logon ID:

   ```
   db2 update USERREG SET STATUS=1, PASSWORDRETRIES=0 WHERE LOGONID='wcsadmin'
   ```

D

Additional material

This Redpaper refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this Redpaper is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

ftp://www.redbooks.ibm.com/redbooks/REDP3684

Alternatively, you can go to the IBM Redbooks Web site at:

ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the redbook form number, REDP3684.

Using the Web material

The additional Web material that accompanies this Redpaper includes the following files:

File name	*Description*
REDP3684.zip	Zipped Code Samples

System requirements for downloading the Web material

The following system configuration is recommended:

Hard disk space: 40 MB minimum
Operating System: Windows 2000
Processor: 1 GHz or higher
Memory: 1 GB or higher

How to use the Web material

Create a subdirectory (folder) on your workstation, and unzip the contents of the Web material zip file into this folder.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this Redpaper.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 265. Note that some of the documents referenced here may be available in softcopy only.

- *WebSphere Commerce Portal V5.4 Solutions Guide, Integrating WebSphere Commerce V5.4, Business Edition and WebSphere Portal V4.2*, SG24-6890
- *WebSphere Commerce V5.4 Handbook: Architecture and Integration Guide*, SG24-6567
- *WebSphere Commerce V5.4 Developers Handbook*, SG24-6190
- *WebSphere Portal V4.1, Windows 2000 Installation,* REDP3593
- *WebSphere Portal V4.1 Developers Handbook*, SG24-6897
- *WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883
- *WebSphere Portal V4.1 Handbook Volume 2*, SG24-6920
- *WebSphere Portal V4.1 Handbook Volume 3*, SG24-6921
- *Access Integration Pattern using IBM WebSphere Portal Server*, SG24-6267
- *Applying Pattern Approaches: Patterns for e-business Series*, ZG24-6722
- *B2B e-commerce With WebSphere Commerce Business Edition V5.4, Patterns for e-business Series*, SG24-6194
- *A Portal Composite Pattern Using WebSphere Portal V4.1*, SG24-6869
- *Mobile Applications with IBM WebSphere Everyplace Access Design and Development*, SG24-6259

Other publications

These publications are also relevant as further information sources:

- *Portlet Development Guide, WebSphere Portal V4.1*, found at:

  ftp://ftp.software.ibm.com/software/webserver/portal/V41PortletDevelopmentGuide.pdf

- IBM WebSphere Commerce product guides found at:

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

  - *Fundamentals Guide, IBM WebSphere Commerce V5.4*
  - *Programmer's Guide, IBM WebSphere Commerce V5.4*
  - *Store Developer's Guide, IBM WebSphere Commerce V5.4*
  - *Installation Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
  - *Additional Software Guide, IBM WebSphere Commerce V5.4 Professional and Business Edition for Windows*
  - *Installation Guide, IBM WebSphere Commerce FixPak V5.4.0.3*
  - *Installation Guide, IBM WebSphere Commerce Studio V5.4 for Windows NT and Windows 2000*

- *Patterns for e-business: A Strategy for Reuse*, by Jonathan Adams, et al. IBM Press, October 2001. ISBN 1931182027
- *Professional WAP*, by Arehart, Charles, et al. Wrox Press, Inc., July 2000. ISBN 1861004044
- *White paper: Pursuing efficiency and revenue with commerce enabled portals* found at:

  http://www.ibm.com/software/webservers/commerce/portal/

Online resources

These Web sites and URLs are also relevant as further information sources:

- Patterns for e-business Web site:

  http://www.ibm.com/developerWorks/patterns/

- WebSphere Commerce V5.4, Business Edition

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

- WebSphere Commerce V5.4, Business Edition Support

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

- WebSphere Commerce V5.4, Professional Edition

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

  http://www.vmware.com
- Symantec Ghost

  http://www.ghost.com
- The Apache Ant Project

  http://ant.apache.org/
- WebSphere Application Server

  http://www-3.ibm.com/software/webservers/appserv/was/support/
- WebSphere Commerce V5.4.0.3 Fix Pack

  http://www.ibm.com/support/docview.wss?rs=497&uid=swg24001839
- Upgrading to WebSphere Application Server

  http://www.ibm.com/support/docview.wss?rs=494&uid=swg21054765
- Commerce Enhancement Pack (April 2003)

  http://www.ibm.com/software/webservers/commerce/epacks/v54
- IBM Directory Server Download

  http://www.ibm.com/software/network/directory/downloads
- FP7_WR21311

  ftp://ftp.software.ibm.com/ps/products/db2/fixes/english-us/db2ntv7/FP7_WR21311/
- jdbctest

  ftp://ftp.software.ibm.com/software/websphere/info/tools/jdbctest
- WebSphereB2CCommerceEnabledPortal.sar Sample Commerce Portal

  http://www-1.ibm.com/support/docview.wss?rs=494&context=SSZLC2&q=publish&uid=swg21083258&loc=en_US&cs=utf-8&lang=en

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Integrating WebSphere Commerce V5.4 and WebSphere Portal V4.1.4

Redpaper

Implement a commerce portal runtime environment

Development environment using VisualAge for Java 4 and WebSphere Studio Application Developer 4

Create and deploy commerce enabled portal B2B store

This Redpaper contains key information that is specific to the integration of WebSphere Commerce V5.4.0.3, Commerce Enhancement Pack - October Edition, and WebSphere Portal V4.1.4 in the runtime environment. We document how to implement a commerce portal development environment with source level debug support using VisualAge for Java V4 and WebSphere Studio Application Developer V4.0.3.

In the latter part of 2002, the redbook *WebSphere Commerce Portal V5.4 Solutions*, SG24-6890, was developed using WebSphere Commerce V5.4.0.3, IBM Commerce Enhancement Pack - October 2002 Edition, WebSphere Portal V4.1.4 Enable, VisualAge for Java 4, Enterprise Edition, and WebSphere Studio Application Developer V4.0.3. That redbook is being refreshed to use WebSphere Commerce V5.4.0.5, IBM Enhancement Pack, April 2003 Edition, WebSphere Portal 4.2, and WebSphere Studio Application Developer V5. We created this Redpaper to preserve the contents of the existing information before the update.

REDP3684

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks