IBM WebSphere Portal V4.1 Handbook
Volume 2

Understand the IBM WebSphere Portal architecture

Step-by-step installation instructions for IBM WebSphere Portal

Implement new and enhanced capabilities of IBM WebSphere Portal

Rufus Credle
Denise Hendriks Hatzidakis
Sunil Hirannah
Gord Niguma
Dwight Norwood
Roshan Rao
Bernhard Stimpfle

ibm.com/redbooks
Note: Before using this information and the product it supports, read the information in “Notices” on page vii.
## Contents

### Notices
- vii

### Trademarks
- viii

### Preface
- ix

- The team that wrote this redbook. x

- Become a published author. xiii

- Comments welcome. xiii

### Chapter 1. Portlet development
- 1
  - 1.1 Basic definitions
    - 1.1.1 Portal. 2
    - 1.1.2 Portlet. 3
    - 1.1.3 Portlet application. 3
  - 1.2 Portlet concepts
    - 1.2.1 Portlet objects. 4
    - 1.2.2 Portlet modes. 6
    - 1.2.3 Portlet states. 7
  - 1.3 Portlet development
    - 1.3.1 Development tools. 8
    - 1.3.2 Portlet development steps. 8
  - 1.4 Available portlets. 23

### Chapter 2. WebSphere Portal administration
- 25
  - 2.1 Introduction
    - 2.1.1 Definitions. 26
    - 2.1.2 Organization. 27
    - 2.1.3 Getting started. 28
  - 2.2 Portlets
    - 2.2.1 Install Portlets. 32
    - 2.2.2 Manage Portlet Applications. 36
    - 2.2.3 Manage Portlets. 50
    - 2.2.4 Web Clipping Portlet. 58
    - 2.2.5 Managing Web Services. 65
    - 2.2.6 Web Services. 69
  - 2.3 Portal Settings
    - 2.3.1 Global Settings. 76
    - 2.3.2 Themes and Skins. 78
    - 2.3.3 Manage Clients. 87
    - 2.3.4 Manage Markups. 90
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 eServer™
- IBM®
- MQSeries®
- pSeries™
- Redbooks (logo)™
- Redbooks™
- Secureway®
- SP1®
- SP2®
- SP™
- Tivoli®
- VisualAge®
- WebSphere®
- xSeries™
- Lotus®
- Lotus Discovery Server™
- Lotus Notes®
- Notes®
- QuickPlace™
- Sametime®
- Lotus Workflow™
- Domino™

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, Windows 2000 and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Red Hat, the Red Hat "Shadow Man" logo, RPM, Maximum RPM, the RPM logo, Linux Library, PowerTools, Linux Undercover, RHmember, RHmember More, Rough Cuts, Rawhide and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.
- Linux is a registered trademark of Linus Torvalds.
- 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

The IBM WebSphere Portal V4.1 Handbook is available in three volumes of Redbooks. This is volume 2.

These IBM Redbooks position the IBM WebSphere Portal for Multiplatforms as a solution that provides a single point of interaction with dynamic information, applications, processes and people to help build successful business-to-employee (B2E), business-to-business (B2B), business-to-consumer (B2C) portals.

WebSphere Portal consists of three packaged offerings:

- Portal Enable
- Portal Extend
- Portal Experience

In the three volumes of the IBM WebSphere Portal V4.1 Handbook, we cover WebSphere Portal Enable and Extend.

The IBM WebSphere Portal V4.1 Handbook will help you to understand the WebSphere Portal architecture, how to install and configure WebSphere Portal, how to administer portal pages using WebSphere Portal; it will also discuss the development of WebSphere Portal portlets and how to use specific WebSphere Portal applications.

Across the volumes of the IBM WebSphere Portal, you will find step-by-step examples and scenarios showing ways to rapidly integrate your enterprise applications into an IBM WebSphere Portal Server environment using state-of-the-art technologies, such as portlets, and implementing new and enhanced capabilities incorporated in the current releases of IBM WebSphere Portal Server offerings, such as access controls and page customization using themes and skins.

In this redbook, we discuss the administration and portlet development of WebSphere Portal. In addition, we discuss the use of Web Services.

A basic knowledge of Java technologies such as servlets, JavaBeans, EJBs, JavaServer Pages (JSPs), as well as XML applications and the terminology used in Web publishing, is assumed.
The team that wrote this redbook

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

Rufus Credle is a Senior I/T Specialist and certified Professional Server Specialist at the International Technical Support Organization, Raleigh Center. He conducts residencies and develops Redbooks about network operating systems, ERP solutions, voice technology, high availability and clustering solutions, Web application servers, pervasive computing, and IBM and OEM e-business applications, all running xSeries systems. Rufus's various positions during his IBM career have included assignments in administration and asset management, systems engineering, sales and marketing, and IT services. He holds a BS degree in business.
management from Saint Augustine’s College. Rufus has been employed at IBM for 22 years.

Denise Hendriks Hatzidakis is a managing director and WebSphere Architect with Perficient, Inc. Denise has a BS in Physics and a BS in Computer Science, as well as an MS in Electrical and Computer Engineering. She joined IBM and spent ten years as a lead developer for VisualAge and WebSphere in various capacities. She has recently joined Perficient, Inc., where she makes extensive use of her skills as a consultant in WebSphere and J2EE technologies.

Sunil Hiranniah is a Software Engineer and works for IBM Developer Relations Technical Support Center in Dallas, USA. He has over five years of experience in the software industry working within various commercial projects. He has wide experience with WebSphere Portal, WebSphere Application Server, J2EE and databases. He has written and published extensively on the WebSphere family of products.

Gord Niguma is an IT Specialist for the Vancouver Innovation Centre in IBM Canada. He has six years of experience in the Web development field, working for customers such as Air Canada and the NHL Players Association. He holds an MS in Computer Science from Simon Fraser University and a BS in Computer Science from Dalhousie University. His areas of expertise include portals and Web content management.

Dwight Norwood is a Director and Senior Consultant for Courtbridge Consulting Group, an IBM Business Partner located in East Granby, Connecticut (U.S.A.). He has 30 years of experience in information technology, with ten years of Lotus Notes and Domino experience. A graduate of the University of Notre Dame, he holds an MS in Computer Science from Rensselaer Polytechnic Institute and an MS in Business Administration from the University of Connecticut. He has written extensively about Notes and Domino development. He has special interests in enterprise knowledge management and publishing, as well as Web-related security.

Roshan Rao is a Senior Consultant with Perficient Inc., with approximately three years of experience in design and development of object-oriented systems. He has a degree in Commerce from the University of Mumbai and is currently pursuing an MS in Computer Science from Maharishi University of Management. He is an IBM Certified Specialist for WebSphere Application Server and MQSeries. His key areas of work include Java technologies, portals, messaging and enterprise application development and integration.

Bernhard Stimpfle is a Pervasive Solutions Architect for the Pervasive Computing Division in Boeblingen, Germany. He reviews architectures, implements customer-specific product add-ons and supports major customers on-site in critical situations. He has spent eight years within the IT industry,
working for DaimlerChrysler Aerospace and managing his own business. His areas of expertise include Pervasive Computing, Unix, Java 2 Enterprise Edition (J2EE) programming, and Solution architectures. He is a Red Hat Certified Engineer (RHCE) and holds a Diplom-Ingenieur degree in Computer Science from Berufsakademie Ravensburg, Germany.

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

Gail Christensen, Cecilia Bardy, Margaret Ticknor, Tamikia Barrow, Diane O'Shea
International Technical Support Organization, Raleigh Center

Mark C Fullerton, Consulting I/T Architect
IBM Ontario

Vishy Gadepalli, Stacy Joines and Sung-Ik So - IBM WebSphere Enablement and Consulting Team
IBM Raleigh

Axel Buecker, ITSO Project Leader
IBM Austin

Stefan Schmitt, Marian Puhl, Ingo Schuster, David S. Faller, WebSphere Portal Development
IBM Boeblingen

Theodore Buckner, IBM Pervasive Computing Division
IBM Raleigh

Frank Seliger, IBM Pervasive Computing Division
IBM Boeblingen

Tim Orlowski, WebSphere Beagle Validation Team Lead
IBM Raleigh

Ulf Arndt, IBM South Africa System Services and Support
IBM South Africa
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 Redbooks to be as helpful as possible. Send us your comments about this 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. HQ7 Building 662
  P.O. Box 12195
  Research Triangle Park, NC 27709-2195
Chapter 1. Portlet development

This chapter introduces portlet development. We shall discuss some basic concepts used in portlet development and walk through the steps for building a simple portlet. This book does not cover Portlet API. For a detailed information and reference to Portlet API, refer to IBM WebSphere Portal Developers Handbook, SG24-6897, available after January 2003.
1.1 Basic definitions

In this section, we discuss the basic definitions for Portal, portlet, and portlet application.

1.1.1 Portal

A portal, as shown in Figure 1-1, is a Web site that provides end users with a single point of access to Web-based resources by aggregating those resources in one place and by requiring that users log in only to the portal itself and not to each application (portlet) they use. Over the years, portals have evolved to provide aggregation of content from sources such as rich text, video, and XML and provide personalized services such as user customization of layout and content.

To accommodate the aggregation and display of such a diverse content, WebSphere Portal Server provides a framework that divides the limited space within a Web page among multiple applications.

Figure 1-1 This is how a portal looks
Characteristics of a portal
The fundamental characteristics of a portal are:

- Information aggregation
- Targeted and personalized information
- Managed content
- Single Sign-On

1.1.2 Portlet
A portlet is an application that is hosted by the portal. In the Portal example shown in Figure 1-1, Welcome, Quicklinks, World Clock, and Reminder are some of the default portlets that come with WebSphere Portal installation. Discussing some of portlet features:

- A portlet is a pluggable component that represents an application.
- From a developer's perspective, a portlet is a Java client that runs on the server.
- A portlet provides output to the user by generating markup output that is assembled into a portal page by the portal.
- A portlet manages the user's preferences for the associated application.

1.1.3 Portlet application
A portlet application is a set of portlets grouped together in an execution context.

- The portlet application provides no code, per se, the application is just a vehicle for grouping portlets.
- Portlets within the same application package share the same context, for example, images, properties files, and classes.
- The set of portlets is packaged into a Web archive file, called a WAR file.
- Portlets in a portlet application may communicate with other portlets in the portlet application using custom messages.

1.2 Portlet concepts
Portlets are more than simple views of existing Web content. Portlets have multiple states and view modes, plus event and messaging capabilities. Portlets run inside the portlet container of a portal server, similar to a servlet running on an application server. The abstract portlet class is the central abstraction of the Portlet API. The portlet class extends HTTPServlet of the Servlet API as shown
in Example 1-1. Therefore, portlets are special types of servlets, with properties that allow them to easily plug into and run in the Portal server.

However, unlike servlets, portlets cannot send redirects or errors to browsers directly, forward requests, or write arbitrary markup to the output stream. This can be done only by the Portal itself, which controls the overall response page.

**Example 1-1** General hierarchy for a portlet class

```
+--javax.servlet.http.HttpServlet
    |   +--org.apache.jetspeed.portlet.Portlet
    |   +--org.apache.jetspeed.portlet.PortletAdapter
    |   +--com.myCompany.myApplication.myPortlet
```

Generally, portlets are administrated more dynamically than servlets. For example, portlet applications consisting of several portlets can be installed or removed while the portal server is running. The settings and access rights of a portlet can be changed by an administrator even in a production environment. The portlet container provides a runtime environment in which portlets are instantiated, used, and finally destroyed. Portlets rely on the portal infrastructure to access user profile information, for communicating with other portlets, accessing remote content, and to store persistent data. In this chapter, we will discuss some basic portlet concepts.

### 1.2.1 Portlet objects

We will discuss some of the control structures accessible by a portlet.

**Note:** Understanding these control structures is essential for knowing Portlet API. However, the scope of this book does not cover Portlet API. The main emphasis is to discuss some of the control structures accessible by a portlet, which will help you to understand portlet development and portal administration.

**PortletConfig**

The portlet container relies on the J2EE architecture implemented by WebSphere Application Server. As a result, portlets are packaged in WAR files similar to J2EE Web applications and are deployed like servlets. Like other servlets, a portlet is defined to the application server using a Web application deployment descriptor, Web.xml.
PortletSettings

In addition to the Web.xml file, the portlet WAR file must also contain a portlet deployment descriptor, Portlet.xml.

Definition: Portlet.xml defines characteristics of the portlet application. Portlet.xml contains configuration parameters like portal application names, portlet titles and other data specific to a particular portlet or portlet application. These configuration parameters are read/write accessible and persistent in the PortletSettings object.

They are changeable by the administrator during runtime. When an administrator deploys a new portlet, or uses the administration user interface to copy an existing portlet, a PortletSettings object is associated with the portlet and a concrete portlet is created. There is no Java object that explicitly represents a concrete portlet.

Definition: The concrete portlet is purely an association of the portlet's Java class instance with a set of configuration parameters. During the lifecycle of a single portlet, many concrete portlets can be created and destroyed. The same concrete portlet can be shared by many users. Each concrete portlet represents one available portlet.

A concrete portlet application is a portlet application parameterized with a single PortletApplicationSettings object. For each portlet application, there may be many concrete portlet applications. A concrete portlet application contains at least one concrete portlet from the portlet application, but it is not required to contain all of them.

PortletData

A concrete portlet is placed on a portal page by a user or an administrator. This creates a concrete portlet instance, which is a concrete portlet parameterized (associated with) a single PortletData object. There can be many concrete portlet instances per concrete portlet. The PortletData object stores persistent information for a portlet on a page. This information cannot be changed by an administrator; it may only be written by the portlet itself. For example, a stock quotes portlet may have an edit page where a user can specify a list of stock symbols to include in the list of quotes. The portlet saves this information in the PortletData object associated with the concrete portlet instance. The scope of

Definition: Web.xml defines the Web application characteristics of the portlet application. It includes portlet class names and portlet configuration data. The portlet can read the configuration data using the PortletConfig object.
the PortletData object depends on the scope of the page containing the concrete portlet instance:

If an administrator puts a concrete portlet on a page accessible by multiple users, then the PortletData object contains data applicable to all users. In the case of the stock quotes portlet, this would mean that every user would see the same list of stock symbols. However, if some of those users have edit access to the portlet, then once the user edits the portlet, a new concrete portlet instance is created. The PortletData object for the new instance contains information for that user alone.

If a concrete portlet is added to a page by a user, the PortletData object contains data for that user alone. The concrete portlet instance is not used by anyone else.

**PortletSession**
The PortletSession object is a subclass of HttpSession. When a user accesses a page that contains a portlet, a user portlet instance is created. A user portlet instance is a concrete portlet instance parameterized by a single PortletSession. There can be many user portlet instances per concrete portlet instance. The PortletSession stores transient information related to a single use of the portlet.

**PortletRequest**
The PortletRequest object is a subclass of the HttpServletRequest. It provides access to attributes (name/value pairs associated with the request), parameters from the URI query string, and other control structures such as the Client, PortletData, and PortletSession objects.

**PortletResponse**
The PortletResponse is a subclass of the HttpServletResponse. It encapsulates information to be returned from the server to the client. It can be used by the portlet to return portlet output using a Java PrintWriter. It provides methods for creating portlet URIs and qualifying portlet markup with the portlet's namespace.

### 1.2.2 Portlet modes

Portlet modes allow a portlet to display different user interface, depending on the task. There are four different modes: View, Edit, Help, and Configure.

**View**
When a portlet is initially constructed on the portal page, it is displayed in its View mode. All portlet's must support view mode. This is the portlet's normal mode of operation.
**Edit**
If Edit mode is supported, the portlet provides a page for users to customize the portlet for their own needs. For example, a news portlet can provide an edit page for a user to enter the number of headlines to be displayed. Edit mode is accessed through the pencil icon on the portlet's title bar.

**Help**
If Help mode is supported, the portlet provides a help page for user's to obtain more information about the portlet. Help mode is accessed through the question mark ("?") icon on the portlet's title bar.

**Configure**
If Configure mode is supported, the portlet provides a page for portal administrators to configure a portlet for a user or a group of users. Configure mode is accessed through a wrench icon on the portlet's title bar.

### 1.2.3 Portlet states
Portlet states allow users to change how the portlet window is displayed within the portal. In a browser, users invoke these states with icons in the portlet title bar in the same way that Windows applications are manipulated. Portlet states are maintained in the PortletWindow.State object with a boolean value. The states are as follows.

**Normal**
When a portlet is initially constructed on the portal page, it is displayed in its normal state, arranged on the page along with other portlets.

**Maximized**
When a portlet is maximized, it is displayed in the entire body of the portal, replacing the view of other portlets.

**Minimized**
When a portlet is minimized, only the portlet title bar is displayed on the portal Page.
1.3 Portlet development

There are five steps required for portlet development.

1. Set up a Portlet Development Environment. You can use any editor such as Wordpad or WSAD for creating portlets.
2. Develop and build the portlet application.
3. Deploy the portlet application for a test.
4. Test the portlet.
5. Deploy the portlet to a Portal production server.

In this section of the chapter, we will show how you can develop and deploy a simple My HelloWorld portlet.

1.3.1 Development tools

You can use a normal text editor like Wordpad or use WebSphere Application Developer for developing portlets. In this chapter, we have used the Wordpad Editor.

WebSphere Studio Application Developer (WSAD)

WSAD is a leading development suite based on WebSphere Studio Workbench. WSAD suite provides a single environment for designing, developing, debugging and deploying J2EE applications. Portlets can be developed using WSAD. WSAD can also be configured with Lotus Sametime toolkit for rapid development of a custom collaborative portlet application.

Portal Development Kit (PDK) plug-in

This plug-in installs the IBM Portlet API, wizards and related documentation into the Studio environment. The PDK provides wizards for developing portlet applications based on different architectural templates and for creating and configuring instances of portal servers.

1.3.2 Portlet development steps

Step 1: Create a directory structure for your portlet

Portlet application directory structure is very similar to the directory structure for a Web application. The directory structure is maintained in the portlet application (Web Archive) WAR file. A Web Archive (WAR) file is Web module containing the application's Web components. A WAR file has a top-level directory, which is the document root.
Create a document root containing the following sub-directories as described in Example 1-2.

**Example 1-2  Create a sample WAR structure**

myhelloWorld\Web-INF - You will place your deployment descriptors (Web.xml and portlet.xml) here.
myhelloWorld\Web-INF\classes - This is where your source will go.
myhelloWorld\Web-INF\lib - This is where you will place the JAR file.
myhelloWorld\META-INF - This is where your manifest file will reside
index.html ...any static/dynamic Web resource(s)

The Web-INF directory may additionally contain Tag library descriptor files (TLD), used for custom JSP tags. The jar utility shipped with Java SDK can be used to create a WAR file. The only difference between a regular JAR file and a WAR is the directory structure. So, you can simply change to the document root of your Web archive structure and issue the proper command.

**Step 2: Create a Java source file**
The Java source file that you will be creating as shown in Example 1-3 needs to go under Web-INF\classes directory. We will create and compile HelloWorld.Java and place it under myhelloWorld\Web-INF\classes.

**Note:** Make sure you use the JDK that comes with WebSphere Application Server.

Once the Java source is compiled, you can create a JAR file in the Web-INF directory. For this portlet development, only one class file will be used and we will not be packaging it as a JAR file.

**Example 1-3  Java Source file**

```java
import com.ibm.wps.portlets.*;
import org.apache.jetspeed.portlet.*;
import org.apache.jetspeed.portlets.*;
import java.io.*;
/* This portlet demonstrates how to display an image in a portlet */
public class HelloWorld extends PortletAdapter {
    public void doView (PortletRequest request, PortletResponse response)
        throws PortletException, IOException{
            PrintWriter pw = response.getWriter();
            pw.println("HelloWorld");
        }
}
```
Step 3: Create deployment descriptors

J2EE applications use deployment descriptors to provide deployment instructions to the application server. Descriptors can also be used to configure and parameterize server-side components. We need to create Web.xml and Portlet.xml. WebSphere Portal V4.1 extends servlets and hence we need to declare the servlet (portlet) class. Save the code in Example 1-4 as Web.xml and place it under myhelloWorld\Web-INF.

Example 1-4   Web.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Web-app
    PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
    "http://Java.sun.com/j2ee/dtds/Web-app_2.2.dtd">
<Web-app id="WebApp_1_1">
    <display-name>Hello Portlet</display-name>
    <servlet id="Servlet_1_1">
        <servlet-name>HelloPortlet</servlet-name>
        <servlet-class>samplepkg.HelloWorld</servlet-class>
        <load-on-startup>0</load-on-startup>
    </servlet>
    <servlet-mapping id="ServletMapping_1_1">
        <servlet-name>HelloPortlet</servlet-name>
        <url-pattern>/HelloPortlet/*</url-pattern>
    </servlet-mapping>
</Web-app>
```

The important aspect of this descriptor is the id attribute of the servlet tag that would be used to map the servlet definition to a portlet definition in a portlet application.

Save the code in Example 1-5 as portlet.xml and place it under myhelloWorld\Web-INF.

Example 1-5   portlet.xml

```xml
!DOCTYPE portlet-app-def PUBLIC "-//IBM//DTD Portlet Application 1.1//EN" "portlet_1.1.dtd">
<portlet-app-def
    <portlet-app uid="samplepkg.HelloWorld" major-version="41" minor-version="0">
        <portlet-app-name>Hello World (Sample code)</portlet-app-name>
        <portlet id="Portlet_1_1" href="/Web-INF/Web.xml#Servlet_1_1">
            <portlet-name>My HelloWorld portlet</portlet-name>
            <cache>
                <expires>0</expires>
                <shared>NO</shared>
```
Let us discuss some of the concepts from the portlet.xml file.

**Note:** We have placed a working myhello.war file in the Additional Materials section of the IBM WebSphere Portal V4.1 Handbook Volume 2 redbook posted at the IBM Redbooks Web site. WebSphere Studio Application Developer 4.0.3 with the Portal Toolkit installed was used to create this file.

You must obtain the Portal Toolkit from CD 3-3 and install it in WebSphere Studio Application Developer before trying to write a portlet.

**<portlet-app-def>**
A required, top level element that contains information about the portlet application. This element includes exactly one <portlet-app> element and one or more <concrete-portlet-app> elements.
<portlet-app uid="uid">
Required. The tag provides the means to package a group of related portlets that share the same context. The context contains all resources, for example, images, properties files, and classes. All portlets must be packaged as part of a portlet application. The uid for each portlet must be unique within the portlet application.
</portlet-app>

<portlet id="id" href="href">
At least one is required. Contains elements describing a portlet that belongs to this portlet application. id and href are required. The id must be unique within the portlet application. The href attribute points to the identifier of the servlet, as in Web-INF/Web.xml#servlet_id, for example, mapping to a servlet defined in the Web application.
</portlet>

<markup name="name">
At least one is required. This indicates the type of markup this portlet supports. Name can have one of the following values: html, wml, chtml. The markup tag can have the sub-elements <view/> (required), <edit/>, <help/> and <configure/>. These tags indicate the modes that the portlet supports.
</markup>

<concrete-portlet-app uid="uid">
A concrete portlet application contains at least one portlet from the portlet application, but it is not required to contain all of them. The following are sub-elements of <concrete-portlet-app>.
</concrete-portlet-app>

<context-param>
Optional. This contains a pair of <param-name> and <param-value> elements that this concrete portlet application can accept as input parameters. A concrete portlet application can accept any number of context parameters. Administrators can change the context parameters when they configure the concrete portlet application. Provide help information using XML comments to explain what values the portlet application can accept. The unique configuration settings for a concrete portlet application make up its PortletApplicationSettings.
</context-param>

<concrete-portlet id="id" href="href">
At least one is required. This contains elements describing the concrete portlet that belongs to this concrete portlet application. id and href are required. The id must be unique within the portlet application. The href attribute points to the identifier of the portlet, as in #portlet_id.
</concrete-portlet>
Guidelines for portlet application UIDs
The UIDs of portlet applications and concrete portlet applications must identify
them unambiguously in the area of their usage, which could be worldwide.
Hence, it is strongly recommended to follow these guidelines.

- Include the portlet's namespace in the UID, using the same format that is
  used for Java packages
- Add portlet application specific description
- Add arbitrary characters to guarantee uniqueness within the namespace, for
  example: com.ibm.wps.samplet.mail.4000
- Add postfixes for the corresponding concrete portlet applications, for
  example: com.ibm.wps.samplet.mail.4000.9
- Portlet IDs must be unique within the application.

Step 4: Create a WAR file
You will need to package the source file and deployment descriptors that we
created as a "WAR" file, before we can deploy the Portlet. We will use the
standard jar command to build the WAR file. Run this from the myhelloworld
directory.

```
set Java_HOME=C:\WebSphere\AppServer\Java
set PATH=%Java_HOME%\bin
jar -cf myhello.war Web-INF
```

Step 5: Deploy the WAR file
There are two ways to deploy a portlet application into WebSphere Portal:
1. Portal administration portlet
2. Portal configuration interface (XML Access)

**Installing portlets using Portal administration portlet**
Let us install the myhello.war file that we created using Portal Administration
portlet. Portal configuration interface will be explained, but we will not install
myhello.war using this functionality.
1. Log in to WebSphere Portal as shown in Figure 1-2 as Portal administrator.

**Note:** It is assumed that you have a successful installation of WebSphere Portal and have administrative privilege for installing portlets.

2. Once you successfully log in, you will see the Portal Welcome Page. On the top left-hand corner of the welcome page, select **Portal Administration**. Portlets Page will be the default. Select **Install Portlets** portlet as shown in Figure 1-3 and browse for myhello.war. Click **Next**.

**Note:** In this example, we have used the user name wpsadmin and password wpsadmin.
3. Check for the portlets that will be installed as shown in Figure 1-4. You can verify based on your portlet.xml description. Click Install.
4. If the portlet installation is successful, you should see a message: Portlets Successfully Installed. If the portlet installation is a failure, check for the Portal Server logs directory and check for the latest log file located under \WebSphere\PortalServer\logs\. At this stage, you have deployed the portlet and now we need to add this to a page.

**Tip:** The name of the log file can be determined with the append of latest time and date stamp on it. (for example, wps_2002.07.27-11.00.47.log)

5. Before you can add a portlet to a portal, you need to determine where to put it in the portal. Portal server has the concept of places (WebSphere Portal Extend version) and pages (WebSphere Portal Enable version). Users navigate through the portal by accessing different places, and then selecting pages within each place. Places can be managed as a unit and you can change the order of places within the portal. Pages are added to places. When defining a page, you identify the layout (rows and columns) for the page. After a portlet is installed, to use the portlet, you must add it to a page.
All resources within the portal, including places, pages, and portlets are subject to access control. For this example, we will add the Portlet that we installed to the Portal Welcome Page.

6. Select **Work with Pages page** group by browsing on the top left-hand drop box option. On the Edit Layout and Content page, under Page Group, select the **Home page** group. Then select the **Welcome page**.

7. Once you select the Welcome Page, you should see a window as shown in the Figure 1-5. Search for the portlet we developed using **GO**. Select **My HelloWorld portlet**. By clicking the plus button, add this to the portlet list. Click **OK** to continue.

![Image](image-url)

Figure 1-5  Search for the Portlet that was installed for adding to the Portal page

8. You can add the My HelloWorld portlet to any part of the Welcome Page. Select the **My HelloWorld** portlet and click the **Add** button on the left or right column of the page. We will add it to the left column of the page as shown in Figure 1-6. Click **Activate** to activate the page with the changes.
9. After clicking **Activate**, your window must show **Deactivate** before choosing the **Home** option. To test how this portlet looks, select **Home** from the top left-hand drop-down option; you can see the Welcome Page with your added portlet as shown in Figure 1-7.
You have successfully developed, packaged and deployed your first portlet.

**Installing portlets using the portlet configuration interface**

The portal configuration interface provides a batch processing interface for portal configuration updates. It allows you to export entire portal configurations or parts of the configuration (for example, specific pages) to an XML file and to re-create the exported configurations from such a file. This technique involves creating an XML descriptor file specifying the portlets to install and/or the pages/places to create/use, and running the XMLAccess utility.

The typical tasks to perform using the portal configuration interface are:

- Backing up and/or restoring entire portal configurations.

**Note:** A full XML export of a portal configuration is not sufficient to re-create the portal. You will also need WAR files for your portlets and possibly additional file resources such as theme files if they are not part of the standard portal installation.
Copying parts of a configuration, such as specific pages and places from one portal to another.

XML Access is a command line utility which is available as a Java interface that ships with WebSphere Portal. This utility is a small HTTP client to portal server that can also be copied from the machine where WebSphere Portal is installed to another machine, and run to update portal configuration. When the XML request has been processed on the server, the resulting XML output is sent back to the client and written to the standard output, which can be redirected to an XML file. The XML Access tool can be invoked by running PortalServer\bin\xmlaccess.bat file and syntax as follows:

```
xmlacess <XML file> <userid:password> <portal config URL>
```

Where
- XML file is XML request file name
- userid and password should be the user ID and password of the portal user with manage rights on any portal
- portal config URL consists of host name, base URI appended with /config

The example for the above command is as follows:

```
xmlacess input.xml wpsadmin:wpsadminhttp://sunil.svo.dfw.ibm.com/wps/portal/config
```

Specifying XML Schema and Root element
The input request XML file can be created using any text editor and it should follow the specifications of the PortalConfig.dtd located in the PortalServer\app\wps.ear\wps.war\dtd\ directory. The root element of this XML document is named <request>. The <portal> sub-element is used to import or export portal resources. The encoding of the input XML must always be UTF-8, as shown in Example 1-6.

```
Example 1-6   input.xml file
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE request PUBLIC "-//IBM//DTD Portal Configuration 1.0//EN" "PortalConfig.dtd">
<request>
  <portal action="locate">
    <!-- XML fragments detailed in the following sections to be inserted here. -->
  </portal>
</request>
```
Installing the portlet using XML Access

The `<package>` element as shown in Example 1-7 corresponds to an abstract portlet application and it is used to install portlet applications. The action attribute of the `<package>` element can be either to **update** or to **install** a portlet application. If the portlet application is already installed, the action attribute with a value of `create` might fail, since the `globalid` attribute should be unique. Its `<URL>` sub-element points to the location of WAR file to be installed.

Here are the matching rules between elements of the request XML file and those of the `portlet.xml`:

- The `<package>` element's `globalid` attribute should be the same as the `<portlet-app>` element's `uid` attribute specified in `portlet.xml`.
- The `<application>` element's `globalid` attribute should be same as the `<concrete-portlet-app>` element's `uid` attribute specified in `portlet.xml`.
- The `<portlet>` element's `name` attribute should be the same as the contents of the `<portlet-name>` element specified in `portlet.xml`.

**Example 1-7 request XML file**

```xml
<package action="create" active="true" globalid="helloworldpkg.EditPortlet">
  <url>file:///d:\tmp\myhello.war</url>
  <application action="create" active="true" globalid="helloworldpkg.CreatePortlet.concrete">
    <access-right permission="delegate" subjectid="wpsadmin" subjecttype="user" update="set"/>
    <access-right permission="manage" subjectid="wpsadmin" subjecttype="user" update="set"/>
    <access-right permission="delegate" subjectid="wpsadmins" subjecttype="user-group" update="set"/>
    <access-right permission="manage" subjectid="wpsadmins" subjecttype="user-group" update="set"/>
  </application>
  <portlet action="update" handle="editmemoportlet" active="true" name="Edit Memo concrete">
    <access-right permission="delegate" subjectid="wpsadmin" subjecttype="user" update="set"/>
    <access-right permission="manage" subjectid="wpsadmin" subjecttype="user" update="set"/>
    <access-right permission="delegate" subjectid="wpsadmins" subjecttype="user-group" update="set"/>
    <access-right permission="manage" subjectid="wpsadmins" subjecttype="user-group" update="set"/>
  </portlet>
</package>
```
The XML configuration client interface allows exporting entire or partial configuration as an XML file and re-creating configuration by importing the XML file. This interface can also be used as an alternative to perform some of the administrative tasks which are part of the Portal Administration page and Work with Pages place. The supported tasks of XML Access interface include creating and updating following resources.

- Portlet applications
- Portlets
- Themes and skins
- Markups and client device definitions
- Places and pages
- Credential Segments and Slots

1. Embed the contents inside the <portal> element as shown in Figure 1-6 on page 18.
2. Save the resulting file as input.xml.
3. Open a command window and change to the directory where you saved the input.xml file.
4. Add the Portal server's bin directory to your PATH environment in order to locate the XML Access tool.
5. Run the following command by substituting the appropriate user ID, password, and server URL.

```
xmaccess input.xml userid:password http://machine.domain.com/wps/config
```

XML Access opens an URL connection to a servlet that serves the Webpath /wps/config and sends the input.xml to the servlet. The servlet verifies the validity of the XML file and updates the portal server. The output of the command itself is XML data that follows the same schema specified in the input.xml. Once you have installed the portlet, you can follow the same steps as described in Step 5 of the Installing portlets using Portal administration portlet section.

**Note:** More information on XML Access is available in the WebSphere InfoCenter.
1.4 Available portlets

WebSphere Portal installation comes with a rich set of standard portlets for displaying syndicated content, performing XSL transformation, accessing existing Web pages, Lotus Notes and Microsoft Exchange productivity portlets, Sametime instant messaging, and Lotus QuickPlace team rooms.

Companies can also create their own portlets or select from a catalog of portlets created by IBM and IBM business partners. You can find this information at:

PortletDevelopmentGuide.doc
WebSphere Portal administration

This chapter describes how to work with the administration portlets provided by WebSphere Portal.
2.1 Introduction

In WebSphere Portal V4.1, administration of the portal is done through the portal itself, either in a centralized or delegated fashion. The administration interface for Portal Server enables quick access to the administration portlets and greatly simplifies the task of administering the portal. Administrators can deliver a new service to users simply by adding new portlets to the pages of the portal. Since these are portlets, just like bookmarks or reminders or news or any other portlets, administrators can control access to them, place them on portal pages, and perform any of the usual steps.

2.1.1 Definitions

You will need to know some of the basic definitions before you start working with Portal administration pages.

Portlet
From a portal administrator's point of view, a portlet is a content container to which users can subscribe. WebSphere Portal administration functionality is delivered via portlets.

Page
A page is a collection of portlets and containers. A page contains one or more portlets and containers.

Pages are subject to the following guidelines:
- A customized page is saved on a per-user and per-page basis.
- Pages, once created, are automatically set to the active state, but with no permissions
- When a page is edited, it is automatically deactivated.
- Pages are made up of row containers and column containers.
- Containers and container content can be locked.
  - Manage rights required
  - Locked containers cannot be deleted
  - Locked container content cannot be moved or deleted
- Place versus page: if you have installed WebSphere Portal V 4.1 Extend, you will see the word *place*, and if you have installed WebSphere Portal V 4.1 Enable, you will see the word *page*. Both have the same meaning.
Page group
This is a collection of pages. In WebSphere Portal, one or more pages that are grouped under one tab are called a page group. Page groups provide a new level of page categorization. Pages can be grouped together and managed as a unit. The way pages are grouped is arbitrary and left up to the page group creator.

- All pages must exist in a page group and there is no way to move or copy page groups between page groups, but it is possible to copy pages.
- Page groups provide multilevel tab functionality.
- Deleting a page group deletes all pages in it.

2.1.2 Organization
Portlets are laid out on pages. The Portal administration pages use a Portlet Selector portlet to provide menu-like access to portlets on the page.

Page groups installed by default within WebSphere Portal are:
- Home (Public)
- Work with Pages (Customizer)
- Portal Administration

Use the Portal Administration pages to:
- Install portlet applications on the portal
- Manage installed portlet applications and portlets
- Publish portlets to a Web service
- Configure the portal
- Define the users and user groups for the portal
- Control which portal resources users and user groups can access

Page groups can have specific access control applied to them. In most cases, only Portal administrators or sub-administrators will have access to the Portal Administration page.

WebSphere Portal 4.1 provides a Page Group called Portal Administration, which allows the Portal administrator to install portlets, create themes and skins, work with users and groups, and secure portlets and ACLs. The Portal Administration page group contains the following portlet pages, which will be discussed in this chapter:
- Portlets
- Portal Settings
The sections are organized in such a manner that administrative functionalities offered by these individual portlets are discussed.

**Note:** Portal administration can also be done using XMLAccess as explained in “Installing portlets using the portlet configuration interface” on page 19. In this chapter, we will focus only on using administrative portlets for portal administration.

---

### 2.1.3 Getting started

**Important:** Before you start working on Portal administration, make sure that you have successfully installed WebSphere Portal and you have the information of a user who has administrative privileges to log in.

To get to the WebSphere Portal Administration page, do the following.

1. Open a Web browser and type in the URL for the login page to WebSphere Portal as shown in Figure 2-1. Click **Log in** to proceed. The Cancel button will take you to WebSphere Portal Welcome page.

**Note:**

- The default user with administrative privilege is generally wpsadmin.
- In our example, we will login with the user wpsadmin and password wpsadmin.
- You can also login to the default portal page http://completedomainname/wps/portal and click the login icon (key symbol) provided on then right-hand side of the page.
2. If you have successfully logged in, you should see the WebSphere Portal Welcome Page as shown in Figure 2-2. On the top left-hand side as shown in the image, select **Portal Administration** page from the drop down list.
3. WebSphere Portal Administration page will open and you should see a window as shown in Figure 2-3.
Figure 2-3  WebSphere Portal Administration Page

**Note:** If you get the error message *There are no Portlets available on this page*, refer to the redbook *IBM WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883, chapter 5, for installation tips.

**Portal Administration Help option**

The Help menu icon (?) is provided on all the Portal Administration pages. When you click this icon, a window pops up with the product documentation information also known as the InfoCenter.

There is also a Help icon (?) on individual Administration Portlets and clicking this icon will get you the product information specific to that particular portlet.

**Navigation:** at any time, you can select any of the administrative portlets by selecting the appropriate tabs on the Portal Administration page.
2.2 Portlets

The Portlets Page contains the following portlets:

- Install Portlets
- Manage Portlet Applications
- Manage Portlets
- Web Clipping
- Web Services
- Manage Web Services

We will explore the above portlet applications individually.

2.2.1 Install Portlets

This feature will help you to install a portlet application. Portlet application is installed through a Web Archive (WAR) file or install remote portlets via UDDI directory (Web Services portlet). The WAR file, which is used to install portlet application, can contain multiple portlets. The install process uploads the WAR file to the server, installs portlets, adds them to the list of available portlets and activates the portlets. Once you install a portlet, it is automatically activated but with no permissions. Use the Access Control portlet to determine which users and groups can view, edit, or manage the new portlet.

Note: For this section, we have used the readparameters.war file. We will walk and explain different portal administration capabilities using this war file. You can refer to Appendix A., “WebSphere Portal Administration sample code” on page 197 for the code. You can package the capabilities as WAR files and save them in a directory on your machine, then install. This is just an example we have used, and users can incorporate administration functionality for their own code.
Perform the following instructions:

1. Select **Install Portlets** portlet. Browse for the WAR file as shown in Figure 2-4. Click **Next**.

   **Important:** This WAR file for installing the portlet application should be in the local directory. The portal administrator should be installing the portlet application on the same machine as that of the portal server. Installing the portlet from a remote machine will fail.

   ![Figure 2-4](image)

   **Figure 2-4**   Browse for the WAR file for installing your portlet

2. Check for the list of the Portlets included in the WAR file as shown in Figure 2-5. In our example, **Read Concrete Portlet 1** is selected for installation. Click **Install** to begin the installation. You can click **Cancel** anytime to stop the installation process.
3. When installation is complete, if successful, you should get the message "Portlets Successfully Installed" as shown in Figure 2-6. You can click Next if you want to install more portlets.

Tip: If portlet installation is a failure, check for the portal server logs directory and check for the latest log file located under \WebSphere\PortalServer\logs\. The name of the log file can be determined with the append of the latest time and date stamp on it (for example, wps_2002.07.27-11.00.47.log).
We have successfully installed the Read Concrete portlet. We will add this portlet into a page following the steps as explained in 1.3.2, “Portlet development steps” on page 8. Once we add it, we should see the page with the Read Concrete portlet as shown in Figure 2-7. The portlet is designed to display the context and config parameters.
2.2.2 Manage Portlet Applications

Manage Portlet Applications helps you to identify and manage the existing installed Web modules (WAR file). It also displays the concrete portlet application corresponding to the selected Web module. Using this Portlet, you can uninstall the portlet application and modify dynamically configured parameters.

Select the Manage Portlet Applications portlet and with the Web module readparameters.war file as shown in Figure 2-8, you can:

- Show Info
- Update
- Uninstall

Web modules can contain one or more portlet applications, servlets JSP files and other files and are defined in the Web descriptor file (Web.xml).
With the portlet applications belonging to the selected module, you can:

- Activate/Deactivate
- Rename
- Copy
- Modify parameters
- Show Info
- Delete

Portal applications can contain one or more portlets. They are created implicitly when the WAR file is deployed and they are packaged as an enterprise application (ear file). You will see the default Web modules in the following figure. These are installed during WebSphere Portal installation.

![Figure 2-8 Manage Web modules and portlet applications](image)
Show Info
Show Info describes the content of the WAR file (Web module), abstract Portlet application and the abstract Portlet. (Complete Portlet application).

1. Select the readparameters.war file and click Show Info.

   This will show the selected Web module, portlet application name, concrete portlet applications belonging to the Web module, and portlets as shown in Figure 2-9.

2. Click Done to come back to Manage Portlet.

   ![Figure 2-9](image)
   Select the war file to get details

Update
The Update option helps you to modify your existing portlet application without uninstalling your existing portlet application.

Note: Update functionality includes updating configuration parameters in your portlet and replacing the portlet code with new code, incorporating all the changes.
3. Select the readparameters.war file. We will show how update functionality works.

4. Click **Update**; it will take you to a window as shown in Figure 2-10.

![Figure 2-10 Browse for the war file for updating your portlet](image)

5. Enter or browse for the updated WAR file (updatedsetparams.war file) location. In this example, we have changed the code in the readparameters.war file and renamed it as the updatedsetparams.war file. You can see the modifications to the code in Appendix A., “WebSphere Portal Administration sample code” on page 197.

6. Click **Next**. You can hit **Cancel** to return.

7. You will get a window as shown in Figure 2-11 highlighting the portlets that will be installed during the update. Check for accuracy and select the **Install** option. You can select **Cancel** to return.
8. If the WAR file is successfully updated, you should see the updated `setparams.war` file as shown in Figure 2-12 and the message "The Web module was updated successfully" at the bottom of the page.
Let us check the changes made to our portlet after using this Update functionality. Go back to the drop-down menu on the top-left hand corner of the page and select your page where you had installed the Read Parameter portlet. When you bring up that page, you should see the window shown in Figure 2-13.

**Tip:** It is not required for you to add the portlet again to the page after doing an Update. Changes are incorporated to the page where the portlet was installed automatically.

**Note:** In our example, we have renamed the readparameters.war file to updatedsetparams.war file after doing changes with the code. There is no need to change the war file name for using Update administrative functionality. We have used it for convenience's sake.
Figure 2-13  Portlet with changes after using Update administrative functionality

Note: Compare Figure 2-13 with Figure 2-7 and you will see that one more configparam value is added to the portlet.

The value of config param configParam2 is value2 - New line.

Uninstall

The Uninstall option helps to uninstall your existing portlet application.

1. Highlight the Web module (updatesetparams.war) to uninstall.

   A confirmation window will prompt for confirmation. Click OK if you want to uninstall or click Cancel to return.

2. You will return to the Manage Application Portlet and on the bottom of the Portlet, you can read the message The Web module was uninstalled successfully and will be removed from the Web module section and the page where the portlet would have been deployed.
Activate/Deactivate

The Deactivate feature helps to temporarily suspend access to your selected portlet application and then, with activating, provide access to the portlet application.

1. Highlight the portlet application to activate or deactivate. You will see in Figure 2-14 that updatesetparams.war will be highlighted. Select the portlet application corresponding to this WAR file (Read config and concrete param concrete).

**Important:** If you do not select any portlet application, a window will prompt you to choose before you proceed.

2. In our example, the portlet application corresponding to updatesetparams.war file is active. Click the **Activate/Deactivate** button and you can see the portlet application being deactivated as shown in Figure 2-15.
3. You can click the **Activate/Deactivate** button to Activate the Portlet application.

**Tip:** Once you deactivate your portlet application, all the portlets that are part of the deactivated application will disappear from your customized portal page.

**Rename**

You can rename a concrete portlet application.

**Purpose:** When you clone a portlet application, you may wish to rename one of the portlet application to avoid duplicate names. The Rename option helps with this functionality.

1. Highlight the portlet application corresponding to updatesetparams.war. Select **Rename**.

A pop-up window will open asking you to provide a new name.
2. Enter the new name and click OK.
3. You will see the new changed name as shown in Figure 2-16.

![Figure 2-16 Rename your portlet application](image)

**Copy (Cloning)**

This option helps to copy your concrete portlet application.

**Note:** This is useful when different portlet configuration parameters are required for different instances of a portlet.

You can activate or deactivate based upon your requirements. When you copy a Portal application, the newly created application is active by default. However, portlets that are part of the newly created Portal application are inactive. To customize this Portal application, you will have to activate it, which will be shown in the Manage Portlets portlet.

1. Highlight the portlet application corresponding to updatesetparams.war. Select **Copy**. A window will prompt you to enter the name for the copy and click **OK**. You can hit **Cancel** to avoid copying.
2. Once the application is copied, you should see the changes as shown in Figure 2-17.

![Figure 2-17 Copy your concrete portlet application](image)

3. You should see the new copied concrete portlet application, Test for cloning.

   **Note:** Compare Figure 9-16 and Figure 9-17 and you will notice a difference. You will find a new portlet application for the updatedsetparams.war Web module.

### Modify Parameters

The Modify Parameters option allows you to modify the configuration parameters of the portlet application. Parameters are originally set by portlet.xml for that instance.

1. Highlight the portlet application (updatedsetparams.war) you want to modify. Select **Modify parameters**.
2. You will see the portlet application name and existing parameter values.

![Image of WebSphere Portal administration](image)

**Figure 2-18  Modify Parameters for your portlet application**

3. To add a new parameter and value, enter the new values. We will change the `contextparam` value (change `app1` to `app2`).

**Note:**
- Context param is defined at the concrete Portlet application level so that all concrete portlets that are part of that application can access `getportletsetting().getportletapplicationsettings.getattribute`.
- Config param is defined per concrete portlet, which can only be accessed from that concrete portlet and using `getportletsettings.getattribute`.

4. Click **Add**. Select **Save**. The parameter and value are saved.

5. Once when you have made the changes or added a new parameter value, you can delete and close the window if no modifications are required.

6. Select **Close** to return to the Manage Portlets page.
7. To test, select the page where the Read Parameter portlet is installed and you should see the parameter modification changes reflected as in Figure 2-19.

![Figure 2-19 Changes to your portlet after modifying the parameters](image)

**Tip:** The character limit and character type are dependent on the database setup you have. Generally, the parameter name can be up to 64 characters and the value can be up to 255 characters. Both can contain letters, numbers, or other characters.

**Show Info**
This option shows information for each concrete portlet application. It displays the names of the concrete portlets that are part of the selected portlet application.

1. Select the concrete portlet application corresponding to the Web module (the portlet application in our example: this is a test for you to see the change) and click **Show Info**.
2. You should see a window open as shown in the Figure 2-20 with information on the portlet application name and corresponding portlets. This provides details on the concrete portlet application.

![Figure 2-20 Show Info for portlet application](image)

3. Click **Done** to return back to Manage Portlet Applications portlet.

**Delete**

This option deletes the Portlet Application.

1. Select the portlet application that you wish to delete. Click the **Delete(X)** button.

2. A prompt window will appear to confirm. Click **OK** or **Cancel**, depending on your requirement.

3. If the deletion was successful, you will not see the portlet application under the Manage Portlet Applications portlet.
2.2.3 Manage Portlets

Manage Portlets allows you to selectively activate, deactivate, rename, copy, delete portlets and modify portlet parameters instead of portlet applications as we did in the previous section.

- You can take the default setting for Manage Portlet, by displaying all of the portlets as shown in Figure 2-21.
- You can also search for portlets by specifying the search criteria (Active/Inactive state) and clicking the Go button.

Note: When you use the default of displaying all Portlets, the other selection options are greyed out.

Figure 2-21 Manage Portlets
Activate/Deactivate

In “Manage Portlets” on page 50, we have shown how to copy your concrete portlet application. Portlets that are part of the newly created Portal application are inactive, as shown in Figure 2-22.

1. You can select the portlet you want to activate/deactivate and click Activate/Deactivate.

![Figure 2-22 Activate portlets that part of the newly created portlet application](image)

2. Once you select the Activate/Deactivate option, the page will refresh and you should see the portlet Read Parameters Concrete Portlet 1_Test as Active.

Rename

You can rename your portlet.

1. Highlight Read Parameters Concrete Portlet 1.
2. Click the Rename option.
3. A pop-up window will appear asking you to provide a new name.
4. In this example, we changed it to Hello Reader. Click OK to accept and Cancel to return.

5. If you click OK, the Manage Portlet page will refresh and you should see the Hello Reader portlet as shown in Figure 2-23.

   ![Figure 2-23 Rename your portlet](image)

**Copy**
This option copies a portlet.

1. Highlight the Hello Reader portlet.
2. Click the Copy option.
3. You will be prompted with a window asking for a name for the portlet after it is copied. In this case, we have used Hello Reader 25. Click OK to continue or Cancel to return.
4. Once the portlet is copied, the Manage Portlet page is refreshed and you will see Hello Reader 25 portlet as shown in Figure 2-24 with an Inactive state. You can click Activate/Deactivate to activate this newly created portlet.
Modify Parameters
Modify Parameters allows you to modify the parameter values of your portlet.

1. Select the Hello Reader portlet.
2. Click Modify parameters.
3. You will see a window as shown in Figure 2-25 with portlet configuration parameters and titles. Select the parameter that requires editing. Enter the new parameter or value. We have modified the values for configParam1 and configParam2.
4. Add new parameters as you wish and click the Add button.

5. Edit Locale Specific Titles will help you change your Portlet Title. Select the Title radio button and click Set title for selected locale.

6. A new window will open as shown in Figure 2-26. Click OK and you will return to the Portlet configure parameter and title page.

**Note:** Changing the title option is not mandatory. It can be done based on individual requirements.
Figure 2-26  Change title for your portlet

7. Click the **Save** button and then the **Close** button.
   You will be taken back to Manage Portlets.

8. To test, go back to the page where you have installed the Read Parameter portlet; you see the portlet with the changed title and changed parameter values as shown in Figure 2-27.
Figure 2-27 Portlet with changed title

Show Info
This option shows the portlet name, portlet title, and portlet description.

1. Highlight the Hello Reader portlet.
2. Click Show info.
   You should see a window as shown Figure 2-28 with the portlet information for the selected portlet.
3. Click Done to return to Manage Portlets.
Delete

You can delete any portlet.

1. Select Hello Reader 25.
2. Click Delete(X).
3. You will get a pop-up window for confirmation. Click OK to confirm and Cancel to return.
4. The Manage Portlets page is refreshed and Hello Reader 25 is deleted.
2.2.4 Web Clipping Portlet

The Web Clipping Portlet allows Web content from other sites to be clipped and displayed within the portlet on a portal page.

With the Web Clipping Portlet Administration:

- Sections of existing Web pages are displayed, visually or between tags.
- Links can be displayed without leaving the portal.
- Each clip creates a new portlet.
- The current version of the Web page is retrieved.
- There is no security, basic authorization or forms-based authentication.
- Credentials are supplied by the user or administrator.

The Web Clipping Portlet uses Transcoding Technology, which allows the administrator to use the front-end user-interface; it provides the ability to create new portlets and to wrap contents by specifying particular URL information. It allows you to identify and extract specific portions of an HTML document as desired by the administrator. Links have been rewritten to go through the Portal.

Two portlets that are involved in Web clipping are:

**Tooling Portlet** (clipping editor)

- Identify and extract specific portions of a document for display in a portlet
- Use to visually specify the source URI
  - Specify the URL rewriting rules (no-proxy, new window)
  - Specify the authentication settings, annotation rules etc.
- Use for a graphic step-based approach
  - Steps the user through the process of building a new portlet
  - Point and click
  - Preview window
- Intended for use by administrators/portlet developers
  - Specify access rights
  - Negotiate copyright
  - Access information (authentication, proxy/firewall settings etc.)
- The portlet created can then be added to a page group and administered as any other portlet.
Clipping Runtime Portlet

- Small UI used in portlet edit mode to specify authentication settings
- A set of Java classes providing a run-time implementation for the new portlet
- Obtains external content and presents the resulting content in the portlet's view mode
- URL rewriting:
  - Enabled - modify URL links to point to the Portal Server as opposed to the original host
  - Do Not Proxy... - URLs identified by the Rule expression are not altered and will not go through the Portal Server (requires Enable URL Rewriting)
  - Open URLs in new window... - indicates which URLs will not be altered and will be opened in a new browser window (takes precedence over other options)

Let us explore steps for adding a Web clipper portlet.

1. Go to Web Clipping Portlet and click Add as shown in the Figure 2-29.
2. You will be taken to the next window, Add a Clipper, as shown in Figure 2-30.
Figure 2-30  Add a Clipper

3. Complete the information requested in the window:
   a. specify the Name and default locale title. In this example, we have specified Handbook Clipper.
   b. Then, specify the Description. This is the description of the material that will be clipped.
   c. You can click **Set locale specific titles and descriptions.** A new window will open, in which you can select the language for the title and description. Click **Done** when finished and you will come back to the Add Clipper portlet, or click **Cancel** to return.
   d. Specify the URL to clip and the Connection time-out in seconds.
   e. Click **Modify clipping type** to manually select the HTML content to clip, or **Keep all content from the base URL**, or clip content between specified text. Click **Done** to make changes and exit or **Cancel** to return.
f. Click **Modify firewall options** to specify a firewall setting if needed to connect to the clipped page. Click **Done** to make changes and exit or **Cancel** to return.

g. Click **Modify authentication options**. If you need a user ID and password to access the content, specify them here. You have the option for choosing between HTTP- Basic Authentication or Form Based Authentication. Information need to be provided accordingly. Click **Done** to make changes and exit or **Cancel** to return.

h. Click **Modify rules for URL rewriting**. If the content of your Web clipper contains links, the URL's in those links will be modified to point to the portal server. However, you can create rules that exclude certain URLs from being modified in this manner. You can also specify URLs to be opened in new browser windows. Click **Done** to make changes and exit or **Cancel** to return.

i. Click **Modify security options**. If the clipped URL contains JavaScript and you want it to run, you have to deselect that option here. Click **Done** to make changes and exit or **Cancel** to return.

**Note:**
- Sometimes JSP pages take time to compile and wait for the page to load.
- In the example for this chapter, we have not used any of these Modify options.

4. Click **Next** and you should see a window as shown in Figure 2-31. By pointing and clicking the content, choose the content you would like to clip. Select the **Preview** option in the top-right hand corner to open or update the Web Clipping Preview window with a preview of the content for your Web clipper. When you are satisfied with the content of your Web clipper, select **Next**. If you want to make some changes, you can select **Clear**. **Cancel** will take you back to Add Clipper Portlet.
5. You will see the Content Preview window as shown in Figure 2-32.

**Note:** In this example, we have selected View documentation content to be clipped.
You should see Handbook Clippers being added to the list of Web Clippers in the Web Clipper Portlet, as shown in Figure 2-33.
Figure 2-33 Handbook clipper added to list of Web Clipperts

7. To test whether this Handbook Clipper Portlet works, add this portlet to the page where you had added the Read Parameter portlet and activate the portlet to the page. You should see the clipped portlet displayed on your page.

Note: You should see the Handbook Clipper Portlet on the available portlets list in the Manage Portlets portlet, with a status of Active.

2.2.5 Managing Web Services

Web Services are a form of distributed computing where a software component is available over the intranet. The functions provided by the Web service are described by the standard XML-based Web Services Definition Language (WSDL), so that the service can be invoked without prior knowledge of the platform, language, or implementation design of the Web service. A service broker maintains a Universal Discovery Description Integration (UDDI) registry as a directory of available Web Services. Service providers publish their Web
Services to a UDDI directory. There are many global registries available today that allow businesses to find each other across enterprise boundaries.

For a more in depth discussion of Web Services and its related technologies, see Chapter 4, “Web Services” on page 165.

Publishing and using a remote portlet in WebSphere Portal involves making the portlet available through a UDDI registry. Before you can work with the registry, you will need a user ID and password to access the registry.

There are several public registries available. Access the following URLs to learn about some of these registries.


**IBM Universal Business Registry:** [https://uddi.ibm.com/ubr/registry.html](https://uddi.ibm.com/ubr/registry.html)

**Microsoft:** [http://uddi.microsoft.com/default.aspx](http://uddi.microsoft.com/default.aspx)

**HP:** [http://hpmiddleware.com/SaISAPI.dll/SaServletEngine.class/products/hp_web_services/registry/default.jsp](http://hpmiddleware.com/SaISAPI.dll/SaServletEngine.class/products/hp_web_services/registry/default.jsp)

In this section, we will explore administering Portal Web Services using one of the public registries.

The Portal Administration -> Portlets -> Manage Web Services task is used to define UDDI registries to WebSphere Portal.

1. Select **Manage Web Services Portlet** and you should see a window as shown in the Figure 2-35.
   - Click the **Add** button to add UDDI registry information.
   - The **Edit** option will allow you to edit registry information.
   - Using the **Delete** option, you can delete the UDDI Web Service registry information.
2. If you click **Add**, you should see a window requesting registry information, as shown in Figure 2-35.
When defining the registry to WebSphere Portal, you need to know three things about the registry:

- Inquiry URL
- Publish URL
- Model Key

The corresponding values for the IBM UDDI Test Registry information that we have used in our example are:

- Display name for Registry: IBM UDDI Test Registry
- Registry inquiry URL: http://dwuddi.austin.ibm.com/uddisoap/inquiryapi
- Registry publish URL: http://dwuddi.austin.ibm.com/uddisoap/publishapi
- Model Key: UUID:UUID:30151A70-0D6F-4651-B578-1AE609AA147C

For a more detailed discussion of determining these values, see 4.4, “Configuring WebSphere Portal with the WebSphere UDDI Registry” on page 187.
3. Once you furnish the URL information and the key, click **OK** to register or **Cancel** to return.

4. If you click **OK**, you should see IBM UDDI Test Registry added to UDDI Web Services registries in the Manage Web Services portlet.

### 2.2.6 Web Services

One you have defined a UDDI registry to the WebSphere Portal as explained in 2.2.5, “Managing Web Services” on page 65, you can use the following tasks found under Portal Administration -> Portlets -> Web Services to access the information in a registry as shown in Figure 2-36. In this section, we will publish a portlet as a Web service.
1. Select the Provide registry authentication information task first. This option allows you to specify user ID and password information for using the registry. The user ID and password information is used by other tasks on this page when connecting to the registry.

2. The Manage Web Services businesses option gives you the ability to add or delete businesses from the registry.

3. To add a portlet to the registry and make it available as a Web Service, choose IBM UDDI Test Registry in the Manage Web Services Portlet and then select the Publish portlets option in the Web Services Portlet.

Tip: The Manage Web Services businesses and the Publish Portlets option automatically connect to the first registry in its registry selection list when the task is opened. To access a different registry, just select it from the drop-down list. The connection is automatically refreshed to the newly selected registry.
4. Once you select the Publish portlets option, a window as shown in Figure 2-37 opens. Any of the installed portlets (for which you have manage access) can be published as Web Services. Use the Get portlets link to populate the list of portlets to publish. Prior to publishing, you can click Edit name and description to modify the corresponding portlet information as it will appear in the registry. You can also choose to publish the portlet to an existing business (multiple portlets can be services under a single business) or you can create a new business.

5. Select IBM UDDI Test Registry for the UDDI Registry. In our example, we have given Publish to a new business, Sunil. You can use an existing business. Click the Get Portlets icon.

6. You should see a window open as shown in Figure 2-38. Select Show all Portlets and the list of Portlets that are currently installed will pop up in the window. Select the Portlet you want to publish as a Web service and click Add. In our example, we have selected Welcome Portlet. Highlight the selected portlet and click OK to proceed or Cancel to return.
7. You should see the Welcome Portlet in your Portlet to be published list. Click Publish.

8. If the portlet is published successfully, you should see the message Portlets Published Successfully as shown in Figure 2-39.
9. To remove a portlet from the registry, use the **Unpublish portlets** task on the Web Services task page.

10. The **Integrate a Web Service as a remote portlet** task is used to locate and bind a remote portlet to your local portal server. Once you define a UDDI registry to your portal, you can query the registry to determine what services exist. Any service using the RPWS service type can be accessed as a remote portlet.

11. Select **Integrate a Web Service** as a remote portlet option. You should see a window open as shown in Figure 2-40.
12. You have two options when searching for portlets in the selected registry:

- Use the **List all portlets** option if the number of published portlets in the registry is not large.

- If there is a large number of published portlets, you should use the **List portlets within business** option. Before you click the **Get businesses** link, you may want to refine the list of businesses returned by entering a string in the **Business name contains** field. From the list of businesses returned, select one from the list.

In our example, we will select **IBM UDDI Test Registry, Get Businesses Sunil**. Once you have finished entering your filter criteria, click **Get portlets** to retrieve the list of portlets. Select **Welcome Portlet** and then click **OK** to integrate a Web service as a remote portlet or **Cancel** to return.

13. If successful, you should see a message as shown in the Figure 2-41.
14. After a remote portlet has been integrated, you can access the portlet in all the same ways you would access a locally installed portlet, for instance, you can add it to pages, assign access control, choose skins, etc. The name of a remote portlet is prefixed with the string `IBM_Proxy`. In our example, you can find the remote portlet as `IBM_ProxyPortlet_Welcome Portlet`.

**Note:**
- You can also remove a remote portlet through the Uninstall option under `Portal Administration -> Portlets -> Manage Portlet Applications`.
- For more information on Web Services, see Chapter 4, “Web Services” on page 165.
2.3 Portal Settings

In the Portal Settings tab, you will be able to set up the basics for our portal, such as default language and messages to users; you will also be able to add new skins, manage the clients (browsers) that your portal will support and give them the priority you want. Also, you can manage the markups available on your portal or create new ones, and manage tracing.

The Portlet Settings Page contains the following portlets:

- Global Settings
- Themes and Skins
- Manage Clients
- Manage Markups
- Manage Search Index
- Enable Tracing

We will explore them individually.

2.3.1 Global Settings

When you select Portal Settings Portlet, the Global Settings portlet opens as the default.

The Global Settings portlet is used to:

- Specify default portal language.
- Determine unauthorized access by ignoring the user or providing an informative message.
- Provide information to returning users:
  - Taking them to the default page, or
  - Taking them to the page from the user’s last visit

When you select the Global Settings Portlet, you will see a window as shown in Figure 2-42.
Figure 2-42  Set Default Parameters for the Portal

- You can change the default portal language.
- Portlet View Permissions for unauthorized users can be set.
- When a user logs in, the first page that he sees can be defined.

15. Once you finish with the settings, you can click **Save** for the changes to take effect.

16. You need to restart WebSphere Application Server for the changes to take effect.

17. Click the **Reset** option and return to your default settings anytime when you are changing the parameters.
2.3.2 Themes and Skins

Themes and skins are templates that provide a page group's look and feel. They provide specific control for branding, navigation, and decoration.

Branding is the general scheme of the page. It usually encompasses logos, color schemes, decorations, fonts, artistic layout, etc.

Navigation refers to the way in which the user gets around on the site. There are several themes that demonstrate some of the different navigation models.

Decorations are the icons and images that are used to provide function and content links as well as a general look-and-feel enhancement.

Each place has a theme associated with it, and each theme has a set of skins associated with it.

Themes
A theme is an attribute of a page group, meaning you create page groups and then apply a theme to it. Themes are not user-specific. All users see the same theme that is applied to the page group. This means that a user could be presented with a completely different site experience when navigating from one page group to the next.

Theme: A theme determines the global appearance of all pages in a place. This will ensure visual consistency. Themes affect the navigational structure, the banner, colors and fonts and other visual elements of a page.

Note:
- In our example, we have used the default settings.
- If you choose the portlet is not displayed, replaced by an informative message option, then an informative message will appear when a user tries to access a portlet to which he or she does not have access.
- If you choose the Will always be the user's default page (portal session will not resume) option, users will always return to the default page after login.
- If you choose the option Will be the page the user most recently visited (portal session will resume), users will return to the page from their last visit. This option is helpful when users lose their portal session in the middle of a task and need to log in to return.
Themes contain various components:

- Cascading Style Sheets (CSS files) provide a mechanism to apply a look and feel to specific HTML tags. This can be done on a broad scale by specifying the attributes of the specific HTML tag. Or you can create classes and apply specific classes to the HTML attributes as desired. For example, you can specify a font size to be used on the `<P>` (paragraph) tag or you can create a class that specifies a font size, and then point to the class when you use the `<P>` tag. This second method provides the ability to apply different attributes to the same tag and achieve a variety of effects. CSS files can be found in the product install directory.

- Images provide specific brand, logos, and decorations. The image components of the theme's supported skins that are sensitive to theme settings are kept with the theme's images.

- Each theme contains its own set of JSPs to render the page groups and pages. This allows a completely different layout and brand experience from one page group to the next.

- Assets (images, JSPs, etc.) that are used in themes and skins are resolved by using WebSphere Portal supplied custom tags. There are several points within the directory structure where assets can be located. When the `<wps:urlFindInxxx>` tag is used, a search for the asset begins deep in the directory structure where the asset may be deployed for a specific country within a locale. If the assets is not found or the directory structure does not exist, the search continues by traversing “up” the directory tree. It's important to deploy default assets in the theme (or skin) root in order to avoid a “not found” situation.

The portal determines the theme for display as follows:

- If there is a theme associated with the displayed page group, the portal uses this theme.

- If there is no theme specified for the page group, the portal-wide default theme is used.

- If no portal default theme is set, the portal uses the theme settings given in the theme main directory, such as /theme/Tamil for HTML.

A default theme is not required for the portal.

Here is a search order example:

```
<!--background='wps:urlFindInTheme file="banner.jpg">'
 	hemes\html\science\ie5\en_US\default.jsp
 	hemes\html\science\ie5\en\default.jsp
 	hemes\html\science\ie5\default.jsp
 	hemes\html\science\en_US\default.jsp
```
An example directory structure where you can find your default themes in WebSphere Portal (\WebSphere\PortalServer\app\wps.ear\wps.war\themes) is shown in Figure 2-43.

Creating a new theme

To create a new theme:

1. Create a new directory for your theme:
   `<WP_HOME>/app/wps.ear/wps.war/themes/html/MyTheme`

2. Choose a current theme closest to the layout you want:
   `/themes/html/Science`
3. Copy the resources into the appropriate directories.
   - JSPs: Default.jsp, Banner.jsp, Navigation.jsp, etc.
   - Images: banner.jpg, navfade.jpg, etc.
   - Style Sheet: Styles.css
   
   **Note:** You may modify the tag definitions and the class definitions.

4. Customize to get the look and feel you are looking for.

**Skins**
Skins are used to apply specific decorations to portlets. They are used in conjunction with the theme in order to accomplish this. For instance, the theme's Cascading Style Sheet is used to specify the color of the portlet's title bar. Some skins use images to produce rounded corners on the title bar. The rounded corner images are stored with the different themes that support the skin. This is done so that the colors match across all of the components of the portlet's title bar. The rest of the skin assets are generic and apply to all theme uses, so they are kept in the skins folder.

Skins contain images that are used to create the visual effects of the portlet. The visual portlet container (lines, shadows, backgrounds, etc.) and the portlet navigation icons (edit, help, back, etc.) are the main components of a skin.

Skins are applied to the portlet via a JSP known as Control.jsp. Each skin has its own version of Control.jsp. It is used to specify the exact implementation of the skin and can be considered the Portlet container.

The search for skin assets works the same way as the themes search. Using the `<wps:urlFindInSkin>` tag, the file system is traversed starting with a specific country within a locale and working up to the skin default.

**Skin:** A skin defines the frame around a portlet, thus determining the look of the portlet. It affects only portlets. You can select a skin for each portlet in a page if the theme has skins associated with it.

A default theme is not required for the portal, but specifying a default skin is mandatory.

The portal determines the skin for display as follows:

- If there is a skin specified for the portlet, the portal displays the component in that skin.
- If there is no skin specified for the component, the portal looks for a skin on the page level and uses it.
If no skin has been set for the page, the portal checks the page group for a skin setting.

If the page group has no skin specified, the portal uses the default skin of the page group.

If no skin has been found so far, the portal default skin is used.

While a default theme is not required for the portal, specification of a default skin is mandatory.

To create a new skin, make a copy of one of the existing ones and modify the images and the JSP in order to get the desired look and feel. Once you finish, you will be able to choose it from the administration portlets.

Creating a new skin
To create a new skin, execute the following steps:

2. Choose a current skin closest to the layout you want (/skins/html/Science).
3. Copy the resources into the appropriate directories:
   - JSPs: Control.jsp, RowContainer.jsp, ColumnContainer.jsp, ...
   - Images: title_edit.gif, ...
4. Customize to get the look and feel you are looking for.
   - Control.jsp is the only JSP that you would want to modify.
   - Images may be modified or new ones created.

Administer Themes and Skins Portlet
To administer Themes and Skins, perform the following steps:

1. Select the Portal Settings and Themes and Skins portlet. You should see Manage themes and skins as shown in Figure 2-44.
2. In the Manage Themes and Skins portlet, you will notice that for our example we have WebSphere as portal default theme and Outline as portal default skin.

3. Themes have four administrative capabilities:
   - Add New theme
   - Edit Theme
   - Delete Theme
   - Set as Default

**Add New Theme**

Let us add the theme we created using this administrative functionality.

1. Click the **Add New theme** option.
2. You will see a window open as shown in Figure 2-45.
3. Enter the name for the theme (default locale title). In our example, we have entered *Testing Theme*.

4. Enter the directory location of your theme. You can specify just the relative path.

   **Note:** We will use *MyTheme* which we created for adding to the portlet.

5. You will have all skins to your left-hand side and you can use the arrow button and choose the skin that you want for the theme.

   **Note:** If only one skin is chosen, it is selected as the default. However, you can choose multiple skins and click *Set as Default* for making a skin the default skin.

6. You can confirm with the message at the bottom on your default skin. In our example, we have chose *NoSkin* as the default skin for our theme.
7. You can change the language and the Theme title (locale-specific theme titles) by selecting **Set locale specific-titles** option. For our example, we will not use this option.

8. Once finished, click **OK** for adding the new theme or **Cancel** to return.

9. You will see Testing Theme being added to the list of available portlet themes.

### Edit Theme

The Edit Theme option will help you modify which skin your theme uses.

1. Select the Theme for which you need to modify the skin.
2. Select the **Edit theme** option.
3. Make the necessary changes. You can also edit local specific titles here.
4. Click **OK** to confirm the changes or **Cancel** to return.

### Delete Theme

1. Select the theme you want to delete and click the **Delete** option.
2. A pop-up window will ask you to confirm your deletion.
3. Select **OK** to confirm or **Cancel** to return.

**Tip:** The files that compose the theme are not deleted from the system.

### Set as default portal theme

Perform the following instructions to set a default portal theme:

1. To set a portal-wide default theme, select a theme from the themes list, then click **Set as default portal theme**.
2. If no theme is set for a place, the portal default theme is used.

**Tip:** You should not apply the Admin theme to the portal. This theme is intended for administrative portlets and renders the portlets without a title bar.

### Add New Skin

You can add a new skin using the Add New Skin option.

1. Select **Add New Skin**; we will add NewSkin, which we created, to the list of available skins.
2. You will see the window shown in Figure 2-46.
3. Specify the skin name (NewSkin), default locale and the directory location where this skin is stored. You can specify just the relative path.

4. Set the Locale-specific Titles option will help you change the locale-specific titles.

5. Click OK to add the new skin or Cancel to return.

6. You should now see NewSkin added to list of available skins.

**Important:** The necessary skin files must already exist within a specific directory path before a theme can be added.

**Delete Skin**
Perform the following instructions to delete a skin:

1. Select the skin you want to delete.

2. A hint window will pop up, asking you to confirm the deletion. Click OK if you are sure or Cancel to return.
Set as default portal skin
This option will help you to set a portal-wide default skin for portlets.
1. Select a skin from the skins list.
2. Click **Set as default portal skin**. If no default skin is set for a theme, the portal default skin is used.
3. The changes will be reflected when the page refreshes.

**Important**: You should not apply the skin with the name NoSkin to a portlet. This skin is intended for administrative portlets and renders the portlet without a title bar.

2.3.3 Manage Clients
Portlets can be accessed through a Web browser, mobile devices, personal digital assistants, etc. The Manage Client portlet will help you to define these devices for accessing portal information. To optimize the data that the portal sends to the client and to handle the limitations and deviations of each individual client browser, the portal server maintains information about all supported client devices in a client registry.

You can do the following tasks from the Manage Client Portlet.

**Add new Client**
You can add a new client for accessing portal information.

1. Select the **Portal Setting** and **Manage Client Portlet**. You should see a window as shown in Figure 2-47.
1. Click Add a new client.

- User Agent is a required field. Make sure that the user agent string that the client sends in its request header matches the value you specify. If you are not sure, enter * and the portal server will search for the closest match to this string.

- Select Markup, which the client supports. This is a required field.

- Specify Markup Version, Manufacturer information, Model and Version. These are optional values.

- List the capabilities for this client you have specified. For example, you could specify specific attributes it supports in HTML, such as JavaScript, etc. You can use Add or Delete options for adding and removing the capabilities.

- Specify the position where you would want this new client information stored in the client registry. The portal server matches the user agent string in the client’s request header to patterns in the client registry. If the default user agent pattern (*), is used then it should be placed in the last
position. Select the drop-down option and you can position the client information according to your requirement.

**Note:** If the user agent sends Microsoft Internet Explorer 5.5 and the portal server finds Internet Explorer 5*, then that registry entry is used to determine the markup sent to the client. For this reason, it is recommended that you place the most specific User Agent patterns at the top of the list.

3. Click **OK** to add the new client or **Cancel** to return.
4. You will see the new added client in the clients list.

**Edit selected client**
If you notice that a certain browser requires different, browser-specific processing, you can change the client registry information by editing the existing client registry entry.

1. Select the client which require editing.
2. Click **Edit selected client**.
3. Make the required modifications. Click **OK** to approve changes or **Cancel** to return.

**Order client before/Order client after**
To send the most exact markup to the client, it is very important that you make sure your client is properly positioned in the client registry.

1. Select the client from the list on the Manage Clients tab.
2. Click **Order client before** to move the client up in the registry.
3. Click **Order client after** to move the client down in the registry.

**Show Info**
Perform the following instructions to show information:

1. Click **Select Info** and the complete client information for all the clients defined to WebSphere Portal is displayed.
2. Click **OK** to return to the client listing.
Delete selected client
Perform the following instructions to delete a selected client:

1. Select the client you want to delete from the client listing and click **Delete selected client**.
2. A pop-up window will ask for confirmation. Click **OK** to delete or **Cancel** to return.

2.3.4 Manage Markups

The Manage Markups portlet will help you define the markup language that will be supported by the portal. By default, WebSphere Portal comes with three markups, chtml, wml and html as shown in Figure 2-48.

Figure 2-48  Manage Markup portlets
Edit Selected Markup
You can edit the markups.
1. Select the markup that you wish to edit and click Edit Selected Markup.
2. Make the necessary changes. You can select Set locale-specific settings and change the markup settings for different languages.
3. Click OK to confirm changes or Cancel to return.

Add new markup
To add a new markup, click Add new markup.

You will be required to specify the markup name. This is a required field.

The directories of this name also have to be created to support the aggregation of the portal for clients that support this markup. For example, to add the markup MathML, the following directories have to be created:

- `<wps_root>/app/wps.ear/wps.war/MathML`
- `<wps_root>/app/wps.ear/wps.war/windows/MathML`
- `<wps_root>/app/wps.ear/wps.war/themes/MathML`
- `<wps_root>/app/wps.ear/wps.war/skins/MathML`

For this reason, avoid characters in the markup name that might cause conflicts inside file or path names, such as /, \, , or &. The markup name also acts as the default title for those languages where no locale-specific title has been set.
1. Specify the MIME type associated with this markup.
   UTF-8 is used as the default character set if the Default character option is left blank.
2. You can use Set locale-specific settings for specifying/changing markup settings for different languages. The default markup title is displayed; you can change this. Click OK to confirm and Cancel to return.
3. Click OK to add the new markup or Cancel to return. Once you click OK, you should see the new markup with the available markup for WebSphere Portal.

Activate/Deactivate Selected Markup
By default, all available markups to WebSphere Portal are in Active state.
1. Select the markup you want to deactivate and click Activate/Deactivate.
2. The page will refresh and you can see the message Inactive next to the selected markup. You will also see a message confirming that the selected markup is inactive.
You cannot render the markup to any portlet until you change the status of the markup to Active.

3. Click **Activate/Deactivate** option again to activate the markup.

**Show info**
Perform the following instructions to show information:

1. Select **Markup Info** to get complete details about all the available markups to WebSphere Portal along with the date created and last modified information.
2. Click **OK** to return to the Manage Markups portlet.

**Delete selected markup**
You can delete any markup by clicking the **Delete** option.

1. A pop-up window will appear asking you to confirm the deletion.
2. Click **OK** to delete or **Cancel** to return.
3. If you choose **Delete**, the markup will not be available in the list of available markups for WebSphere Portal.

### 2.3.5 Manage Search Index
WebSphere Portal provides integrated text search capabilities, including a search portlet, a crawler, and a document indexer.

- **Document search** -- a search engine used with HTML or text documents. In this case, a portal administrator uses the Manage Search Index portlet to create an index to be used for the search. Once the index is created, the administrator configures an instance of the Document Search portlet to use that index.

- **Enterprise Information Portal Search** -- provides both a federated or advanced search to a local or remote EIP server. A federated search performs a function similar to the EIP thin client application. The user enters parameters on one of possibly many predefined templates, and the results are presented in a table. The advanced search is an unstructured search using EIP's Information Mining feature. The user can select from a list of categories to narrow the search results.

- **Lotus Discovery Server** -- creates expertise and knowledge maps by analyzing and categorizing documents. It creates profiles of users based on their document activity, including their topics of interest and their area of expertise. The Lotus Discovery Server is a separately purchased product.

The search service can search the portal’s document repository as well as Internet content. The portal server’s built-in search engine is optimized for
full-text searching of small and medium-sized collections where precision is essential. Searchable resources can be stored on the local portal server or on remote sites.

The search engine supports free-text queries, with query assistance and query word completion. Search queries use advanced query operators (+ or -) to indicate keywords that must be in the document or keywords that must not be in the document. The search engine can search documents in any language, and also supports synonyms and stop word lists. Search results include document summarization and search results clustering.

To prepare for searching, the search engine builds a full-text index in order to search documents that are stored in the local file system. The indexer supports multi-word indexing for high precision. The index can be compressed, and the size can be controlled for situations where the size of the index needs to be limited.

Select the **Portal Administration -> Portal Settings and Manage Search Index**portlet, and you should see a window as shown in Figure 2-49. It has two options, Configure search index and Manage search index.
Configure Search Index

The Configure Search Index option manages the configuration information for one or more search indices. It does not actually create or modify an index, just the configuration information for that index.

1. Select **Configure Search Index** and you should see a window opened as shown in Figure 2-50.
2. Specify the index that the user will select in the search portlet. More than one index can be created. The location you specify for the search index becomes the name of the search index. Use this index name when you configure the search portlet.

3. Specify the location of the index. This is the location where index files are located on your system.

   **Important:** Multiple indexes may not share a common location.

4. Choose the option **New index**.
   - The radio button selections New Index and Update Index do not modify the index, rather they indicate whether the configuration information you are specifying is for a new or existing index.
   - To actually affect the index, you must use the Manage Search Index task on the main page of the Manage Search Index portlet.
The Delete Index option is used for deleting the index. Select the index you need to delete and select the radio button option **Delete**. However, you need to ensure that the index file is physically removed from the system.

5. If you enable **CJK language support**, it enables Chinese, Japanese, and Korean languages. Select this option to allow these languages to be indexed.

6. Select the **Document types to be indexed** (HTML or plain text).

7. Specify the URL which the crawler will use as a starting point to traverse through the hyper-linked documents, beginning with the protocol.
   - A single index can search multiple sites. See the instructions in the WebSphere Portal InfoCenter for modifying the crawler.properties file.
   - A proxy server can be used to access the site you want to index. See the instructions in the WebSphere Portal InfoCenter for modifying the crawler.properties file.
   - The Levels of linked documents to index option determines the number of links to follow within the URL for fetching content for the index. This option can be used to control the number of documents returned and processed for sites that are of unknown size.
   - The Number of linked documents to index option will determine the total number of documents to be indexed.

8. Once you specify index configuration information, click **OK** to create a new index and **Done** to return. If you click **OK**, a new index will be created.

9. To change configuration information for an existing index, select the index from the drop-down Index option. Make the necessary modifications and click **OK** to confirm the changes.

**Manage Search Index**

The Manage Search Index task is used to initiate creation of a search index.

1. Select the **Manage Search Index** option from the Manage Search Index portlet and you should see a window as shown in Figure 2-51.
2. Select the index you wish to create from the drop-down selection list.

3. Click **Begin index update** to initiate creation of the index. The crawler used to construct the search index processes asynchronously.

   - The Last updated completed at field is updated with date and time information when a new index has been created or an existing index refreshed.

   - Number of active documents indicates the number of documents retrieved and indexed.

   **Note:** Depending on the index size, this process can take several minutes to complete. You can click the browser refresh button to see if the update has completed, or continue elsewhere in the portal and return to this task later on to see if the update has completed.

4. Click **Done** to return.
Enable Tracing
The Enable Tracing portlet lists all WebSphere Portal Trace Loggers and allows you to enable or disable each Trace Logger listed.

1. Click the check box (Trace On) next to a Trace Logger name to enable that Logger.
2. Select the Save option at the top or bottom of the window to save the Logger settings.

Tip: Use Enable Tracing portlet to set logging options for the current portal session only.

Note: You can refer to Chapter 3., “Search capabilities” in the IBM Redbook, IBM WebSphere Portal V4.1 Handbook Volume 3, SG24-6921 for additional information.

Figure 2-52 Enable Tracing portlet
You can find portal trace information in the `WebSphere\PortalServer\logs` directory.

To disable the trace for a particular logger, uncheck that option.

## 2.4 Users and Groups

Centralized administration of users and user groups is provided through the Member Services component of Portal Server. Users can register and manage their own account information, or an administrator can provision and manage users. Administrators can also create groups and maintain group membership. A user group (sometimes called “member group”) is an arbitrary collection of members. The members of the group are users or other groups.

The Member Services component:

- Manages user profile information (which excludes authentication data).
- Manages user group information.
- Provides a user repository containing user profiles, group definitions, and organizational entities. The physical implementation of the registry can be a database or a directory server.

### Users and Groups

The main goal of Users and Groups is to help the administrator create portal users and groups without using the LDAP interface directly.

### 2.4.1 Manage Users

Users can be generic or registered users. Generic users are basically anonymous users; registered users have an associated user profile and a user ID and password kept in the authentication registry. Member Services accesses the user authentication registry for updating user ID and password information.

The user registration page shipped with WebSphere Portal exposes a limited set of user attributes. You can add or delete attributes as required for your portal implementation, either by exposing additional attributes from the underlying user repository (LDAP) that are not currently exposed, or by extending the user profile to include new attributes.

- Portal will recognize an existing user/group in existing repository (LDAP).
- A user can modify his profile (except the user ID).
- Users may belong to multiple groups.
User membership can also be managed externally by using the LDAP Directory Management Tool (DMT).

Typically, a portal operator will separate its users in groups. Separating smaller groups from bigger groups will enable sophisticated structuring of the users in the system.

**Note:** Using an LDAP directory as a user database, grouping users will not lead to branches in the LDAP directory. By default, all users go to the cn=users branch and all groups to the cn=groups branch. The groups will hold the information of the users via the uniqueUsers field. See also 5.2.4, “Secureway LDAP” in the IBM Redbook *IBM WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883 for information on setting up the LDAP structure during install time.

Using the Manage Users portlet, you can do the following:

- Create a new user (the window has the same look as in self registration)
- Edit a user profile
- Delete a user
- Show ID
- Search for users

**Create new user**

By default, there are three different ways to create a user:

- Add the user directly to the appropriate branch in the LDAP directory. To do this, use the Administration Interface of your LDAP directory or create an appropriate LDIF file and import it to your LDAP directory. See “Adding entries to the LDAP directory” on page 606 in the IBM Redbook *IBM WebSphere Portal V4.1 Handbook Volume 1*, SG24-6883 as an example of how it is done in the WebSphere Portal installation on the AIX platform.

- Use the Self-Registration functionality of WebSphere Portal. To do this, use the **Sign Up** button located at the upper right corner in the portal default startup page (see Figure 2-53). Fill out all required values in the window and submit them. The values definitions are the same as explained in the third option below.

- Go to Users and Groups portlet and select **Manage Users** portlet by clicking the left tab of the lower tabbed buttons.
  - Click **Create new user** option. See Figure 2-54.
  - Provide the required values as shown in Figure 2-55.
Figure 2-53 Use the Sign Up button for self registration
Figure 2-54  Use the Administration Portlet for creating a new user
Figure 2-55  Provide new user information

i. Specify User ID information. The (*) represents values that are required for registering the new user.

**Note:** The user ID must be 3-60 characters in length. It can contain alphanumeric characters and the hyphen "-", period ".", and underscore "_" characters. No other characters are permitted in this field.

ii. Provide password information for the user and confirm the password.

iii. Type the first and last name for the user.

iv. Specify the e-mail ID.

v. You can select the preferred language for the user from the drop-down menu list. If no language is selected, Portal will assume that the user will use the default language.
vi. Select any particular interests the user might have from the drop-down menu list.

vii. Click **OK** to register the new user or **Cancel** to return.

viii. Test: Open your LDAP and you should be able to see this new user listed.

Either of these ways create a valid new user that is part of no group and has therefore the minimum default set of permissions.

**Search for users**
You must search for users prior to editing or deleting users.

On the left-hand side of the Manage users portlet, you will see the option Search for users.

1. Type the text for search in the *Name is* field. You can use the asterisk character, *, as a wildcard.

2. You can type a group name in the Restrict to Group option if you want to restrict the search to a particular group. If the field is left blank, all groups are searched.

3. Click **Get users**. The page will refresh and the list of users will be populated and displayed.

The search list is based on name and not on ID.

**Edit user profile**
Using this option, you can edit user information.

1. Select the user for which you need to edit the information and click **Edit user profile**. If no user is selected and you click this option, a pop-up window will appear asking you to select the user.

2. User ID information cannot be changed. You will see a window as shown in Figure 2-56.

3. The values are already populated with the current values. Make the necessary changes.

4. Click **OK** when done or **Cancel** to return to the Manage users portlet.
Figure 2-56   Edit user information

Delete user from Portal
This option will help you remove a user information from the database.

1. Select a user from the list and click the **Delete user from portal** button.
2. A pop-up window will appear requesting you to confirm the deletion. Click **OK** to proceed or **Cancel** to return.
3. The user information will be deleted from the database if you click **OK**.

Show ID
This shows you the ID of a highlighted user.

1. Search for the user you want to view.
2. Highlight the user's name in **Users**.
3. Click **Show ID** to display a box with the user's ID information.
4. Click **OK** when the ID is shown in the pop-up window.
2.4.2 Manage User Groups

With Manage User Groups, you no longer have to go to LDAP and you can manage groups and group memberships from within the portal. Groups can contain groups and they can be managed in the same administration window. Even though you can manage users and groups from within the portal, you can import an LDIF file or manage users and groups by using the LDAP Directory Management Tools.

- Access permission for resources can be given to both groups and users. If a user is added to a group, it will inherit the groups permission. This means that a user has in no case fewer permissions than his group. If a user is a member of more than one group, the user inherits the highest permission for each particular resource.

- This is also true for groups. Groups will also inherit the permissions of the groups to which they are added.

**Note:** You will not see the inherited permissions of a group in the Access Control List Administration Portlet in WebSphere Portal Version 4.1.2, but if you add a user to this group, the user will show the inherited permissions.

The Manage User Groups portlet provides you with tools to help you search for and assign users to your groups. The following options are available with this portlet:

- Create Group
- Membership
- Delete group
- Show ID
- Search for groups

**Create Group**

By default, there are two different possibilities for creating a group:

- Add the group directly to the appropriate branch in the LDAP directory. To do this, use the Administration Interface of your LDAP directory or create an appropriate LDIF file and import it to your LDAP directory.

- Use the WebSphere Portal User and Groups Administration Portlet. To do this, use the following instructions:

  a. Go to Users and Groups page and select Manage User Groups by clicking the right tab of the lower tabbed buttons.
b. Fill the field Group Name with the name of the group that you intend to create.

**Note:** The group name must be less than 256 alphanumeric characters. It can contain alphanumeric characters and the hyphen, -, period, ., and underscore, _ characters. No other characters are permitted in this field.

c. Click the plus symbol or **Create new group** button on the right side of the window.

d. You will see a message stating that the group was successfully created, as shown in Figure 2-57.

e. Test: open your LDAP and you should be able to see this new group added.

Either of these ways create a valid new group.
Membership: assigning a user to a group

**Important:** When the users register themselves, the administrator has to assign them to a group. If you do not do this, you will not see the users in the Manage Users tab. Go to Membership and assign them to groups there.

By default, there are two different approaches to assigning users to groups.

- Use your favorite LDAP tool to directly modify the group entry in the LDAP database. Add to the uniqueMember section the Distinguished Name (DN) of the user you intend to have in this group.

- Use the WebSphere Portal User and Groups Administration Portlet. To do this, use the following instructions:

1. Search for the user group to which you want to add members.
   a. Highlight the group’s name in User Groups under the Manage Users portlet.
   b. Click **Membership**. A window will appear as shown in Figure 2-58.
   c. Add the user that you wish to add to the selected group to the Search Results field. Add an appropriate search value to the **Name is** field. You can also use a wild card search (*). To further qualify your search, you can add a value to the Restrict to Group field.

   **Note:** For the Restrict to Group field, you can also use a wild card search using an asterisk (*). An asterisk on its own will, however, restrict you to users that are members of any group. This means that it will not list the users that are not members of any group yet.

   d. Click the image icon **Go** to submit your search request and display the result in the Search Results list.

   e. Select the appropriate user from the search results list and click the **Add to group** button.

   f. The page will refresh and you can see the user listed under the group you chose earlier.

   g. Click the **OK** button to make the changes persistent and to switch back to the previous window or add more users to this group. Click **Cancel** to return.
Assigning a group to a group

By default, there are two different approaches to adding groups as sub-groups to other groups:

- Use your favorite LDAP tool to directly modify the group entry in the LDAP database. Add to the uniqueMember section the Distinguished Name (DN) of the group you intend to have as sub-group in this group.

- Use the WebSphere Portal User and Groups Administration Portlet. To do this, use the following instructions:
  a. Search for the user group to which you want to add members.
  b. Highlight the group's name in User Groups under the Manage Users portlet.
  c. Click Membership. A window will appear as shown in Figure 2-58.
  d. To add the group to the Search Results field, click the Add groups to group radio button and insert an appropriate search value to the Name is field. You can use * as a wildcard to enhance your search. If you want to
restrict the search for a user to a particular group, type a group name. If the field is left blank, all groups are searched.

e. Click the image icon Go to submit your search request and display the result in the Search Results list.

f. Select the appropriate group from the Search Result list and click Add to group image icon as shown in Figure 2-59.

g. Click the OK button to make the changes persistent and to switch back to the previous window or add more groups to this group. Click Cancel to return.

Figure 2-59 Adding groups to a group

Show ID
This shows you the ID of a highlighted group.

1. Search for the user group you want to view.
2. Highlight the user group in User Groups.
3. Click Show ID to display a box with the group's ID information.
4. Click **OK** when the ID is shown on the pop-up window.

### 2.5 Security

The Security portlet will help you to assign permissions for groups, portlets, places, pages, Web modules for a group or user and will also help you select a security vault management task.

WebSphere Portal Security Administration comes with two portlets:

- Access Control List
- Credential Vault

#### 2.5.1 Access Control List

Access control is also referred to as authorization. Before a user can be authorized to access a resource, he must be authenticated, that is, he must have successfully logged into the portal. Other than the requirement for a successful authentication, authorization is independent of the WebSphere Application Server (or any custom authentication proxy being used by WebSphere). WebSphere Application Server protects servlets and EJBs; Portal protects all portal resources:

- Portlet application
- Portlet
- Page
- Place
- User
- User group
- Resource collection
- Portal
- External ACL

Portal has five types of permissions: **view**, **edit**, **manage**, **create**, and **delegate**. The meaning of each permission type with respect to the resource can vary based on what is meaningful for the type of resource. For instance, view access for a portlet, page, and place allows the user to “see” the portlet, page, and place in the appropriate context within the portal. For other resource types, such as user and user group, view access is not applicable.

1. Select **Security Portlet** from the Portal Administration page and you should see a window as shown in Figure 2-60.
The Access Control List portlet allows you to configure access to portal resources by granting permissions to users and groups. The Access Control List portlet also passes control of resources to and from external security mechanisms if desired.

Before a portal can be deployed, the access control associated with portal resources must be established. When setting up access control, it is important to remember that permission to a “higher level” resource does not imply the same permission to a “lower level” resource. For example, access permission to a page does not automatically grant access to the portlets contained on that page.

Giving a user or a group of users Manage authority on the object “Portal” effectively makes them administrators.

The Access Control List portlet allows you to configure access to pages, places, portlets, etc. by granting permissions to users and groups.
2.5.2 Credential Vault

The Vault Service stores user IDs and passwords for various back-end resources. It is a portal service to assist portlets and portal users in managing multiple identities, perhaps among multiple Vault implementations.

A vault segment can contain one or more credential slots, which are containers where portlets store and retrieve a user's credentials. A credential slot contains only one credential and is linked to a resource in a vault implementation, the place where the credential secrets are actually stored.

A vault segment can be either administrator-managed or user-managed. Admin-managed segments could be corporate resources such as Lotus Notes databases or intranet passwords. WebSphere Portal provides one simple database vault implementation for user or admin-generated mappings to user IDs and passwords for other enterprise applications. It can also contain user-generated mappings to user IDs and passwords for non-enterprise applications (for example, your Hotmail account in a POP3 portlet).

The Credential Vault provided by WebSphere Portal distinguishes between three different types of credential slots:

- A system credential slot stores system credentials where the actual secret is shared among all users and portlets.
- A shared credential slot stores user credentials that are shared among the user's portlets.
- A portlet private slot stores user credentials that are not shared among portlets.

Many portlets need to access remote applications that require some form of user authentication. In order to provide a single sign-on experience, portlets must be able to store and retrieve user credentials for the back-end application they need to access. For accessing applications outside the portal's security realm, the Portal Server provides a Credential Vault service that portlets can use to store user ID and password (or other credentials) information.

Note: Portlets can (on behalf of a portal user) set and retrieve credentials in both types of segments. However, they can only create credential slots in user-managed segments.

Note: For information on using Access Control List portlet and more information on Portal security, refer to Chapter 4. of the IBM Redbook IBM WebSphere Portal V4.1 Handbook Volume 3, SG24-6921.
The Credential Vault supports the WebSphere Portal default database vault or IBM Tivoli Access Manager. You can also use other Credential Vault implementations. If you provide your own vault implementation, you will need to write a vault adapter.

A portal administrator uses the Portal Administration -> Security -> Credential Vault task to manage vault segments and slots as shown in Figure 2-61.

**Figure 2-61  Credential Vault portlet**

**Note:** For information on using Credential Vault portlet and more information on Portal Security, refer to Chapter 4. of the IBM Redbook *IBM WebSphere Portal V4.1 Handbook Volume 3*, SG24-6921.

### 2.6 Portal Content

WebSphere Portal includes a Content Organizer portlet that enables portal users to contribute and share documents. The content organizer portlet provides a workspace for storing, navigation, viewing, and searching portal documents and
other content. The organizer is preconfigured to work with files and Rich Site Summary (RSS) formats. Additional content types, formats, and back-end systems can be integrated easily.

The content organizer maintains properties and attributes of content, which can be searched by the portal’s built-in search service.

The content model used by the organizer is a lightweight, open, extensible content model based upon the WebSphere Personalization resource engine. Third-party Web content management systems can also be integrated.

The purpose of this portlet is to provide a basic portal (enable version) with small content management capabilities.

We will highlight the Portal Content Organizer capabilities here. It is possible to upload content from both a file system and a Web site into Portal Content Organizer resource collections, and annotate and bookmark the content.

First of all, you will need to have a users and groups tree to assign different permissions to folders. If you want to create one to understand its capabilities, do the following:

1. Log in as wpsadmin and open the Portal Administration page group.
2. Create groups called xxxadmins and xxxusers.
3. Make the wpsadmins group a member of the xxxadmins group.
4. Add your non-administrative groups as members of the xxxusers group.

2.6.1 Manage Content Organizer

Figure 2-62 illustrates the Manage Content Organizer portlet in WebSphere Portal.
It is also possible to change the content format in Portal Content Organizer. For example:

1. To open the Portal Content Organizer, click Portal Content -> Manage Content Organizer.

2. Choose Manage content formats.

3. Choose Document, and then click Modify selected content format and you will see a window as shown in Figure 2-63.
4. In the Show in details field, check the date box and save your change.

5. You will see a message: Content format modified successfully.

6. Click Done.

**Define content categories**

This specifies the category label to be used when assigning content to categories. The Administrator assigns labels to the categories.

1. Open the Portal Content -> Manage Content Organizer page.
2. Choose Define content categories.
3. Add categories.
4. Click OK to add the categories and Cancel to return to the Manage content organizer portlet.
5. You can perform any edits for the previous documents using this option.
Figure 2-64 illustrates the Manage Content Organizer portlet to define content categories.

![Image of Manage Content Organizer portlet]

**Register resource collections**

This shows registered and unregistered collections. Collections correspond to content formats. After content is added, it is considered unregistered. You must register the collection for users to access the collection.

1. Select **Register the resource collections**.
2. Register all of the resource collections. Figure 2-65 shows the window to register resource collections.
3. Select the resource collection and click **Register option**.

4. Once you register, click **Done** to finish and return.

**Content repository status**

Content repository status provides you with the information on the number of documents that are available, the number of deleted documents and also the date information for the last content that was collected.

**Update content repository**

This updates the content repository. The number of documents listed in the content repository status box will update when you select this option. You will see a message: **Content repository updated successfully.**

**Purge deleted content**

This deletes the documents marked for deletion in the database. The number of deleted files will update in the content repository status box.
2.6.2 Content Organizer

You can add the Content Organizer portlet to a page by opening the Work with Pages page group.

1. On the Content Organizer portlet, click the **Upload content** tab and choose **Browse**.
2. Browse any folder containing documents.
3. Choose files and make some (but not all) of the documents shared and then click **Upload** as illustrated in Figure 2-66.

![Figure 2-66   Upload content](image)

Viewing and annotating content can be done as follows:

1. Click the **Explorer** tab and open the shared and private folders.
2. Click the radio button next to a PDF file, and then choose **View**.
3. Choose **Click here to download** and save the file to your local disk using a file name that you will remember.
4. When the download is complete, click **Open**.
5. Review the file, and then choose **File -> Exit**.

6. Click **Close** to close the Explorer view.

7. If you have Microsoft Word Viewer or Microsoft Word on your machine, use the same procedure to open the Word file.

8. To add annotations to that Word file, choose **Annotate**.

9. Add a Document name, Description, Type, Subject, Title, and Author.

10. Click the **Auditing** category, and then click **OK**.

The annotations could be used for categorizing your content. Figure 2-67 shows a content file list.

![Figure 2-67  Content file list](image)

Figure 2-68 illustrates the annotation window.
To bookmark content, bookmark files by selecting the file and clicking Add to bookmarks. You can also click the My bookmarks tab to review your bookmarks.

**Note:** If you log off and then log in as a different user, you will see that the PCO changes were made only for this user.

How to use Web content to create a new document

1. Copy some files in an area that will be served by the Web server, for example the IBM HTTP Server htdocs directory.
2. On the Content Organizer portlet, click Create new document.
3. Complete the Document name, Description, Type, Subject, Title, and Author fields.
4. Choose a category and in the URI list for this document field, type http://<your_server_name>/foldername and then click Add URI.
5. Click **OK**.

6. Select the radio button to the left of your new entry in the Explorer.

7. Click **View**.

8. Select `http://<your_server_name>/foldername` and then click **View**.

9. Choose **Click here to download**. Portal Content Organizer launches a browser window showing the folder name home page.

10. Bookmark the home page by selecting the radio button to the left of the entry in the Explorer and then clicking **Add to bookmarks**.

Figure 2-69 shows the window to create a new document.

![Create a new document](image)

**Figure 2-69    Create a new document**

**Note:** For additional information on Web content management and Portal Content Organizer refer to Chapter 1. of the IBM Redbook *IBM WebSphere Portal V4.1 Handbook Volume 3*, SG24-6921.
WebSphere Portal customization

This chapter describes the general customization concepts, themes and skins and customization portlets that come with the WebSphere Portal Work with Pages page.

Note: Most of the concepts covered in this chapter are an extension to Portal Administration. You can refer to Chapter 2, “WebSphere Portal administration” on page 25, for basic definitions that will be used in this chapter (such as page, page group, portlet, portal, etc.) and for additional reading material.
3.1 General customization

**Definitions:**

- **Customization** is presenting tailored content and layouts based on explicit user specifications.
- **Personalization** is presenting content to a user based on their profile.

Users can have one or more personalized pages, navigating to each one from the home page. Pages are arranged into page groups or places. Each page group can have its own choice of color themes, skins and page layouts. Themes are used to define the fonts, colors, spacing, and other visual elements; themes consist of cascading style sheets, JSP files, and images. Skins are decorations and controls placed around portlets, such as title bars, borders, shadows, etc. Since the look and feel of each page group can be completely different, page groups can be used to create multiple virtual portals running on one portal server.

In a page group, each personalized page can have a different set of portlets. The portlets on a page can be selected by end users or by administrators, depending on their access rights for the page. Administrators can specify that certain portlets be required, so that end users cannot remove them or rearrange them. Pages can also be rearranged to achieve a different navigation order.

All of the functions related to customizing page layouts, page contents, color themes and skins are found in the pages of the Work with Pages page group. Using these tools, users can see the arrangement of the page and can move portlets around easily.

### 3.1.1 Customization roles

Customization is one of WebSphere Portal main strengths. Portal administrators, Web designers and end users each have specific roles that contribute to the end users’ experience.

Web designers are responsible for building the look and feel of the portal. They design the graphics and the layouts that will be used in the portal.

Portal administrators are responsible for controlling user access to the portal. They can make decisions on which applications and designs are available to users. They are also responsible for selecting what designs will be applied to the portal.
Portal administrators can also assign specific applications and portal pages are mandatory for users. They can specify where applications reside and what capabilities.

Users are able to add applications to their portal, as well as create new portal pages. They can modify the default layout of the portal page, assuming the portal administrators have given them the proper permissions to do so.

### 3.1.2 Portal layout

Portal Content is organized into Page groups, Pages and Portlets.

- Portals may have multiple Page Groups.
- Page Groups may contain multiple portlets.
- Pages may contain multiple portlets.
- Customized pages are saved on a per-usage and per-page basis.
- If a portlet supports multiple devices, the portlet customization is also saved on a per-device basis.
- Portal layout is made up of and controlled by Row Containers and Column containers, which contain either more containers or controls.

**Note:** A container is a row or column on the page. A row container stacks content horizontally. A column container stacks content vertically.

- Containers and container content can be locked, but keep the following points in mind.
  - Manage rights are required.
  - Locked containers cannot be deleted.
  - Locked container content cannot be moved or deleted.
  - Containers can be container content of other containers.

**Note:** For additional information on Row and Column containers, refer to the redbook *IBM WebSphere Portal Developers Handbook*, SG24-6897 (available after January 2003).

Portlets are laid out on pages. All WebSphere Portal functionality (administration, customization, etc.) is delivered via portlets. The Portal Administration pages use a Portlet Selector portlet to provide menu-like access to portlets on the page. When combined with the NoSkin skin, these functions appear to be single windows served to the user. This technique can be used on any type of page.
Various WebSphere Portal layout options are shown in Figure 3-1.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>![icon]</td>
<td>Add portlet from list</td>
</tr>
<tr>
<td>![icon]</td>
<td>Add a column container. Allows you to add a column to a row.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Add a row container. Allows you to add a row to a column.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Delete container or portlet. When you delete a container that has portlets, individual portlet settings are lost.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Add portlet from page</td>
</tr>
<tr>
<td>![icon]</td>
<td>Place existing layout and content in a new container. If the root container is a row, this icon places the existing layout inside a column, and a new row is created beneath the existing row. If the root container is a column, this icon places the existing layout inside a row, and a new column is created to the right of the existing column.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Edit portlet</td>
</tr>
<tr>
<td>![icon]</td>
<td>Move selected portlets to chosen area</td>
</tr>
<tr>
<td>![icon]</td>
<td>Move area or portlet down</td>
</tr>
<tr>
<td>![icon]</td>
<td>Move area or portlet to the left</td>
</tr>
<tr>
<td>![icon]</td>
<td>Move area or portlet to the right</td>
</tr>
<tr>
<td>![icon]</td>
<td>Move area or portlet up</td>
</tr>
</tbody>
</table>

**Figure 3-1 Portal layout options**

Portlets have customization options, based on access permissions. Take a look at the following figure (Figure 3-2).
World Clock is a portlet with Edit, Help, Minimize and Maximize options. On your portlet, the options are as follows.

- Click the Minimize icon and you can minimize your portlet; click the Maximize icon to maximize your portlet. For this option to be available, you will have to include this feature with your portlet code.
- The Edit option will allow you to edit your portlet. Changes take effect immediately. You can see this option based on access permission.
- The Help option will take you to information about the portlet. For this option to be available, you will have to include this feature with your portlet code.

### 3.2 Themes and skins

The success of a portal is tied closely to the user’s experience. As with any other Web site, the look and feel has to be appealing in terms of colors, style, and ease of use. In most cases, you will also need to match a company’s standard logo and presentation colors and patterns. WebSphere Portal makes it easy to create new themes and skins.

Themes represent the overall look and feel of the portal, including colors and fonts. They also define the layout of the portal components.
There are three basic kinds of files that make up a theme:

- **JSPs**
  Default.jsp, Banner.jsp, and Navigation.jsp are used to provide the layout and determine where the window elements go. There are a series of JSP tags available to inspect the configuration of the portal when dynamically constructing the markup for the current page. For example, you can obtain the list of active places and determine which place is currently selected.

- **Images**
  banner.jpg is the background image used by the portal banner. navfade.jpg is used by the left-side navigation. You can modify these images or create your own and access them from the theme JSPs.

- **Style sheets**
  Styles.css is the default cascading style sheet. For incoming Internet Explorer requests, the ie/Styles.css style sheet is used.

For more information on themes and skins, refer to 2.3.2, “Themes and Skins” on page 78.

Several default themes are packaged with WebSphere Portal server. These are:

- WebSphere
- Science
- Finance
- Engineering
- Corporate
- Admin

To explain the differences among different themes, we will walk through an example.

The WebSphere theme is depicted in Figure 3-3 and the Finance theme is depicted in Figure 3-4.

Notice how the portal changes when a different theme is selected. The WebSphere theme uses a color scheme based on purple, black and grey, while the Finance theme's colors are blue and grey.

The tools to navigate through the portal have also changed. In the top left corner of the WebSphere theme is a pull-down menu that allows users to select a page group. The Finance theme uses page tabs labelled Home, Work with Pages, and Portal Administration to allow the user to navigate through page groups.
Figure 3-3  WebSphere default theme
Themes are typically built by usability experts and graphical designers who have expertise in user interface design, HTML and Java Server Pages. The portal administrator is responsible for importing a theme into the portal and selecting it as a default. This is done through the portal administrative tools.

WebSphere is the default theme when WebSphere Portal is installed.

### 3.2.1 Skins

Skins are the look and feel surrounding a portlet. Skins control the frame surrounding the portlet and the title bar of the portlet. Each portlet can apply a different skin.

Several skins are packaged with WebSphere Portal. They are:

- Album
- Outline
- Hint
- NoSkin
- NoBorder
- Clear
Examples of portlet skins are shown in Figure 3-5, Figure 3-6, Figure 3-7 on page 134 and Figure 3-8 on page 134. They are examples of outline, shadow, album and noskin, respectively. The Noskin example shows a portlet without any borders or title graphics. Notice how this contrasts with the other skins.

Each skin can define the graphics surrounding the outer edge of the portlet. The shadow skin in Figure 3-4 has created a shadow effect around its borders and has tapered its top right edge. The album skin shown in Figure 3-5 has created a dotted edge border with creases at the bottom.

Figure 3-5  Portlet with Outline skin

Figure 3-6  Portlet with Shadow skin
Skins can be assigned to portlets by an administrator, a user or as a default skin for a theme. When a skin is assigned as the default for a theme on a page, all portlets are assigned to the skin automatically.

An administrator can also select which skins will be available for a given theme.

The default skin Outline is applied when WebSphere Portal Server is installed.

### 3.3 Work with pages

WebSphere Portal Server allows users and administrators to modify the applications, design, and navigation that appear on a page. The flexibility in WebSphere Portal Server allows developers, administrators and end users to provide a unique experience based on each individual's preferences. After all, the very essence of having a portal server is to be able to customize the Web site and personalize it so that users and customers feel welcome.

The *Work with Pages* page group provides the user with the ability to customize his/her experience. This includes creating page groups and pages, laying out portlets on pages, choosing skins, and locking portlets in place on a page. Users may only be allowed to customize their own experiences, or they may have access permissions that allow them to make changes that affect others.

Before you sign in to WebSphere Portal, you have the following options on the top-left hand banner of the page.
I forgot my password: if a registered user has forgotten his/her password, clicking the I forgot my password icon brings up a page which is a placeholder for a mechanism to retrieve one’s password information. That could be a phone call to the portal administrator or an e-mail notification to the support person.

Sign-up: new users can sign up and register with WebSphere Portal. Options are provided, through which users can customize their portal information based on language of choice or interests, etc. A step by step walk-through of this process is explained in Chapter 2, “WebSphere Portal administration” on page 25.

Help option: clicking the Help option will take you to the WebSphere Portal Infocenter, which is also the product documentation.

Log in to WebSphere portal and you should see a Welcome page. On the Welcome Page, select the Work with Pages page group from the drop-down menu list that you see in the top left hand corner as shown in Figure 3-9.

Figure 3-9  Select Work with Pages page group
We now explain some of the icons that you see on the top left-hand banner of the Welcome page:

- **Edit your profile**: this option will allow you to edit user profile. Click the Edit your profile option, make the changes, confirm your changes and return.

- **Help option**: clicking the Help option will take you to WebSphere Portal Infocenter, which is also the product documentation.

- **Log -off**: clicking on this option will sign you off from WebSphere Portal.

If you select the **Work with Pages** page group option, you should see a window, as shown in Figure 3-10.

![Figure 3-10 Portlets that comprises work with pages page group](image-url)
The Work with Pages page group comes with the following portlets:

- Edit Layout and Content
- Manage Places and Pages
- Set Permissions
- Choose Skins

These portlets will help you with customizing your page.

3.4 Manage Places and Pages

Use the Manage Places and Pages portlet to add a new page group. One attribute of a page group is a theme. An administrator can assign a specific theme to a page group or you can specify that the page group should use the portal default theme. Once the page group is created, the administrator will be able to change the page properties, delete pages, activate/deactivate pages and create pages for that page group.

With the Manage Places and Pages portlet, shown in Figure 3-11, you can:

- Create or delete a place
- Modify the properties for a place (the theme, markup, and locale specific titles)
- Change the ordering of places
- Create or delete a page belonging to a place
- Modify the properties for a page (the markup and locale-specific titles)
3.4.1 Create place

You can use the Create place option to name and create a new place. To be able to use this option, you must have Create permission.

By default, any user in WebSphere Portal will have Create permission for Places and Pages.

To make sure you have those permissions, log in to WebSphere Portal as administrator or user with admin privileges.

Note: Each place can contain multiple pages.
1. Go to **Portal Administration->Security->Access Control Lists**; under Select the objects for permissions, make sure you select **Resource type permissions**.

2. Click **Get groups and users**.

3. Select the user. You can do a blind search (*). Click the **Go** button.

4. Add the user to the list and click **OK**.

   You will see the user displayed under the selected users and groups list.

5. Click the **Go** button once you have made this selection.

6. The page will refresh and you will see the privileges or access permissions for this particular user.

To create a place, do the following.

1. Click the **Create place** option in the Manage places and pages portlet.

   You will see a window open as shown in Figure 3-12.

---

**Figure 3-12   Create a place**
2. Specify the new place name under the Place name and default locale title option.

**Note:** In our example, we have named the place Handbook Place and have kept the default settings. We shall use this place throughout for an explanation of customization.

3. Select the theme from the drop-down menu for your place. The selected theme will appear in the theme preview on the right-hand side of the page. You can select various themes and, based on the preview, finalize the theme.

**Note:** You should not apply the theme with the name *Admin* to a place. This theme is intended for administrative portlets and renders portlets without a title bar.

4. Choose the markup that your place will support.

5. Click **Set locale-specific titles** to custom place titles to other languages. You will see this option only if your portal is configured to support multiple languages. A list of locales appears. Select a locale, then click **Set title for selected locale**. Click **OK** to make the changes effective.

6. Click **OK** to create a place or **Cancel** to return.

7. You will see this new page added to the list of pages you can manage.

### 3.4.2 Manage place properties

This option will allow you to edit an existing place name, change the theme for the place, and set locale-specific titles.

1. Select the place which you need to edit.

2. Click **Manage place properties**.

3. Make the necessary changes.

4. Click **OK** to accept the changes or **Cancel** to go back to the previous window.
3.4.3 Activate/Deactivate place

When you create a place, by default it is active. This option will be useful while creating and modifying a place.

1. Select the place you would like to deactivate.
2. Click the Activate/Deactivate option.
3. You will see Inactive next to your place. Clicking Activate/Deactivate again will activate this place. You cannot use the place when it is in Inactive state.

3.4.4 Delete place

This option will help you to delete a place.

1. Select the place you would like to delete and click Delete place.
2. You will see a window with a warning message for confirmation.
3. Click OK if you would like to delete or Cancel to return.
4. If you select OK, this place will be deleted from the list of available places in the portal.

Important: When a place is deleted, pages within that place are also removed, and any individual user settings made to portlets on those pages are lost.

3.4.5 Order all places

You can use this option to change the order of your place. By default, Home page is the first place that you see on the list. This option will allow you to have your place as the first place if required.

1. Click the Order all places button.
2. Select your place and choose either the Before arrow or After arrow to move your place up or down.
3. Click Done. Changes take place immediately. After the changes, take a look at Figure 3-13, which shows the order of the Handbook place. We have moved one place higher in the list and you can also check this change with the drop-down list in WebSphere Portal Home page.
Figure 3-13  Change the order of your place

**Important:** This option only appears to a user who has Manage permission for the entire portal.

### 3.4.6 Manage pages

You can use this option only if you have Manage permission.

Select a place for which you need to create a page and click the Manage Pages option. You should see a window as shown in Figure 3-14.
You have the following functionalities with this portlet:

- Create page
- Manage page properties
- Activate/Deactivate page
- Define list of permitted portlets
- Delete a page

**Create page**

This option will help you create a new page from an existing place. You must have Create permission to create a page to a place. Using this option, you can also reference an existing page, apply a layout, select supported markups, define a list of associated portlets, and specify locale-specific titles.

If you reference an existing page, the page name, layout, supported markups, locks, skins, portlet list, and locale-specific titles are predetermined by the...
If you choose to reference an existing page, you must have Manage and Delegate permissions for the page that is referenced.

1. Click the **Create page** option.

2. A window will open as shown in Figure 3-15.

![Create page options](image)

*Figure 3-15 Create page options*

3. Select the **Create new** option. This option will create a new blank page. The Reference Existing option will inherit all layout, content, locks, titles, markups, and portlet lists from another page. The option to create a new page that references an existing page is present only if you have Create permission for pages, View or greater permission on a place, and Manage and Delegate permission for a page in a place. In this example, we will use the Create new option.

4. A page will open as shown in Figure 3-16.
5. Specify the administrative name and default locale title. For this example, we have specified the page name as ITSOTester.

6. Select a page layout you would like the page to use. The column layout can be changed later using the Edit layout and content portlet.

7. Specify the supported markup. In our example, we have only HTML as the option since we had selected HTML as the only markup for our Handbook Place.

8. Click **Set locale-specific titles** to specify locale-specific names for the place. A list of locales appears. Select a locale, then click **Set title for selected locale**. A prompt for the title appears. Type in the place name for the selected locale, then click **OK**.

   **Note:** This option will appear only if your portal is configured to support multiple languages.

9. Click **OK** to create a page or **Cancel** to return.
10. You will see the page ITSOTESTER added to the list of pages you can manage, as shown in Figure 3-17.

![Figure 3-17  New page added to the list of pages you can manage](image)

**Limitation:** With the current version, which is used for this book, WebSphere Portal 4.1.2 cannot add a page into a page, or a place into a place. This functionality will be offered with the next release of WebSphere Portal.

**Manage page properties**

Using this option, you can edit the page name, supported markup, and locale-specific titles with this option. If the page references another page, only the Administrative name and default locale title can be modified. The option to specify locale-specific titles is only available if the portal is configured to support multiple languages.

1. Select the page for which you need to modify information.
2. Make the necessary edits and click the **OK** button to confirm the changes or **Cancel** to return.
Activate/Deactivate
Use this option to activate/deactivate a page. By default, the new page you create is in Active state.

1. Select the page and click the **Activate/Deactivate** button.
2. The page will refresh and you will see Inactive next to your page. When your page is in Inactive state, you cannot use the page.

Define list of permitted portlets
Using this option, you can define a list of portlets associated with a page or add portlets to an existing list of portlets associated with a page. This option allows you to associate specific portlets with a page. If you reference a page that already contains a portlet list, you can add new portlets to the list, but you may not remove existing portlets inherited from the referenced page.

1. Select the page from the list of pages you can manage.
2. Click **Define list of permitted portlets**. You will see a window open as shown in Figure 3-18.

![Figure 3-18 Define list of permitted portlets](image-url)
3. Click the Add portlets option to add portlets to the page. If the selected page references another, a list of inherited portlets is displayed. This list can only be modified by making changes on the referenced page.

4. Select a search method. You can show all available portlets, or you can search by name or date modified.

5. For the option Modified since, enter the date the portlet was modified using the YYYY, MM, DD format.

6. Click Go to display the search results.

7. A list of available portlets, including name, title, description, and supported markups, is displayed in the Search Results table.

8. Click OK to add a portlet to the portlet list. You can continue to enter new search criteria and search until your list is complete.

9. In the portlet list, you can remove a portlet from the list by selecting a portlet and clicking Remove from list. You can also clear the list of all portlets.

10. By default, you will have the option Allow all portlets that a user can access selected. This allows users to any portlet in this page.

11. If you choose the option Allow only these portlets, you can limit the users to the portlets that you have associated with this page.

12. To remove a permitted portlet from the page, highlight the portlet you want to remove and click Delete portlet.

13. Similarly, you can use the Clear list button to clear the portlets from the list of permitted portlets.

14. Click OK to define a list of permitted portlets or Cancel to return.

Delete a Page

This option will allow you to delete a page from a place. When a page is deleted, layout settings and any individual user settings made to portlets on those pages are lost. If you delete a page that another page references, both pages are lost.

1. Select the page that you would like to delete.

2. Click the Delete a Page option.

3. You will see a page open with a warning message requesting your confirmation before the page is deleted.

4. Click OK if you are sure or Cancel to return. If you click OK, this page will be deleted from the place.

5. Click Done to return to Manage Places and Pages portlet.
3.4.7 Order pages

This option is used to change the order of the page in your place. You need to have Manage access for the place. This option displays all pages within the group.

1. Select the place from the list of places you can manage.
2. Click the Order pages option.

You will see a window as shown in Figure 3-19.

![Figure 3-19 Set the order for the pages](image)

3. To change the order of the pages in the place, click Before and After and you will notice the changes.
4. Click Done to exit this window.
5. If you do not make any changes, the top page will be the default page that place will display when the user logs in next time.
6. For the current session, the move of these pages is immediate and you can notice the changes.
3.5 Edit Layout and Content

The Edit Layout and Content portlet allows you to define the physical appearance of a portal page in terms of rows and columns, along with the specific portlets on the page.

Using the Edit Layout and Content portlet, you can:

- Determine how a page is structured
- Determine what portlets you see in a page
- Set column widths
- Arrange the order or portlets or where they will be positioned on a page
- Create column and row containers

**Tip:**

- Each page group can have its own theme.
- Each portlet can have its own skin.

1. Select the **Work with Pages** page group; the Edit Layout and Content portlet is the default portlet that you see, as shown in Figure 3-20.
2. Select a **Place** from the drop-down menu available in the Edit Layout and Content portlet.

3. We will select for this example **Handbook Place**, which we created earlier.

4. The page will refresh and will provide you with a list of available pages for this place.

5. Select a page you wish to modify. We will select **ITSOTESTER** as shown in Figure 3-21.
6. Once you have selected a page for modification (in our example, **ITSOTESTER**), you should see a window open as shown in Figure 3-22.

![Figure 3-21 Select a page you wish to modify](image_url)
3.5.1 Adding portlets to a page

The first step before you add a portlet to a page is to generate a list of available portlets.

1. Click the **Get portlets** icon.
2. Specify a search criteria for searching the portlets. You can search for all portlets or a specific portlet by name or portlet modified date.
3. Click the **Go** button.
4. You will see a list of portlets displayed.

**Note:** If specific portlets are already associated with the selected page, the portlet list will be locked. Only portlets available in the portlet list can be placed on the selected page.
5. Select the Portlet you wish to add to your page by clicking the plus sign next to the portlet name. Once you click the plus sign, you should see the portlet added to the portlet list. You can click **Remove from list** to remove a portlet from the available portlet list or click **Clear** to clear the list of portlets. In our example, we have added Welcome Portlet and World Clock as shown in Figure 3-23.

![Figure 3-23 Select Portlets required for your page](image)

6. Click **OK** to confirm your selection or **Cancel** to cancel the selection list.

You should see the portlets you have selected, in our example Welcome Portlet and World Clock portlet, listed as the available portlets, which you can add to your page.

7. Select the portlet you want to add to the page and click the add symbol in the desired column or row container to add the selected portlet from the list to the portlet container.

8. You can use the icon with a down-arrow and a check mark to move a portlet from one container to another.
In our example, we have added Welcome Portlet to the left side of the page (left column container) and World Clock to the right-side (right column container) as shown in Figure 3-24.

When you select a page to modify, the page is deactivated to prevent users from seeing the page as you work on it. The tasks that you perform on the page, depends on the access permission.

9. Click the **Activate** icon to activate the page when you have finished, so that users can access it. You will notice that the page will turn to Deactivate state.

To test how the portlets are laid on your page:

1. Select your page group from the top left-hand corner of the page. In our example, we select **Handbook Place**.

2. You will see two pages, HELLO TESTER and ITSOTESTER. These are the two available pages in our Handbook Place page group.

3. Select the **ITSOTESTER** tab and you should see a window as shown in Figure 3-25.
You will notice Welcome Portlet on the left-side of the page and World Clock portlet on the right-hand side of the page.

**Tip:** If you open your page and do not see any portlet, make sure that you have activated the page. For any changes that you make on the page, you need to click **Activate**.

### 3.5.2 Modifying page layout

Once you create a page, you can choose to display one, two, or three columns. The Edit Layout and Content portlet provides you with the capability to customize your page. You can rearrange Portlets and Containers.
Rearranging portlets
You can rearrange portlets anywhere you want on the page.

- The checkbox you see on the container next to the portlet is only for selecting a portlet to move it to another container on the page.
- You can move the Welcome Portlet from the left side container to the right side by checking the Welcome Portlet on the left side and clicking the down arrow and check mark icon on the right-column container. You will see the changes with the Edit Layout and Content portlet, as shown in Figure 3-26, and changes in your page as shown in Figure 3-27.

Figure 3-26  Rearrange your portlets
To add a container for example, click **Show layout controls**. You will see additional icons as shown in Figure 3-28. The various options as shown in the figure are as follows.

- **A** - This option will allow you to add column containers.
- **B** - Clicking this option will place your entire layout into a separate container. If the root container is a row, this icon places the existing layout inside a column, and a new row is created beneath the existing root row. If the root container is a column, this icon places the existing layout inside a row, and a new column is created to the right of the existing column.
- **C** - Clicking this icon will open a pop-up window where you can set column width for the page. Column width can be set by pixels or by percent.
- **D** - Moves the column container to right.
- **E** - Moves the column container to left.
– F - Adds Rows to Column.
– G - You can use this to delete a portlet.

![Additional options for changing your layout.](image)

Figure 3-28  Additional options for changing your layout.

Once you finish making changes to the layout, you can click **Hide Layout Controls**. Then click **Activate** for the changes to take effect.

### 3.6 Set Permissions

The Set Permissions portlet allows you to lock containers on a page and the contents of containers on a page. For example, in an intranet portal, you want to be sure your employees see the content of a Bulletin Board portlet containing important employee notices; but at the same time, you want the employees to be able to tailor other content (portlets) on the Welcome page.

You can also decide which portlets can or cannot be deleted from the page. These settings control how a user can work with the page in the Edit Layout and Content portlet.
Use the drop-down lists at the top to select the place and page that you want to modify. Then lock or unlock the containers or container contents. All changes occur as you make them. When you select a page to modify it, the page becomes deactivated so that users cannot access it. Be sure to activate the page when you are finished making changes.

You must have Manage access for a page in order to modify the permissions settings. An end user must have Edit access for a page in order to modify unlocked containers or container contents.

Figure 3-29   Set Permissions portlet

Select the Set Permissions portlet. You will be asked to select the place and a page, for which you need to set permissions (see Figure 3-29).

For our example, we will select Handbook Place and the ITSOTESTER page. You should see a window similar to Figure 3-30.
You will see a check mark next to the portlet on your page. This is the default setting.

- Portlets can be deleted if the Delete icon for the portlet is available for users who are modifying the page using Edit Layout and Content portlet. This is when the portlet is check-marked.
- Deselect this option, and users using the Edit Layout and Content portlet cannot delete the portlet since this option will not be available on the portlet.

**Locking the container content (A or B)**

When you click the option A or B as shown in Figure 3-30, container contents for a page are locked, and a user with Edit access to the derived page cannot perform any of the following tasks with the contents (the icons for performing these tasks do not appear when the user works with this page in the Edit Layout and Content portlet).

- Add portlets to the container.
- Remove portlets from the container.
Move portlets inside of the container into other containers.
Add sub-containers (columns or rows) to the container.
Remove sub-containers (columns or rows) from the container.
Modify portlet positions within the container.
Modify sub-container positions within the container.
Set the width for the container.

You need to click the same option again to unlock the container contents. When the container content is unlocked, a user with Edit access to the derived page can perform any of the tasks with the contents. The icons for performing these tasks appear when a user works with this page in the Edit Layout and Content portlet.

**Container locked (C or D)**

When a container is locked, it cannot be removed from the page. The Delete icon for this container is not available when the user works with this page in the Edit Layout and Content portlet.

Click the icon again to unlock the container. When it is unlocked, it can be removed from the page. You can see the delete icon when the user tries to edit the page using Edit Layout and Content portlet.

**Important:** You need to click **Activate** for changes to take effect.

### 3.7 Choose Skins

Choose Skins will allow you to change the way an individual portlet appears on a specific page.

1. Select the **Work with Pages** page group and the **Choose Skins** option from the tab.
2. Select the place and page for which you need to choose skins. For our example, we will choose the **Handbook Place** and the **ITSOTESTER** page as shown in Figure 3-31.
Changing the skin will change the look and feel of the portlet. The page that the portlet is on belongs to a place. The place has a theme associated with it. The theme has a set of skins associated with it. This set of skins is the set from which you can choose a skin for the portlet.

1. Take a look at Figure 3-25. We will change the skin using Choose Skin portlet and you can notice the difference.

2. Next to the Portlet, you have a drop-down menu option. Select the skin you want your portlet to have on the page. Each portlet can have different skins on a page.

3. Click the eye icon and you can preview the skin.

4. Outline is the default skin applied to the portlet when WebSphere Portal is installed. For our example, we have changed Welcome Portlet to have Shadow Skin and World Clock portlet to have Hint Skin.

5. Select your page group and page from the drop-down menu option at the top of the page and you will see a window as shown in Figure 3-32.
6. Click **Activate** for the changes to take effect.

**Note:** You can refer to Chapter 2, “WebSphere Portal administration” on page 25 for additional reference.
Chapter 4. Web Services

This chapter describes how Web Services are used in WebSphere Portal V4.1; we include scenarios using distributed portals.
4.1 Web Services

Web Services are self-contained software components that are published, located, and invoked over the Internet or an intranet, using standard protocols and interfaces. Requesters and providers exist in today’s architectures, but the standards-based nature of Web Services adds powerful integration to the nature of distributed services.

For example, the Web Services architecture allows an application at one company to query a service at another company to determine the availability of a product. This is currently possible without using Web Services, though it requires each company to modify its applications so that the services can be located and interact with each other. Using standard enabling technologies, an application or service can be made available over the network without regard to platform, language, location, or implementation of the service.

WebSphere Portal uses the Web Services Architecture and technologies described in the following sections to provide the remote portlet capabilities.

4.1.1 Web Services concepts

This section gives a brief overview of the Web Services concepts used by WebSphere Portal.

Web Services architecture

To enable Web Services, a service provider must make their service available, a service requester must locate the service and then the requester must invoke the service to complete the communication. This is accomplished via Publish, Find and Bind in the services-oriented architecture, as seen in Figure 4-1.
Chapter 4. Web Services

Publishing and unpublishing involves promoting a service to a registry or removing those entries. When a service is listed in a UDDI registry, it can be discovered and subsequently invoked by the service requestor.

*Find*
The Find operation is performed by service requestors and service brokers together. The service requestors describe the kinds of services they are looking for, and the service brokers deliver the results that best match the request.

*Bind*
The Bind operation takes place between the service requestor and the service provider. After locating a particular service, the requestor can bind to the service using the SOAP protocol.

*Service requestor*
A service requestor uses an API to ask the service broker about the services it needs. When the service broker returns results, the service requestor can use those results to bind to a particular service. Those services can be invoked to create applications.
**Service broker**
The service broker helps service providers and service requestors find each other.

**Service provider**
The service provider deploys a service to a UDDI registry that makes those services available. The functions of a given Web service are described using the Web Services Description Language (WSDL).

**Web Services technologies**
Web Services rely on standard transport technologies (such as HTTP) and data encoding techniques (such as XML) to make it easier for applications and devices to share information across the Internet. This interaction uses an XML document to define the interface and describe the service, along with a network protocol (which could be HTTP, SMTP, or JMS). Because the service provider and the service requester do not know what platforms or languages the other is using, interoperability is achieved. Figure 4-2 on page 170 shows the main components and operations of the Web Services architecture.

**UDDI (Universal Discovery, Description and Integration)**
UDDI is a specification for information registries of Web Services. A UDDI-based registry is where a Web service is discovered. UDDI's approach to discovery is to have a registry of services distributed across the Web. In that distributed registry, businesses and services are described in a common XML format. The structured data in those XML documents is easily searched, analyzed, and manipulated. Currently, there are a number of global registries that allow businesses to find each other across enterprise boundaries.

**WSDL (Web Services Description Language)**
WSDL is an XML-based language for describing the interface of Web Services. The service requestor can use WSDL to find a compliant service and the service provider uses WSDL to describe the service it is providing.

**SOAP (Simple Object Access Protocol)**
SOAP is a joint submission to the World Wide Web Consortium (W3C) by IBM, Microsoft, and other industry leaders. It is an XML protocol for exchanging messages and defining how those messages are to be processed. These messages can be sent using standard transport protocols, such as HTTP or HTTPS.
4.2 Web Services in WebSphere Portal

As Web Services become the predominant method for making information and applications available programmatically on the Internet, portals need to allow for integration of Web Services as data sources and as remote application components. WebSphere Portal makes use of the Web Services technologies to provide remote portlet capability. In this section, we will look at how WebSphere Portal integrates with the Web Services technologies to provide remote portlets.

There are two kinds of portlets:

- **Local portlets** - portlets that run on the Portal server itself.
  
  Local portlets are installed using the standard methods of Archive files described in “Step 5: Deploy the WAR file” on page 13.

- **Remote portlets** - portlets that run as Web Services on remote servers.
  
  Remote portlets are published as services to a UDDI registry. When the administrator adds a remote portlet to the portal, a portlet proxy is registered in the Portal's portlet registry. The portlet proxy is a generic placeholder that invokes a portlet located on a remote server via the Remote Portlet Invocation (RPI) protocol based on SOAP.

One possible application of portlets as Web Services, is in a federated portal scenario. Consider the ability to expose portlets from multiple portals into a composite portal. Within a large organization, it may be desirable to have separate portals for various divisions within the organization. In cases where the need exists for the same portlets across several portals, Web Services gives you the flexibility to install the common portlets on just one server and use Web Services to surface the portlet on another portal. From the end user perspective, the remote portlet is seamlessly integrated into the rest of the portal, when in fact it is actually running remotely, on the original Portal server that published it. The effect is to have a federated portal, where portlets may be running at any location in the network of portals.

Individual portlets can also use Web Services internally to deliver their functionality, as seen in Figure 4-2. For example, a search portlet might query the user for a search string, then use a search Web service to search the Internet; or a calendar portlet might act as a front end, providing views for a calendar Web service. WebSphere Studio Application Developer provides development tools for quickly developing Web Services and for generating proxy classes from WSDL descriptions. In the case of remote portlets, the portlet is actually running on a remote server and is accessed via the Portlet Proxy, making it transparent to the local user. In the case of a local portlet calling a Web service, the portlet is running locally and the Web service is running remotely also access by a proxy. The difference is where the portlet itself runs.
Web Services are accessible as portlets to WebSphere Portal as long as they expose the Remote Portlet Web Service (RPWS) interface. Once a service requestor locates a Web service to use in a portal, the requestor binds with the Web service through the Simple Object Access Protocol (SOAP). Not all Web Services use SOAP. It is the definition of the service through WSDL that actually defines it to be a Web service.

For remote portlets to be dynamically integrated into a portal, the portlets have to be provided as a Web service. This requires a Remote Portlet Web Service (RPWS) interface description in WSDL. The WSDL description defines a common set of methods for all remote portlets along with the required parameters and return values for each method. This RPWS description is produced and placed into the UDDI registry when the WebSphere Portal administrator publishes the portlet (Figure 4-3).

Since the WSDL describes simply the methods, parameters and return values, the implementation does not have to be Java. As long as the service described conforms to the RWPS interface it can be used as a remote portlet.
Once the remote portlet is published to the UDDI registry, a portal administrator searches the registry for services that implement the RPWS interface and add them to their portal's registry. As discussed earlier, a portlet proxy actually gets registered on the portal registry. Once the portals are in the registry, the user can add them to their portal Web pages.

When the page that references a remote portlet is rendered by the portal aggregation, the portal uses the portlet proxy to invoke the remote portlet Web service via the Remote Portlet Invocation (RMI) protocol (Figure 4-4). The portlet invokes the portlet proxy the same way it would a local portlet passing portlet request and portlet response objects.
The portlet proxy internally invokes the SOAP Proxy that marshals all the parameters and invokes the SOAP Wrapper. The SOAP Wrapper on the Web Service side unmarshals the parameters and invokes the Web Service. On return the response in returned in a SOAP response from the Web Service via the SOAP Wrapper. The SOAP Wrapper marshals the response and sends a SOAP response to the SOAP Proxy who in turn unmarshals the response for the portlet proxy. The portlet proxy finally returns a portlet response in a similar way to how a local portlet would return a portlet response. The response is then returned by the portal and the user transparently sees the response from the remote portlet as if it were a local portlet.

The WebSphere portal administration function for Web Services, as seen in 2.2.5, “Managing Web Services” on page 65, describes the functions performed by the portal administrator to publish and find Portlet Web Services. Through these functions, the WSDL description of the RPWS interface is automatically generated and placed in the UDDI registry and the Portlet Proxy is generated and placed in the portal registry for you.
4.3 The WebSphere UDDI Registry and WebSphere Portal

There are four ways of working with Web Services in WebSphere Portal.

- **Publishing portlets**
  Administrators publish portlets to the UDDI registry for use in other portals and remote portlets.

- **Finding and binding**
  Administrators can locate remote portlets via the UDDI registry and add them to the portal to be invoked.

- **Using remote portlets**
  Users invoke remote portlets transparently from their local portal.

- **Advanced finding and using remote portlets**
  The UDDI registry can be browsed by an advanced user to locate a remote portlet and make use of its services.

The key to WebSphere Portal remote portlets is the use of the UDDI registry. WebSphere Portal provides the ability to define portlets as Web Services. Through the portal user interface, you can access a UDDI registry to:

- Publish a portlet as a Web service
- Search for a portlet on the registry to add to your portal

In 2.2.5, “Managing Web Services” on page 65, we used the IBM test registry to publish and locate a portlet Web service. In this section, we install and configure the IBM WebSphere UDDI registry for use with WebSphere Portal.

4.3.1 Installing the IBM UDDI Registry V1.1.1

Review the following sections to begin the installation of the IBM UDDI Registry.

**Prerequisites**

The IBM WebSphere UDDI Registry assumes that the following products are already installed on the user’s system:

- DB2 Enterprise Edition 7.2 FP5 or FP6

**Important:** Please note that if DB2 is not installed in the default location of C:\Program Files\SQLLIB; you may be required to perform some additional steps after completing the installation.
A Web server, such as IBM HTTP Server 1.3.19, Internet Information Server 5.0, or any other Web server supported by WebSphere Application Server (refer to: "http://www-4.ibm.com/software/webservers/appserv/doc/v40/prereqs/ae_v403.htm"

WebSphere Application Server Advanced Edition (AE) V4.0.3 or WebSphere Application.

WebSphere Server Advanced Edition-Developer Only Option (AEd) V4.0.3. can also be used for the WebSphere UDDI registry. We do not explore the use of AEd in this chapter.

Important: It is recommended that you install the UDDI registry on an independent server. At the time of writing this book, this was required because WebSphere Portal does not run on WebSphere Application Server 4.0.3, which is required for the UDDI registry.

A browser:

- Internet Explorer V5.5 with SP2 and security fix Q321232 (these must be applied in that order), or
- Netscape Navigator 6.1 or later

The IBM WebSphere UDDI Registry will run on the following platforms:

- Windows 2000 Advanced Server SP1 or SP2
- Windows 2000 Server SP1 or SP2
- Windows NT Server 4.0 SP 6a
- Red Hat Linux 7.1, 2.4 kernel
- SuSE Linux for Intel 7.1, 2.4 kernel

Download the IBM WebSphere UDDI Registry V1.1.1 zip or tar file from:


Expand the downloaded file into a temporary directory to begin installation.

Preinstallation

Several steps are necessary to ensure a successful install. These steps are as follows:

1. Ensure that you have upgraded the JDBC drivers for DB2 to the 2.0 level. Installation of DB2 defaults the JDBC drivers to the 1.0 level. You must
upgrade these drivers to the 2.0 level by running the usejdbc2.bat command file.

**Note:** On Windows systems, usejdbc2.bat is located in<br/>\<DB2-install-dir>\Java12, where \<DB2-install-dir> is the install directory for DB2 (the default for \<DB2-install-dir> is C:\Program Files\SQLLIB).

On Linux systems, usejdbc12 is located in $HOME/sqllib/Java12, where $HOME is the home directory of the DB2 instance.

Note that the DB2 JDBC Applet Server must be stopped (for example, via the Services menu under Windows) when you run usejdbc2.bat (usejdbc12 under Linux), otherwise you will get a file locked error.

Failure to run usejdbc2.bat on Windows, or usejdbc12 on Linux, will cause the WebSphere Application Server to fail on startup with an error code of 10.

- If you do not have security turned on in your WebSphere Application Server, then you are advised to install the IBM WebSphere UDDI Registry before enabling WebSphere security. If you have security turned on, then the UDDI Registry installation program prompts you for your WebSphere user ID and password several times. You must provide your WebSphere user ID and password within a few minutes of being prompted, otherwise some of the WebSphere setup performed by the installation will fail.

- Ensure that DB2 is started.
  - On Windows operating systems, it will be started automatically, unless you have changed the installed defaults. You can check whether DB2 is started, and start it manually, from the Windows Services application (accessed via the Control Panel for Windows NT and via the Administrative Tools option in the Control Panel for Windows 2000). The service name is DB2-DB2.
  - On Linux operating systems, you should switch to the DB2 instance you intend to use for the UDDI Registry, using su - db2inst1, where db2inst1 is the instance ID, then ensure that sqllib/db2profile has been executed, either by having been called from the ID’s profile, or by executing '../sqllib/db2profile'. You can then run db2start to start DB2.

- Ensure that you have a DB2 user ID (and password) with administrative privileges, needed to create and access the UDDI Registry database. This user ID should obey the rules for DB2 user ids. You will be prompted for this user ID and its password during the UDDI installation.

- As part of the installation process, the module visibility of your application server is changed to Application, which is required by the UDDI Registry application.
Installation

Please note that the WebSphere Application Server will be recycled during this install, so you should stop any other WebSphere work before running setup.exe, and close the WebSphere Administrative Console.

To start the installation of the IBM WebSphere UDDI Registry, do the following:
1. Run the setup program extracted from the UDDI registry download.
2. Click OK in the UDDI Registry Welcome window, as seen in Figure 4-5.

The setup program:
- Checks whether you already have an instance of the IBM WebSphere UDDI Registry installed on your system. If you do, then it will give you the option to overwrite the existing installation, or to abort the install process. Unless you are re-installing the same service level of the product, you should abort the install process and remove the existing installation using Add/Remove Programs from the Control Panel, before running setup.exe again.
- Checks that you have the prerequisite level of database product installed (see “Prerequisites” on page 173).
  - If you have the required level of one of the prerequisite database products installed, the installation uses this product for the UDDI registry. If you have more than one of the prerequisite database
products installed, then you will be asked which you would like to use for the UDDI Registry database.

- If you do not have a prerequisite level of database product installed, the installation will offer you the choice of continuing the installation (by answering Yes) or aborting the installation (by answering No). You should only answer Yes if you are planning to do a custom install of only the Information Center, the UDDI4J package, the Samples, or the EJB client files.

- Checks for the version of WebSphere Application Server installed on the system. If the appropriate level is not installed, then the installation offers you the choice of continuing the installation (by answering Yes) or aborting the installation (by answering No).

**Important:** You should only reply Yes if you are planning to do a custom install of just the Information Center, the UDDI4J package, the Samples, or the EJB client files.

You are asked to accept the licence terms and conditions (Figure 4-6). These are the licence terms and conditions that you accepted when you obtained the product; for example, from the download site.

![Software License Agreement](image)

**Figure 4-6** IBM WebSphere UDDI Registry License Agreement

3. Click OK.
4. Next, you are given the option to override the installation directory location (Figure 4-7).

To override the default directory location:

a. Select Browse and then either browse to find a suitable directory location or type an alternative location.

b. Click OK.

   If the directory does not exist, you are asked if you want to create it. Click Next to continue.

In the rest of these instructions, the directory location is referred to as <UDDI-install-dir>.

![Figure 4-7   IBM WebSphere UDDI Registry installation directory](image)

5. Next, select either a **Custom** or a **Typical** install: the Typical install is the default.

   This installs all the components of the IBM WebSphere UDDI Registry that you need to use it.

   The Custom install allows you to select one or more of the following components:
   - UDDI Registry Files
   - UDDI4J
   - InfoCenter
   - Sample files
Chapter 4. Web Services

179

The main purpose of the Custom Install option is to allow you to install other components, such as the Information Center, or the Samples, or the UDDI4J package, on a separate machine from the UDDI registry files. The components that can be selected on a Custom Install are:

- **Setup UDDI Registry Database**

  The main purpose of the Custom Install option is to allow you to install other components, such as the Information Center, or the Samples, or the UDDI4J package, on a separate machine from the UDDI registry files. The components that can be selected on a Custom Install are:

  - **UDDI Registry Files** - Installs the IBM WebSphere UDDI Registry program.

    If this is your first installation then you must select **Set up UDDI Registry Database**. The InfoCenter is automatically selected.

    If you select **UDDI Registry Files**, which installs the IBM WebSphere UDDI registry program, then you should also select at least the **Setup UDDI Registry Database** component as well.

  - **UDDI4J** - Installs the UDDI4J package.

  - **InfoCenter** - Installs the Information Center.

  - **Sample Files** - Installs the installation verification programs and the other UDDI sample programs.

  - **Setup UDDI Registry DB** - Installs the code that will load your UDDI database.

    It is essential that you select this option if this is a new installation.

    You can choose not to set up the UDDI Registry database (by not selecting **Setup UDDI Registry Database**), for example, if you are installing on top of a previous version of the IBM UDDI Registry Database, and want to preserve existing data in the registry. However, be aware that the format of the registry data may have changed, so that your old registry database may no longer be compatible.

  - **EJB** - Installs the EJB client classes and Javadocs for the EJB interface.

6. Select **Typical** and click **Next** as seen in Figure 4-8.
7. Next, you are prompted for the database user ID and password as seen in Figure 4-9.

This user ID is used to create and access the UDDI Registry database, and should be a user ID with administrative privileges which obeys the rules for DB2 user IDs.
8. Next, the UDDI registry install sets up and configures the database used by the registry.

Some of this setup needs the WebSphere Administrative Server to be running; if it is not, the installation process starts it for you.

The database setup is comprised of the following steps:

a. Create a JDBC provider in your WebSphere Application Server called UDDI JDBC Driver and an associated data source (UDDI data source). If you already have a JDBC provider called UDDI JDBC Driver then this, together with any associated data source, is replaced.
b. Create a database called UDDI20 and populate it with the tables and standard categorization schemes that are required for the UDDI Registry. If you already have a database called UDDI20, then this will be replaced (unless you choose not to Set up UDDI Registry Files, as part of a Custom install).

9. The installation process then places the uddi.ear into <UDDI-install-dir>, copies it into the installableApps subdirectory of the WebSphere application server installation, and deploys it into WebSphere.

Note: the UDDI registry will be installed on the default application server. Also, the module visibility will be set to Application. This is required to make the UDDI registry work as it is composed of Web Modules and EJB modules.

10. When the installation has finished, you will see a window asking you if you want to look at the readme. If you wish to view the readme then select the box.

11. Click Finish to complete the installation.
After completing the installation, you will be able to see the UDDI Registry application using the WebSphere Administrative Console, as follows:

1. Start the WebSphere Administrator's Console if it is not already running.

2. In the navigation pane, expand the administrative domain so that you can see Enterprise Applications, and expand that to show the applications that are installed. An application called UDDI Registry should be shown, as seen in Figure 4-15.
You have now successfully installed the WebSphere UDDI Registry. The next section details the steps necessary for verification.

4.3.2 Verification

In order to use the IBM WebSphere UDDI Registry, start the WebSphere default server (or stop and restart it if it is already running).

From the WebSphere Administrative console:

1. Expand the tree view in the left pane to show Nodes -> node-name -> AppServers -> Defaults Server.

2. Right-click Default Server and select Start.

3. Click OK when the start completes.

Installation diagnostic output

The output from the various steps that are performed is logged into log files in the <UDDI-install-dir>/logs directory, where <UDDI-install-dir> is the target directory you used when installing the UDDI code.
Installation Verification Program (IVP)

This section describes how to run the installation verification programs (IVPs) to verify that the IBM UDDI Registry has been installed correctly. It assumes the samples are installed on the same system as the Registry code.

There are two IVP SOAP samples called SOAPSampleIVPa and SOAPSampleIVPb. They are intended to verify the successful installation of the product, and should be used in conjunction with the UDDI Users Console (GUI). SOAPSampleIVPa saves some data to the registry which you can then find using the GUI. Finally, you can delete the data by running SOAPSampleIVPb.

The IVP samples are installed into the same target directory as the other SOAP samples and they use the same XML files as the basic Java SOAP samples.

SOAPSampleIVPa saves three businesses, six services (two per business) and three tModels. The data structures are very basic and consist only of a name. The keys returned by the save_* UDDI API calls are then written to a file, SOAPSampleIVPa.out. SOAPSampleIVPb then reads in these keys from the file in order to delete the saved data from the UDDI registry.

Note: Each time you run SOAPSampleIVPa, it overwrites the output file SOAPSampleIVPa.out so, if you wish to use SOAPSampleIVPb to delete the data, you must run this before you next run SOAPSampleIVPa.

To run the IVPs, complete the following steps on the same system as the UDDI Registry:

1. Ensure that DB2 and the WebSphere Admin Server are started.
   
   For SOAP samples to work, you need to ensure that the Client JDK is either the one shipped with IBM WebSphere Application Server or a later IBM JDK.
   
   - For Windows - ensure that C:\WebSphere\AppServer\Java\bin is present in the PATH statement before any other JDKs
   
   - For Linux - ensure that /opt/WebSphere/AppServer/Java/bin is present in the PATH statement before any other JDKs

   Note: You must use the IBM WebSphere supplied JDK or a later level of the IBM JDK.

   For Windows, the default system path can be set by clicking Start -> Settings -> Control Panel -> Settings-> System-> Advanced Properties-> Environment Variables.
Alternatively, this can be accomplished just for the shell where you plan to run the samples by modifying the path within the shell:

- For Windows - set path=c:\WebSphere\AppServer\Java\bin;%path%
- For Linux - export PATH=/opt/WebSphere/AppServer/Java/bin:$PATH

2. Copy the samples from <UDDI-install-dir>\samples\soap and *.xml to a directory, and compile and run them there as follows:
   a. Compile both SOAPSAMPLEIVPa and SOAPSAMPLEIVPb by typing javac SOAPSAMPLEIVPa.Java and javac SOAPSAMPLEIVPb.
   b. Run SOAPSAMPLEIVPa by typing java SOAPSAMPLEIVPa. This should publish a number of businesses and services and service types into the registry.
   c. Start your Web browser on the same system as the UDDI Registry.
   d. To display the UDDI GUI home page, type one of the following URLs:
      - If you have WebSphere security disabled: http://localhost:9080/uddigui
      - If you have WebSphere security enabled: https://localhost:9433/uddigui

3. On the find page, complete the following steps:
   a. Select the business radio button.
   b. In the data entry field, type % (the percent symbol is the wild card symbol).
   c. Click Find.
      You should get a results page returned with three businesses (mybusiness1, mybusiness2, and mybusiness3). This demonstrates that the API and the UDDI Console are working correctly.

4. To see the services that are available for a business, click the Show Services option next to the business.

5. To delete all of the IVP data, run SOAPSAMPLEIVPb (from the command prompt as before, by typing java SOAPSAMPLEIVPb).

6. On the find page, complete the following steps:
   a. Select the business radio button.
   b. In the data entry field, type % (the percent symbol is the wild card symbol).
   c. Click Find.
      You should get an empty results page returned.
4.4 Configuring WebSphere Portal with the WebSphere UDDI Registry

In this section, we walk through configuring WebSphere Portal to work with the IBM WebSphere UDDI Registry.

**UDDI GUI**

For the purposes of this example, we will not set up SSL. By default, the WebSphere UDDI Registry installs such that publishing through the UDDI GUI requires SSL. To change this we must perform the following steps.

1. Open the following file for editing:
   
   `<WAS-Root>/installedApps/UDDI_Registry.ear/gui.war/Web-INF/Web.xml`

2. Search for the tag `<transport-guarantee>` and change its content to `NONE`, for example:
   
   `<transport-guarantee>NONE</transport-guarantee>`

3. Make sure both `<transport-guarantee>` tags are set to `NONE`.

4. Save the file.

5. Restart the UDDI registry application: locate UDDI Registry, right-click it, stop and start it.

6. Invoke the registry GUI, click **Publish**.
   
   You should see a prompt for authentication; use wpsadmin / wpsadmin.

**Inquiry and publish URLs**

1. Open the following file for editing:
   
   `<WAS-Root>/installedApps/UDDI_Registry.ear/soap.war/Web-INF/Web.xml`

2. Search for the tag `<transport-guarantee>` and change its content to `NONE`, for example:
   
   `<transport-guarantee>NONE</transport-guarantee>`

3. Make sure all `<transport-guarantee>` tags are set to `NONE`.

4. Save the file.

5. In the WebSphere Application Server Admin console, restart the Default Server.

6. For verification, invoke the publish and inquiry URLs:
   
   - `http://<uddi_machine_name>:9080/uddisoap/inquiryapi`
   - `http://<uddi_machine_name>:9080/uddisoap/publishapi`
In both cases, you should get a SOAP message, that is, an XML file, indicating the SOAP fault because of which the URL cannot respond to HTTP, as seen in Figure 4-16.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Body>
    <Fault>
      <faultcode>Protocol</faultcode>
      <faultstring>Protocol Error</faultstring>
    </Fault>
    <detail>
      <dispositionReport generic="2.0" xmlns="urn:uddi-org:api_v2"><operator>www.mycompany.com/uddi</operator>
        <result errno="10500">
          <errInfo errCode="E_fatalError">E_fatalError
            (10500) Serious technical error has occurred while processing the request. IBM WebSphere UDDI Registry can only respond to POST requests.</errInfo>
        </result>
      </dispositionReport>
    </detail>
  </Body>
</Envelope>
```

Figure 4-16  SOAP fault - protocol error

### 4.4.1 Web Services administration

To work with Web Services in WebSphere Portal, you must first log in to the Portal as an administrator.

1. Select **Portal Administration pages** and the **Manage Web Services** task as seen in Figure 4-17.
2. Add the UDDI Registry by clicking **Add** next to the UDDI Web Services registries list.

3. Next, specify the registry parameters as seen in Figure 4-18.
4. Specify the UDDI registry parameters as follows.
   
   - **Display Name for Registry**
     
     This is the display name you want for the UDDI registry you are configuring. This can be any name you choose.
     
     We chose ITSO UDDI Registry.
   
   - **Registry Inquiry and Publish URIs**
     
     These are the URIs used to Query and Find information in the registry. Given a default installation (and keeping in mind that we have disabled SSL), these are:
     
     - Inquiry URI
       
       `http://<UDDI host>:9080/uddisoap/inquiryapi`
     
     - Publish URI
       
       `http://<UDDI host>:9080/uddisoap/publishapi`
tModel Key

The tModel Key is a key in the registry that identifies the service type. There is no straightforward way to determine this value. We must define the service type and then obtain its tModel Key to place in this field.

We have used the tModel Key UUID:3A29EC6935-3ECF-4D02-825C-037A05F12AFB.

For information on defining a service type and determining the tModel Key, see “Determining the tModel Key” on page 192.

5. Click OK to add the registry definition.

When the addition of the registry is successful, you will see it added to the list of registries on the Administration pages for managing Web Services, as seen in Figure 4-19.

This completes the configuration of WebSphere Portal for use with the IBM WebSphere UDDI Registry.
Determining the tModel Key

In order to determine the tModel Key that will be used, we must:

- Define a business entity.
- Add a service to the business entity.
- Determine the tModelKey for the service type we have defined.

This section walks through the steps to determine the tModel Key to be used with the UDDI registry.

1. Invoke the UDDI GUI via the URL `http://<uddi server>:9080/uddigui`.

   Note that the UDDI Registry GUI runs by default as a Web Application on the internal HTTP server, so we must specify port 9080 in the URL.

2. Define a business as seen in Figure 4-20.

   a. Select the UDDI GUI Publish tab.
   b. Select business under Quick Publish.
   c. Enter the name of the business.

   We have chosen a business called ITSO Portal.
3. Click **Publish** to publish the business.

Next, we must add a service to our business.

![WebSphere UDDI Registry](image)

**Figure 4-21  IBM WebSphere UDDI Registry - Show owned entities**

From the Publish tab in the UDDI GUI:

1. Click **Show owned entities**.

2. The business we previously defined will be displayed as seen in Figure 4-21.

3. Click **Add Service** and the Add services form will be displayed as seen in Figure 4-22.

We must now specify the name of the service we would like to define and publish it.
1. Enter the Name of the service to publish.
   Our service name is ITSO Remote Portlets.

2. Click Publish Service.

   It is the tModel of our service that we must specify in the UDDI registry definition. Unfortunately, there is no straightforward way to determine the tModel Key. To find the tModel Key for our service, do the following.

1. From the Publish tab in the UDDI GUI:
   a. Click Show owned entities.
   b. Scroll down to the Technical Models at the bottom of the page, as seen in Figure 4-23.
c. To find the tModel Key, you must select the tModel and open it in another window.

    In our case, right-click **ITSO Remote Portlets** and select **Open in New Window**.

d. The URL in the new window contains the tModel Key, as seen in Figure 4-24.
Figure 4-24  IBM WebSphere UDDI Registry - tModel key

The tModel Key is the portion of the URL that starts with UUID. In our example, the tModel Key is
UUID: 3A29EC6935-3ECF-4D02-825C-037A05F12AF8.

Note the substring %3A is replaced by a colon, ";:".
WebSphere Portal
Administration sample code

This appendix provides you with the sample Java code along with the deployment descriptors. You can package them as a WAR file using any of the packaging tools. The images that you see in 2.2.1, “Install Portlets” on page 32 and 2.2.2, “Manage Portlet Applications” on page 36 use this WAR file.

Example 4-1  Web.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
"http://java.sun.com/j2ee/dtds/web-app_2.2.dtd">
<web-app id="WebApp_1_1">
  <display-name>Read Param Portlet webapp</display-name>
  <servlet id="Servlet_1_1">
    <servlet-name>ReadParamPortlet</servlet-name>
    <servlet-class>samplepkg.ReadParamPortlet</servlet-class>
    <load-on-startup>0</load-on-startup>
  </servlet>
  <servlet-mapping id="ServletMapping_1_1">
    <servlet-name>ReadParamPortlet</servlet-name>
    <url-pattern>/ReadParamPortlet/*</url-pattern>
  </servlet-mapping>
</web-app>
```
Example 4-2 Portlet.xml

```xml
<!DOCTYPE portlet-app-def PUBLIC "-//IBM//DTD Portlet Application 1.1//EN" "portlet_1.1.dtd">
<portlet-app-def>
    <portlet-app major-version="41" minor-version="0">
        <portlet-app-name>Read config and context param app (Sample code)</portlet-app-name>
        <portlet id="Portlet_1_1" href="/WEB-INF/web.xml#Servlet_1_1">
            <portlet-name>Read Parameters Portlet</portlet-name>
            <cache><expires>0</expires><shared>NO</shared></cache>
            <allows><maximized/></allows><minimized/>
            <supports><markup name="html">
                <view output="fragment"/>
            </markup></supports>
        </portlet>
    </portlet-app>
    <concrete-portlet-app uid="samplepkg.ReadParamPortlet.concrete">
        <portlet-app-name>Read config and context param concrete app (Sample code)</portlet-app-name>
        <context-param><param-name>contextParam</param-name><param-value>app1</param-value></context-param>
        <concrete-portlet href="#Portlet_1_1">
            <portlet-name>Read Parameters Concrete Portlet 1</portlet-name>
            <default-locale>en</default-locale>
            <!-- Begin translation: -->
            <language locale="en">
                <title>Read Config and Context params</title>
                <short-title>rcc</short-title>
                <description>Portlet demonstrates how to read config and context params</description>
                <keywords>config, context</keywords>
            </language>
            <config-param><param-name>configParam1</param-name><param-value>value1</param-value></config-param>
        </concrete-portlet>
    </concrete-portlet-app>
</portlet-app-def>
```
Example 4-3  ReadParamPortlet.Java

```java
import com.ibm.wps.portlets.*;
import org.apache.jetspeed.portlet.*;
import org.apache.jetspeed.portlets.*;
import org.apache.jetspeed.portlet.event.*;
import java.io.*;
import java.util.*;

class ReadParamPortlet extends PortletAdapter{
    public void doView( PortletRequest portletRequest, 
            PortletResponse portletResponse) throws PortletException, 
            IOException{

        PrintWriter pw =  portletResponse.getWriter();
        String s = 
        portletRequest.getPortletSettings().getApplicationSettings().getAttribute("contextParam");
        pw.print("The value of context param <b>contextParam1</b> is " + s);

        s = portletRequest.getPortletSettings().getAttribute("configParam1");
        pw.print("<br>The value of config param <b>configParam1</b> is " + s);
    }
```
Additional material

This redbook refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

ftp://www.redbooks.ibm.com/redbooks/SG246920

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, SG24-6920.
Using the Web material

The additional Web material that accompanies this redbook includes the following files:

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web.xml</td>
<td>Web application deployment descriptor</td>
</tr>
<tr>
<td>Portlet.xml</td>
<td>Portlet deployment descriptor</td>
</tr>
<tr>
<td>ReadParamPortlet.Java</td>
<td>Java source file</td>
</tr>
</tbody>
</table>

System requirements for downloading the Web material

The following system configuration is recommended:

- **Hard disk space:** 100 Kb minimum
- **Operating System:** Windows 98, ME, 2000, XP
- **Processor:** Pentium
- **Memory:** 32 MB

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.
## Abbreviations and acronyms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM</td>
<td>International Business Machines Corporation</td>
</tr>
<tr>
<td>ITSO</td>
<td>International Technical Support Organization</td>
</tr>
<tr>
<td>B2B</td>
<td>Business-to-Business</td>
</tr>
<tr>
<td>B2C</td>
<td>Business-to-Customer</td>
</tr>
<tr>
<td>B2E</td>
<td>Business-to-Employee</td>
</tr>
<tr>
<td>CRM</td>
<td>Customer Relationship Management</td>
</tr>
<tr>
<td>DMT</td>
<td>Directory Management Tool</td>
</tr>
<tr>
<td>DN</td>
<td>Distinguished Name</td>
</tr>
<tr>
<td>DNS</td>
<td>Domain Name Service</td>
</tr>
<tr>
<td>DNS</td>
<td>Domain Name Services</td>
</tr>
<tr>
<td>EJB</td>
<td>Enterprise JavaBeans</td>
</tr>
<tr>
<td>ERP</td>
<td>Enterprise Resource Planning</td>
</tr>
<tr>
<td>GNOME</td>
<td>GNU Network Object Model Environment</td>
</tr>
<tr>
<td>GNU</td>
<td>UNIX-like operating system</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language</td>
</tr>
<tr>
<td>IBM</td>
<td>International Business Machines Corporation</td>
</tr>
<tr>
<td>IHS</td>
<td>IBM HTTP Server</td>
</tr>
<tr>
<td>IIOP</td>
<td>Internet Inter-ORB Protocol</td>
</tr>
<tr>
<td>ITSO</td>
<td>International Technical Support Organization</td>
</tr>
<tr>
<td>J2EE</td>
<td>Java 2 Platform, Enterprise Edition</td>
</tr>
<tr>
<td>JDBC</td>
<td>Java Database Connectivity</td>
</tr>
<tr>
<td>JDK</td>
<td>Java Development Kit</td>
</tr>
<tr>
<td>JRE</td>
<td>Java Runtime Environment</td>
</tr>
<tr>
<td>JSP</td>
<td>Java Server Pages</td>
</tr>
<tr>
<td>JVM</td>
<td>Java Virtual Machine</td>
</tr>
<tr>
<td>KDE</td>
<td>K Desktop Environment</td>
</tr>
<tr>
<td>LDAP</td>
<td>Lightweight Directory Access Protocol</td>
</tr>
<tr>
<td>LTPA</td>
<td>Lightweight Third Party Authentication</td>
</tr>
<tr>
<td>LUM</td>
<td>License Use Management</td>
</tr>
<tr>
<td>PDA</td>
<td>Personal Digital Assistant</td>
</tr>
<tr>
<td>RDN</td>
<td>Relative Distinguish Name</td>
</tr>
<tr>
<td>RPM</td>
<td>Red Hat Package Manager</td>
</tr>
<tr>
<td>SASL</td>
<td>Simple Authentication and Security Layer</td>
</tr>
<tr>
<td>SCM</td>
<td>Supply Chain Management</td>
</tr>
<tr>
<td>SMIT</td>
<td>System Management Interface Tool</td>
</tr>
<tr>
<td>SSL</td>
<td>Secure Socket Layer</td>
</tr>
<tr>
<td>URI</td>
<td>Uniform Resource Identifier</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>WCM</td>
<td>WebSphere Content Manager</td>
</tr>
<tr>
<td>WCP</td>
<td>Web Content Publisher</td>
</tr>
<tr>
<td>WML</td>
<td>Wireless Markup Language</td>
</tr>
<tr>
<td>WMS</td>
<td>WebSphere Member Services</td>
</tr>
<tr>
<td>WPS</td>
<td>WebSphere Portal</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
<tr>
<td>XSLT</td>
<td>Extensible Stylesheet Language Transformations</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 2003. All rights reserved.
Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 207.

- Domino and WebSphere Together Second Edition, SG24-5955-01
- Deploying Quickplace, SG24-6535-00
- Customizing Quickplace, SG24-6000-00
- Lotus Discovery Server 2.0: Deployment, Planning, and Integration, SG24-6575-00
- Inside the Lotus Discovery Server, SG24-6252-00
- WebSphere Portal Collaborative Components, REDP0319
- IBM WebSphere Portal V4.1 Handbook Volume 1, SG24-6883-00
- IBM WebSphere Portal V4.1 Handbook Volume 3, SG24-6921-00
- IBM WebSphere V4.0 Advanced Edition Handbook, SG24-6176-00
- Enterprise Business Portals II with IBM Tivoli Access Manager, SG24-6885-00

Referenced Web sites

These Web sites are also relevant as further information sources:

- Lotus Domino Workflow
  http://www.lotus.com/products/domworkflow.nsf
- InfoCenter - Lotus Workflow
- WebSphere Personalization
- WebSphere Portal
- Lotus Developer Domain
  http://www-10.lotus.com/ldd/
- Domino 5.0.8 Release Notes
  http://doc.notes.net/uafiles.nsf/docs/rn508/$File/readme.pdf
- Sametime 2.5 Release Notes
- Sametime Installation Notes
  http://doc.notes.net/uafiles.nsf/docs/ST25/$File/stinstall.pdfs
- Quickplace Installation Notes
  http://doc.notes.net/uafiles.nsf/docs/QP208/$File/QPAdminBP.pdf
- Juru - Full-text search library
  http://www.haifa.il.ibm.com/km/ir/juru/
- IBM Corporation
- WebSphere Application Server - Infocenter
  http://www-3.ibm.com/software/webservers/appserv/doc/v40/ae/infocenter/was/0606080004aa.html
- Portlet Development Guide
- UDDI Business Test Registry
- UDDI Business Test Registry 2
  https://uddi.ibm.com/ubr/registry.html
- Microsoft Business Registry Node
  http://uddi.microsoft.com/default.aspx
- HP Business Registry
  http://hpmiddleware.com/SAISAPI.dll/SaServletEngine.class/products/hp_web_services/registry/default.jsp
- IBM WebSphere UDDI Registry
How to get IBM Redbooks

You can order hardcopy Redbooks, as well as view, download, or search for Redbooks at the following Web site:

ibm.com/redbooks

You can also download additional materials (code samples or diskette/CD-ROM images) from that site.

IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.
Index

A
abstract portlet class 3
access control list 111
Access Control List Administration Portlet 106
Access Control portlet 32
administration portlets 26
assets 79
authentication 111
authorization 111

B
bind operation 167
branding 78
Bulletin Board 159

C
Cascading Style Sheets (CSS files) 79
Choose Skins 162
CJK language support 96
calendar device definitions 22
Clipping Runtime Portlet 59
core portlet 5
core portlet application 5
Configure Search Index 94
Content Organizer portlet 114, 120
core repository 119
credential segments and slots 22
credential vault 113
customization 126
customization roles 126

D
DB2 JDBC Applet Server 175
decorations 78
deployment descriptors 10
Directory Management Tool (DMT) 100
Distinguished Name (DN) 109
DMT 100
document search 92

E
EIP 92
EJB 179
Enable Tracing 76, 98
Enterprise Information Portal (EIP) 92
Enterprise Information Portal Search 92

F
Figure 2-8 36
find operation 167
finding and binding 173

G
Global Settings 76

H
help menu icon 31
hierarchy for a portlet class 4
HTML 132
HTTP 168
HttpServletRequest 6
HttpSession 6

I
IBM Portlet API 8
IBM Tivoli Access Manager 114
IBM UDDI Registry 173
IBM WebSphere UDDI Registry 175, 187
images 79, 130
InfoCenter 179
input request XML file 20
Install Portlets 32
Installation Verification Program (IVP) 185
Internet Explorer 174

J
J2EE applications 10
JAR file 9
jar utility 9
Java SDK 9

© Copyright IBM Corp. 2003. All rights reserved. 209
Index

R
Read Concrete portlet 35
readparameters.war 38
readparameters.war file 32
Red Hat Linux 174
Redbooks Web site 207
Contact us xiii
register resource collections 118
Remote Portlet Invocation (RMI) 171
Remote Portlet Invocation (RPI) 169
Remote Portlet Web Service (RPWS) 170
remote portlets 169, 173
request XML file 21
Rich Site Summary (RSS) 115
RPWS 170

S
security 28
Security portlet 111
service broker 168
service provider 168
service requestor 167
servlet 3
servlet class 10
Set Permissions 159
shared credential slot 113
Simple Object Access Protocol (SOAP) 168, 170
skins 79, 81, 132
SMTP 168
SOAP 168–170
SOAP protocol 167
SOAP proxy 172
SOAP wrapper 172
style sheets 130
SuSE Linux 174
system credential slot 113

T
Tag library descriptor files (TLD) 9
theme 78, 130
Themes and Skins 76
themes and skins 22, 129
TLD 9
valueOfKey 191–192, 195
Tooling Portlet 58
Trace Logger 98
types of permissions
delegate 111
del 111
manage 111
view 111

U
UDDI 168
UDDI directory 32
UDDI registry 167, 171
UDDI Registry Files 179
UDDI Web Services 69
UDDI4J package 179
Universal Discovery, Description Integration (UDDI) 65, 168
updatesetparams.war 42
usejdbc12 175
usejdbc2.bat 175
users and groups 28

V
vault segment 113
vault service 113

W
WAR 8, 13
Web application deployment descriptor, Web.xml 4
Web archive file (WAR) 3, 8, 32
Web Clipping 32
Web Clipping Portlet 58
Web content 3
Web pages 23
Web Services 32, 165, 169
Web Services architecture 166
Web Services Description Language (WSDL) 168
Web Services portlet 32
Web.xml 4
Web-INF directory 9
WebSphere Application Server 9
WebSphere InfoCenter 22
WebSphere Personalization 115
WebSphere Portal 10, 20, 23, 26, 166, 169
examples 3
fundamental characteristics 3
WebSphere Portal Welcome Page 29
WebSphere Studio Application Developer (WSAD) 8
WebSphere Studio Workbench 8
The IBM WebSphere Portal V4.1 Handbook is available in three volumes of Redbooks. This is volume 2. These IBM Redbooks position the IBM WebSphere Portal for Multiplatforms as a solution that provides a single point of interaction with dynamic information, applications, processes and people to help build successful business-to-employee (B2E), business-to-business (B2B), business-to-consumer (B2C) portals.

WebSphere Portal consists of three packaged offerings:
- Portal Enable
- Portal Extend
- Portal Experience

In the three volumes of the IBM WebSphere Portal V4.1 Handbook, we cover WebSphere Portal Enable and Extend.

The IBM WebSphere Portal V4.1 Handbook will help you to understand the WebSphere Portal architecture, how to install and configure WebSphere Portal, how to administer portal pages using WebSphere Portal, it will also discuss the development of WebSphere Portal portlets and how to use specific WebSphere Portal applications.

In this redbook, we discuss the administration and portlet development of WebSphere Portal. In addition, we discuss the use of Web Services.