BEA WebLogic Portal 4.0™

Getting Started with Portals and Portlets

Version 4.0
Document Date: October 28, 2002

Copyright

Restricted Rights Legend

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

Getting Started With Portals and Portlets

Document Edition	Date	Software Versions
4.06	October 28, 2002	BEA WebLogic Portal 4.0, SP2

Contents

About This Document

1. Introducing WebLogic Portal

2. Overview of Portal Administration

3. Using the E-Business Control Center Portal Tool

4. Portal Administration Tools

5. Customizing Portals and Portlets

6. Portal Management JSP Tag Library Reference

7. Portal Management Database Schema

About This Document

This document introduces WebLogic Portal, and is meant to provide complete information for users, administrators, and developers. The document begins with a description of BEA's Portal plaform and the many strategic possibilities it presents, then continues with audience and task-based chapters which ultimately provide all portal constituents with all the information necessary for exploiting its many advantages.

This document covers the following topics:

- Chapter 1, "Introducing WebLogic Portal," which summarizes the most important aspects of the new platform, identifying key concepts you will need to make use of the new features.
- Chapter 2, "Overview of Portal Administration," which explains the processes and roles required to manage portals on this platform.
- Chapter 3, "Using the E-Business Control Center Portal Tool," which explains using the E-Business Control Center to aggregate and deploy, or "sync", portal resources for use in web applications.
- Chapter 4, "Portal Administration Tools," which provides step-by-step instructions on using the browser-based Portal Administration tools to create associations between resources deployed using the E-Business Control Center.
- Chapter 5, "Customizing Portals and Portlets," which explains developer essentials such as customizing WebFlow and PipeLine components.
- Chapter 6, "Portal Management JSP Tag Library Reference," which lists all the JSP Tags specific to Portal.
- Chapter 7, "Portal Management Database Schema," which lists specifications for database schema used by WebLogic Portal.

What You Need to Know

This document is intended for the following audiences:

- End-users include Business Analysts, who should be comfortable using a web browser such as Netscape or Internet Explorer, as well as using whatever functionality being exended by the Portal they are using.
- Administrators, who must understand how to manage portals, will benefit by reading Chapter 2, “Overview of Portal Administration.” Instructions on using the tools for designating, deploying and associating portal resources are covered in Chapter 3, “Using the E-Business Control Center Portal Tool,” and Chapter 4, “Portal Administration Tools.”
- Developers interested in using the BEA WebLogic Portal framework to produce e-commerce web sites should first consult Chapter 5, “Customizing Portals and Portlets.” Familiarity with BEA’s Weblogic Server, as well as the J2EE specification will also be helpful.

e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the “e-docs” Product Documentation page at http://e-docs.bea.com.

How to Print the Document

You can print a copy of this document from a Web browser, one file at a time, by using the File—>Print option on your Web browser.

A PDF version of this document is available on the WebLogic Portal documentation Home page on the e-docs Web site. A PDF version of this document is also available in the documentation kit on the product CD. Or you can download the documentation kit from the WebLogic Portal portion of the BEA Download site.

If you do not have the Adobe Acrobat Reader, you can get it for free from the Adobe Web site at http://www.adobe.com/.

Related Information

The *Portal Tour* document offers a hands-on look at the sample applications included with WebLogic Portal.

The following BEA documents contain information relevant to the entire WebLogic Portal product suite.

- *Deployment Guide*
- *Strategies for Developing E-Business Web Sites*
- *Guide to Managing Presentation and Business Logic: Using Webflow and Pipeline*

For more information about BEA's Personalization and Commerce offerings, refer to the Bibliography at http://edocs.bea.com/.

Contact Us!

Your feedback on the BEA WebLogic Portal documentation is important to us. Send us e-mail at **docsupport@bea.com** if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the WebLogic Portal documentation.

In your e-mail message, please indicate that you are using the documentation for the BEA WebLogic Portal 4.0 release.

If you have any questions about this version of BEA WebLogic Portal, or if you have problems installing and running BEA WebLogic Portal, contact BEA Customer Support through BEA WebSupport at **www.bea.com**. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
boldface text	Indicates terms defined in the glossary.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.
italics	Indicates emphasis or book titles.

`monospace text`	Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard.
	Examples:
	`#include <iostream.h> void main ( ) the pointer psz`
	`chmod u+w *`
	`\tux\data\ap`
	`.doc`
	`tux.doc`
	`BITMAP`
	`float`

`monospace boldface text`	Identifies significant words in code. *Example*: `void` **`commit`** `( )`
`monospace italic text`	Identifies variables in code. *Example*: `String` *`expr`*
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. *Example*s: LPT1 SIGNON OR

Convention	Item
`{ }`	Indicates a set of choices in a syntax line. The braces themselves should never be typed.

`[ ]`	Indicates optional items in a syntax line. The brackets themselves should never be typed.
	Example:
	`buildobjclient [-v] [-o name ] [-f file-list]...`
	`[-l file-list]...`

`|`	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.

`...`	Indicates one of the following in a command line:
	■ That an argument can be repeated several times in a command line
	■ That the statement omits additional optional arguments
	■ That you can enter additional parameters, values, or other information
	The ellipsis itself should never be typed.
	Example:
	`buildobjclient [-v] [-o name ] [-f file-list]...`
	`[-l file-list]...`

. . .	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.

CHAPTER

1 Introducing WebLogic Portal

BEA WebLogic Portal simplifies, personalizes, and lowers the cost of access to information, applications, and business processes for enterprise, customer, and partner portals. WebLogic Portal is built upon the BEA WebLogic E-Business Platform - an integrated suite of infrastructure products built on open standards that support high-volume transactions, business process management, application integration, and business collaboration within and across the enterprise.

WebLogic Portal features can be categorized as follows: Portal Foundation Services, personalization and interaction management, intelligent administration, integration services, and portlets.

This topic includes the following sections:

- Portal Functionality Becomes a Platform
- Key Portal Concepts
- What's New in WebLogic Portal?

Portal Functionality Becomes a Platform

Early content-driven portals allowed visitors to log in, select aggregated content, and perform limited layout customization. Users of a simple news portal, for instance, could register and create a custom layout that included news coverage of specific topics, as well as weather updates for a certain region.

Recently, so-called “second-generation” portals were introduced. These portals run on an application server, allowing the ad hoc introduction of business logic within an intranet. WebLogic Portal provides a complete platform by wrapping a presentation framework around WebLogic Portal and WebLogic Personalization Server with Campaign and Commerce services.

Major enhancements have been made to the tools available for developing, deploying, and managing portal resources. Support for delegated administration and role-based entitlements have also been added.

WebLogic Portal simplifies the creation of personalized portal sites by providing a user interface framework, pre-built presentation elements and templates. A rich set of administration and customization tools controls the behavior of the user interface framework, the content presented via the framework, and the user interaction with the portal site.

Who Can Use WebLogic Portal?

The WebLogic Portal platform is designed for a wide spectrum of users, including business analysts, administrators, JSP developers, HTML and graphics designers, and system analysts.

- HTML and graphics designers can easily define new look-and-feel specifications and portal page layouts, applying these new designs to content aggregated by portal administrators.
- JSP developers can customize the default portal presentation and layout templates to accommodate sophisticated user interface requirements, adding fine-grained personalization, navigation logic and session functionality using portal Webflow JSP tags.
- JSP developers can create new portlets, as explained in the section “Creating Your Own Portlet,” in Chapter 5, “Customizing Portals and Portlets.” JSP developers can also work with Java and EJB developers to integrate third-party applications, or customize the portal framework and create new applications. This can be accomplished using Portal functionality exposed in JSP tag libraries, such as user and group profile, rule-based personalization, and content management.
- Finally, individuals can be assigned limited control of certain parts of the group portal to which they belong, enabling page maintenance to be

delegated, reassigned, and revoked in a few simple steps. This feature is explained in more detail in the section “Delegated Administration.”

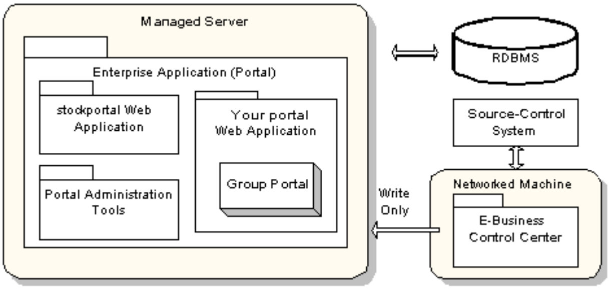
Key Portal Concepts

Many of the terms defined in this section have been used differently elsewhere; a few terms are new and are specific to WebLogic Portal. Together, these definitions provide an unambiguous starting point for understanding how the platform works.

Relationships Among Applications and Portals

The J2EE application hierarchy in WebLogic Portal can be described starting at the domain level on a WLS application server. This domain contains enterprise applications which in turn contain web applications. The enterprise application depicted in Figure 1-1 constitutes a single instance of the portal framework. This single web application can contain multiple group portals. Each group portal can include multiple pages, also known as portal pages. Each of these pages consists of a configurable array of discrete content modules called portlets.

Figure 1-1 Portal Application Overview


Managed Server
Enterprise Application (Portal)
stockportal Web Application
Your portal Web Application
Group Portal
Portal Administration Tools
RDBMS
Source-Control System
Networked Machine
E-Business Control Center
Write Only

What Is a Portal?

A portal is a framework for a Web site or for an application that aggregates information from a variety of sources. A portal allows access to an information network and to a set of services via the World Wide Web.

The portal included with the WebLogic Portal product provides the following:

- A layout paradigm for dynamic, interactive, personalized content
- Content modules called portlets, laid out in rectangular grids
- The ability to personalize portlets at many levels
- The ability to save customized layout settings made by a visitor to your Web site
- The ability to define multiple views for the same group portal
- The ability to designate and customize color schemes called skins
- Delegated administration, which enables complex, distributed security implementations

What Is a Group Portal?

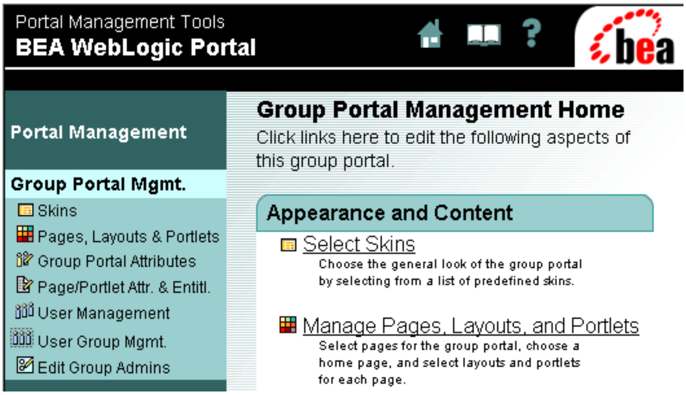
A group portal is a specific view of the portal defined for a given group of users. You can create multiple group portals within a portal Web Application. The group portals can share portal resources, such as layouts and portlets, but can be configured differently to satisfy the needs of each group separately. Because users are designated individually as members of a group, the group portal uses a static form of personalization. This user group is associated with a portal to form a group portal.

Other forms of content personalization include the following:

- Dynamic, rules-based personalization, where resources or capabilities are granted or denied at runtime, based on customizable profile parameters
- Visitor customization, where an authenticated user makes changes to individual settings that will persist from one session to the next

Figure 1-2 illustrates using the WebLogic Portal Administration Tools to administrate a group portal.

Figure 1-2 Administrating a Group Portal



Group Portal Management Home

Click links here to edit the following aspects of this group portal.

Appearance and Content

Select Skins

Choose the general look of the group portal by selecting from a list of predefined skins.

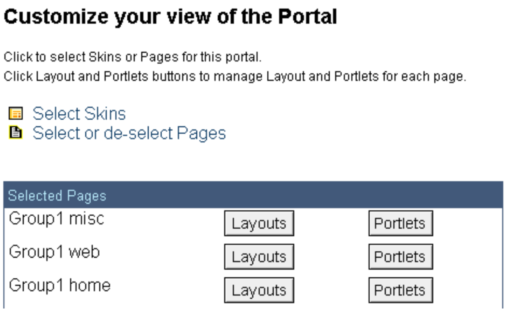
Manage Pages, Layouts, and Portlets

Select pages for the group portal, choose a home page, and select layouts and portlets for each page.

What Is a Portal Page?

A portal page is a single view of an array of content modules that can be configured by users at many different levels. Figure 1-3 shows visitor customization of a portal page after authenticating. Changes made during this session will be evident immediately.

Figure 1-3 Visitor Customization of Portal Pages



Customize your view of the Portal

Click to select Skins or Pages for this portal.
Click Layout and Portlets buttons to manage Layout and Portlets for each page.

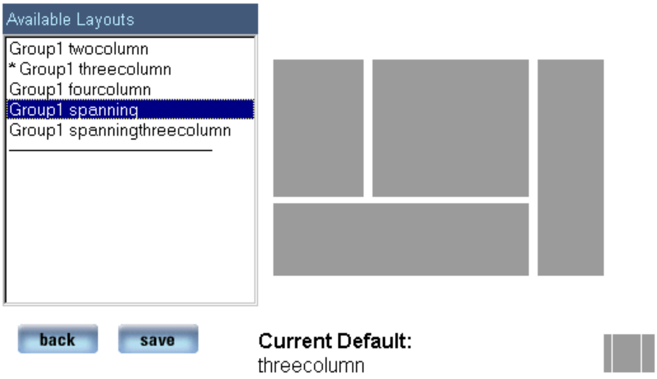
Selected Pages		
Group1 misc	Layouts	Portlets
Group1 web	Layouts	Portlets
Group1 home	Layouts	Portlets

Within a group portal, multiple tabbed pages, also referred to as portal page views, can be created and customized, enabling you to make optimal use of layout space and allowing for quick and efficient navigation within a portal.

What Is a Layout?

Layouts are predefined configurations of named placeholders within a portal page. A set of portlets can be arranged on a given layout in one page, and can be reused or rearranged on the same or a different layout for a different page. Figure 1-4 illustrates how to use the WebLogic Portal Administration Tools to choose the default layout.

Figure 1-4 Selecting Layouts



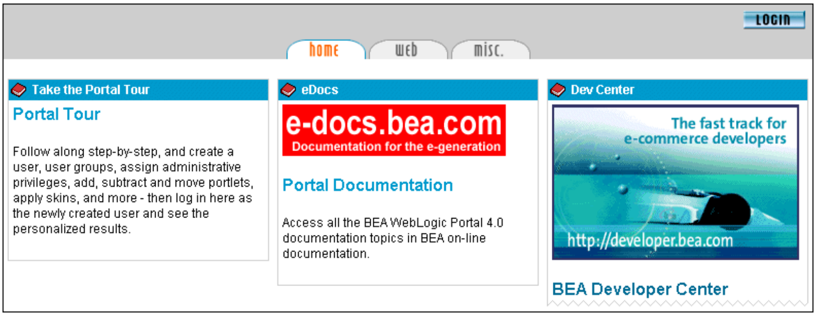
Current Default:
threecolumn

What Is a Portlet?

Architecturally, a portlet is a JSP with some XML-based metadata that fits into a portal. Strategically, a portlet is a highly focused channel of information directed to a specific user or group using a portal as its container.

Figure 1-5 shows three portlets called Portal Tour, eDocs, and DevCenter, displayed on a portal page called Home.

Figure 1-5 Portlets Within a Page



Portal Tour

Follow along step-by-step, and create a user, user groups, assign administrative privileges, add, subtract and move portlets, apply skins, and more - then log in here as the newly created user and see the personalized results.

Portal Documentation

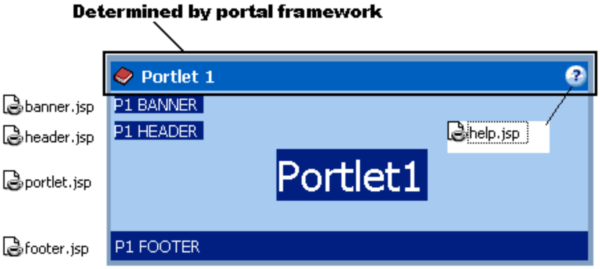
Access all the BEA WebLogic Portal 4.0 documentation topics in BEA on-line documentation.

BEA Developer Center

Each portlet can contain certain layout elements.

Figure 1-6 shows the banner, header, content, and footer sections of a portlet as they appear in a portal page, with references to the JSP files used to create them. In this reference implementation, one of a number of title bars can be displayed, based on the state of the portlet at runtime.

Figure 1-6 Anatomy of a Portlet



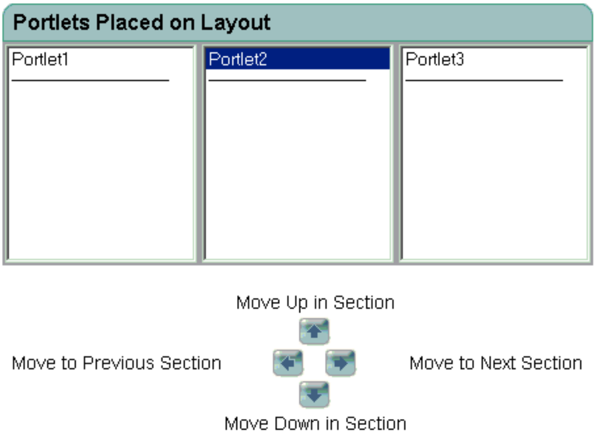
Each of these sections can be designated as separate JSP pages, and the portlet can further be assigned the following characteristics:

- Editable— user can access a developer-supplied edit page to make changes to the portlet.
- Floatable—portlet can be displayed outside the portal container in a separate browser window.
- Minimizable—user can minimize the portlet, showing only the title bar.
- Maximizable—visitor can expand the portlet to the full size of the browser window.

- Helpable—user can open online help from within the portlet.
- Mandatory—User cannot remove the portlet.
- Defaults Minimized—Only the title bar is initially displayed.
- Login Required—Visitors must authenticate to be able to see the portlet.

Individual portlets are arranged in portals using layouts. Figure 1-7 shows how portlets can be manipulated using directional arrows in the Layout Editor tool.

Figure 1-7 Arranging Portlets in a Layout



Multiple portlets can be assigned to one or more portals, their inclusion based on a flexible set of determiners. Application functionality can be implemented within these highly focused content channels. For example, an employee portal could elicit immediate employee feedback to a general corporate communication, subgroups could be given access to departmental applications that have been “portalized,” and each employee could be given secure access to personal benefits data within a custom portlet.

Much of this content personalization can be accomplished with relatively little coding on the part of the Business Engineer (BE).

Note: For an explanation of how roles are defined within this documentation set, consult the “Visitor, Customer, and User Roles” topic in the *Strategies for Developing E-Business Web Sites* document.

What Are Skins?

Skins are sets of presentation attributes applied to a portal by the portal framework. Using this mechanism, you can efficiently manage the appearance of a portal without changing the content.

Skins define the look and feel of a portlet, including portlet control images (buttons used to invoke, edit, or minimize portlet views), default font face and size, and color attributes for parts of the portlet. Skins are made up of Cascading Style Sheet (CSS) files and directories of images, and you can design your own and place them inside your portal framework.

Personalization Versus Customization

In the WebLogic Portal platform, *personalization* means the ability to change the content and functionality presented to an authenticated visitor; that is, to an end user who has successfully logged into a portal. Once logged in, a visitor can *customize* various aspects of the portal, provided these controls have been made available by the administrator who set up the portal.

The WebLogic Portal platform enables personalization at several different levels, as follows:

- When a visitor logs in, the visitor’s membership in one or more groups is evaluated.
- Once group membership has been evaluated, entitlements allow access to portal resources, based on dynamic evaluation of attributes.
- If the visitor has performed any customization of the portal, these settings are applied to the portal resources to which the visitor has access, and the page is rendered accordingly.

Visitor Entitlements

WebLogic Portal provides a powerful mechanism to control what is presented to a visitor to your portal. Visitor entitlements are managed within the WebLogic Portal Administration Tools, and make use of Entitlement Segments, which are created in the E-Business Control Center. This process is explained in detail in Chapter 5, "Customizing Portals and Portlets."

Basically, for pages and portlets, you can define the access and the degree to which a user may change the behavior of a portlet. For example, you could give members of one group the ability to edit the layout of a certain portlet, while denying the same ability to members of another group.

Visitor entitlements in the WebLogic Portal platform go far beyond this simple example. The rules you can create in the E-Business Control Center can involve user attributes stored within the user properties set, or they can be based on system parameters, dates, and so on. These entitlement segments can also control an array of factors beyond simple access to view a portlet. They can be used to invoke personalization, campaign, or commerce functionality as well. This is explained in some detail in Chapter 5, "Customizing Portals and Portlets."

Delegated Administration

Maintaining large Web applications requires the ability to mete out administrative tasks to individuals. For instance, you might want to assign the management of skins to a graphic design department, and user management to the IT department. Delegated administration allows these areas to be compartmentalized efficiently, enabling the creation of administrator roles with fine-grained administrative privileges.

Using this Delegated Administration functionality is explained in Chapter 2, "Overview of Portal Administration," in the section called "Delegated Administration."

What’s New in WebLogic Portal?

In addition to the new presentation and interface features introduced in the section called “Key Portal Concepts,” this release of WebLogic Portal introduces the following features:

- New Deployment Scheme
- New Administration Features
- New Development Features
  - Reference Implementation Portals and Portlets
  - Custom JSP Tags
  - The E-Business Control Center
  - Portal Administrator Tools
  - New Application Infrastructure

New Deployment Scheme

Metadata for portal applications is now deployed using the sync feature of the E-Business Control Center portal module. This deployment is in parallel with whatever method you use to move the actual graphics, HTML files, JavaScript files, cascading style sheets, skins or layouts that make up the presentation component of your portal applications. BEA recommends that the E-Business Control Center obtain this metadata from your version control system.

The E-Business Control Center assists in the management of metadata for portal web applications, allowing administrators with little or no understanding of XML configuration files to perform complex configuration and deployment tasks.

For detailed instructions on deploying applications, consult the *Deployment Guide* for WebLogic Portal.

New Administration Features

The browser-based Portal Administration Tools allow fine-grained delegated administrator types specific to Portal web applications as well as Group Portals. Fine-grained, predefined administration tasks can be delegated to these new delegated administrator types.

New Development Features

Several new development features have been added to the portal framework. Webflows are now part of each portal, and webflows can be assigned to individual portlets.

Reference Implementation Portals and Portlets

The WebLogic Portal includes an entire, functioning application which can be copied and used as a template for your portal applications. The sample portals in Table 1-1 demonstrate some of the capabilities of the new platform:

Table 1-1 Reference Portlets

Portlet Name	Comment
Portal Tour	Demonstrates a number of the portal features available in WebLogic Portal.
eDocs	Online library of all BEA documentation.
Developer Center	Centralized online resource for code samples, enhancements to products, and developer discussions.
About Avitek	Retrieves HTML documents using a content selector. Clicking a document header opens the document in a full-screen mode within the portal.
Stock Quote	Retrieves a stock quote from a web site.
Dictionary	Look up the definition of a word by connecting to Merriam-Webster site.

Table 1-1 Reference Portlets

Portlet Name	Comment
Web Search	Web search through Yahoo, Excite, or Lycos.
Basic Portlet Sample	A "hello world" example portlet.
Portlet WebFlow	Demonstrates a simple portlet Webflow.
Portlet 3 – text display portlet	Demonstrates a static content portlet.

New Ways to Develop Portal Applications

The Portal framework has been rebuilt according to J2EE specification for enterprise applications. The following mechanisms are provided to assist you in creating custom portal applications.

New Application Infrastructure

The new framework enables portals and portlets to be associated with pre-built or customized presentation elements, as well as business logic customized for the intended audience. For example, individual portlets can be reused among separate web applications, and can be assigned orthogonal webflows within different portal applications. Standards-based implementation such as J2EE and XML ensure interoperability, ease of deployment, as well as availability of expertise for customization. The presentation elements included with the product can be extended, allowing you to create custom layouts as well as custom look and feel sets called skins.

Custom JSP Tags

Custom tag are included with the installed product, along with documented samples to illustrate how these JSP tags can be used to invoke functionality in portlets. These JSP tags can be used within portlets to enable HTML designers to avoid code and still make use of advanced features.

The E-Business Control Center

New modules within the E-Business Control Center are used to define and deploy custom portals and portlets. This tool also allows the creation of custom webflows for portals and portlets.

Portal Administrator Tools

The browser-based Portal Administrator Tools run on WebLogic Portal server. These tools allow you to configure your portal with fine-grained control over what visitors can see and do on your site, and allow you to grant permissions for administrative tasks, called delegated administration.

2 Overview of Portal Administration

Managing WebLogic Portal effectively requires an understanding of J2EE security concepts as well as a grasp of security features unique to the WebLogic Portal platform. Portal administration requires the ability to manage access on a many-to-many basis. In other words, access to many different groupings of resources must be provided to many different groupings of users. For this reason, the WebLogic Portal platform goes beyond the J2EE security standard as currently written, and also encompasses existing WebLogic Server security schemes.

This topic provides additional detail about the three levels of administration that WebLogic Portal supports, and includes information about how the access granted to these administrator users is scoped. Additionally, this topic includes information about how administrative users are managed, and how administration tasks may be delegated. Finally, some information about visitor entitlements for WebLogic Portal is provided.

Using the information in this topic, a System Administrator should be able to understand the basic tasks required to make portals available to developers, other administrators and ultimately to site visitors.

This topic includes the following sections:

- Portal Administration Concepts
- Administrator Tasks

Portal Administration Concepts

This section explains the following key aspects of administrating the WebLogic Portal platform:

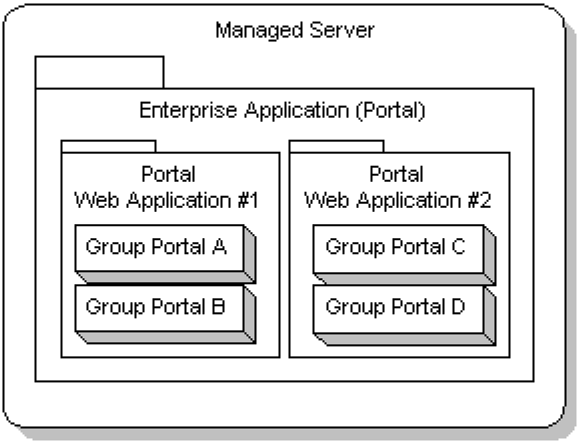
- Scoping Portal Resources and Services
- Understanding the Administrator Hierarchy
- Administration Considerations Specific to Portal
- Portal Rendering

Scoping Portal Resources and Services

Administrating portal Web applications means managing relationships among Web application-scoped objects such as users, user groups, portlets, skins, and layouts. But the Portal Platform also exposes enterprise-scoped elements such as Campaign and Commerce services.

Figure 2-1 shows two portal Web applications within a single enterprise application. Two group portals reside within each of these portal Web applications. In this illustration, group portal A and group portal B can share Webflows, portlets, skins, and layouts. To use the same Webflows, portlets, skins or layouts in portal Web applications #1 and #2, however, you would need to make copies of all the files used to support those resources—meaning the J2EE files and the metadata files maintained by the E-Business Control Center. Campaign and Commerce services, however, are enterprise-scoped, and thus can be accessed from any portal Web application within the enterprise application.

Figure 2-1 Application and Enterprise Scoping



Note: The relationship between enterprise applications, portal Web applications and group portals is introduced in "Relationships Among Applications and Portals" on page 1-3.

Understanding the Administrator Hierarchy

The "Development Roles" section of the *Strategies for Developing E-Business Web Sites* documentation explains most of the significant constituents in the WebLogic Portal workflow. A few are specific to the WebLogic Portal Administration Tools, and pertain specifically to the management of portal resources and visitor access to those resources.

Three Levels of Administrator Permissions

The WebLogic Portal platform recognizes three basic subdivisions of Administrators: the System Administrator (Portal SA), Portal Administrator (Portal PA), and Group Administrator (Portal GA). Individual Portal PAs and Portal GAs can be assigned fine-grained privileges, enabling the creation of a very complex administrator hierarchy customized to fit very specific security models.

It is important to note the distinction between these Portal Administrators and `system`, the default account used to control the WebLogic Server Administration Console. The Portal SAs, PAs and GAs use the browser-based WebLogic Portal Administration Tools, whereas `system` is the sole member of a special, unchangeable user group called `Administrator` used to start and stop WebLogic Server.

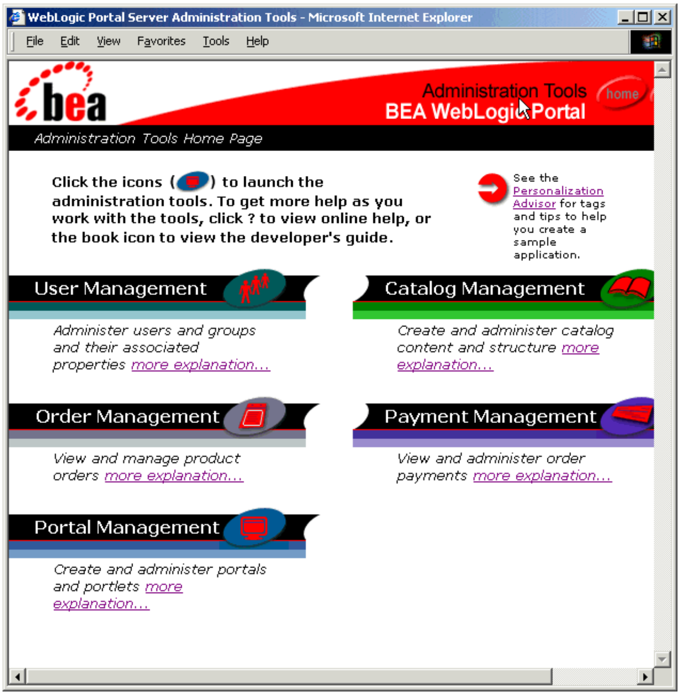
Note: For instructions on creating, editing, and deleting SA, PA, and GA users, see Chapter 4, "Portal Administration Tools."

SA- System Administrators

Out of the box, WebLogic Portal includes a Portal SA user called `administrator`, which has unlimited access to administrative tasks anywhere within the enterprise portal application.

When a Portal SA logs into portalTools Web application, (by navigating to `http://<hostname>:<port>/portalTools/index.jsp`), the WebLogic Portal Administration Tools page appears, as shown Figure 2-2.

Figure 2-2 WebLogic Portal Administration Tools home page


WebLogic Portal Server Administration Tools - Microsoft Internet Explorer
File Edit View Favorites Tools Help
bea
Administration Tools
BEA WebLogic Portal
home
Administration Tools Home Page

Click the icons ( ) to launch the administration tools. To get more help as you work with the tools, click ? to view online help, or the book icon to view the developer's guide.

See the Personalization Advisor for tags and tips to help you create a sample application.

User Management

Administer users and groups and their associated properties more explanation...

Catalog Management

Create and administer catalog content and structure more explanation...

Order Management

View and manage product orders more explanation...

Payment Management

View and administer order payments more explanation...

Portal Management

Create and administer portals and portlets more explanation...

Within the WebLogic Portal, the System Administrator (Portal SA) may perform any of the following administrative actions:

- Deploy portal applications using the E-Business Control Center
- Create a new user group
- Create and entitle Portal Administrators (PAs)
- Create/Edit a group portal
- Create and entitle Group Administrators (GAs)
- Delegate administrative tasks

Because the Portal SA has access to all possible administrative tasks available within the WebLogic Portal Administration Tools, it is recommended that you observe the following guidelines:

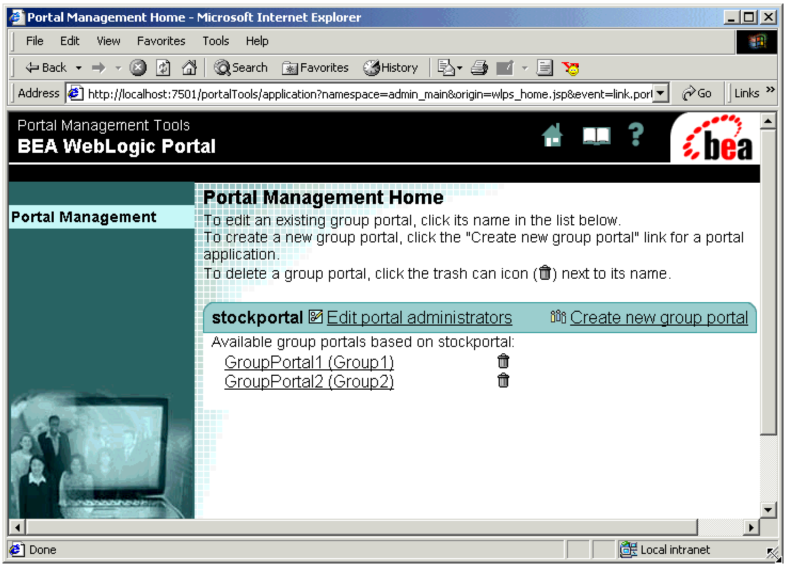
- Only use `system` to create new users, place new users in the `SystemAdministrator` user group (thereby making them Portal SAs), adding users to user groups such as `AdminEligible`, or to create a new user group. Limit the use of this powerful user, and remember to log off when necessary tasks have been completed. This will lessen the chance of its credential being compromised.
- Limit the number of users you place in the `SystemAdministrator` user group. Fewer Portal SA users on your instance of WebLogic Portal will reduce the effort required to keep the passwords fresh, and lessen the probability that one of these user accounts may be compromised.

PA - Portal Administrators

Out of the box, WebLogic Portal includes a pair of test users called `demopa1` and `demopa2`. These users have access to administrative tasks on a portal-wide basis, and as such, they can be be granted administrative priveleges to any group portal within the portal Web application.

When a Portal PA logs into portalTools Web application, (by navigating to `http://<hostname>:<port>/portalTools/index.jsp`), the WebLogic Portal Management home page appears, as shown in Figure 2-3.

Figure 2-3 WebLogic Portal Management Home page


Portal Management Home - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go Links
Portal Management Tools
BEA WebLogic Portal
bea
Portal Management
Done
Local intranet

Address: http://localhost:7501/portalTools/application?namespace=admin_main&origin=wlps_home.jsp&event=link.port

Portal Management Home

To edit an existing group portal, click its name in the list below.
To create a new group portal, click the "Create new group portal" link for a portal application.
To delete a group portal, click the trash can icon (🗑) next to its name.

stockportal Edit portal administrators Create new group portal

Available group portals based on stockportal:

- GroupPortal1 (Group1) 🗑
- GroupPortal2 (Group2) 🗑

Within one or more portal Web applications, the Portal Administrator (PA) may be granted the ability to perform any of the following administrative actions:

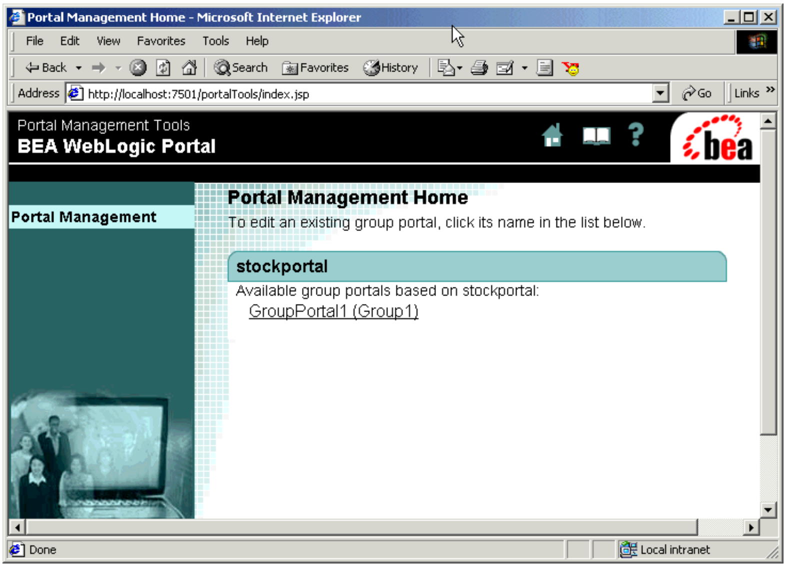
- Create and entitle Portal Administrators (PAs)
- Create/Edit a group portal
- Create and entitle Group Administrators (GAs)
- Delegate administrative tasks

GA - Group Administrators

Out of the box, WebLogic Portal includes a test users called `demoga1, demoga2` and `demoga3`. These users have access to administrative tasks at the group portal level. These users can be granted administrative priveleges to any group portal within the portal Web application. Also, Portal GAs can be promoted by Portal SA or Portal PA users.

When a Portal GA logs into portalTools Web application, (by navigating to `http://<hostname>:<port>/portalTools/index.jsp`), the WebLogic Group Portal Management home page appears, as shown in Figure 2-4.

Figure 2-4 WebLogic Group Portal Management Home page



Portal Management Home - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://localhost:7501/portalTools/index.jsp
Go Links
Portal Management Tools
BEA WebLogic Portal
bea
Portal Management
Done
Local intranet

Portal Management Home

To edit an existing group portal, click its name in the list below.

stockportal

Available group portals based on stockportal:

GroupPortal1 (Group1)

Note: Portal GAs may manage more than one group portal. In that case, the the page shown in Figure 2-4 would include the name of another group portal.

Within one or more group portals, the Portal Administrator (Portal PA) may be granted the ability to perform any of the following administrative actions:

- Create and entitle Group Administrators (GAs)
- Edit a group portal
- Delegate administrative tasks

Application Assembler/Deployer

Though not given a specific role within the administrative workflow of WebLogic Portal, the Application Assembler/Deployer is distinct in that it is most closely associated with the use of the Portal Module of the E-Business Control Center.

The Application Deployer/Assembler is the user who performs the synchronize task using the E-Business Control Center, which requires Portal SA privileges. (In other words, the Application Deployer/Assembler must be a member of the SystemAdministrator user group.)

How Users are Managed

The User Management home page allows Portal SA users to add or remove specific users from pre-defined user groups. After the user belongs to the appropriate user group, a mechanism called Delegated Administration is used to bestow specific privileges upon a single user.

Note: For information on Delegated Administration, consult the section called "Delegated Administration."

User Groups

Membership in WebLogic Portal user groups are used to grant privileges to the user as follows.

- When any user is added to WebLogic Portal, the user is automatically a member of a user group called `everyone`.
- When a company creates user accounts for administrators, all these users should be added to the `AdminEligible` user group.
- Any user can be turned into an all-powerful system administrator— regardless of membership in other user groups: simply add the user to the `SystemAdministrator` user group from the User Management home page. Adding or removing a user from this group does not affect any other user group memberships.
- Importantly, promoting a user to GA or PA within the Portal Management or Group Portal Management home pages does eliminate other group memberships for that user. For example, if a GA user is promoted to PA, that user's membership as a GA is eliminated.

Administration Considerations Specific to Portal

A few important deployment concepts should be understood before specific administration tasks are introduced.

Life Cycle of a Portal Application

Table 2-1 lists the various tasks in the cycle of developing, deploying and managing a portal Web application.

Table 2-1 Life Cycle of a Portal Web Application

Task	Role	Reference
Develop portlets, skins, images, JavaScript	JSP/HTML Developer, Web or User Interface Designer	Chapter 5, “Customizing Portals and Portlets.”
Define portal resource metadata	Business Engineer, Business Analyst	Chapter 3, “Using the E-Business Control Center Portal Tool.”
Install portal objects such as JSPs, image files, JavaScript, HTML	*J2EE* System Administrator, Application Assembler/Deployer	Chapter 5, “Customizing Portals and Portlets.”
Deploy portal resource metadata	Business Engineer, Business Analyst	Chapter 3, “Using the E-Business Control Center Portal Tool.”
Manage Administrator Roles	System Administrator	Chapter 4, “Portal Administration Tools.”
Manage portal resources and visitor entitlements	*Portal* System Administrator, Portal Administrator, Group Administrator	Chapter 4, “Portal Administration Tools.”
Configure sample users in LDAP directory	*Portal* System Administrator, Portal Administrator, Group Administrator	Replicate Sample Users in LDAP Service

Which Tool for Which Task?

Throughout the life cycle of a portal application, different tools are used to complete different steps. Table 2-2 shows the tools used for each of the tasks commonly performed throughout the life cycle of a portal application.

Table 2-2 Which Tool for Which Task?

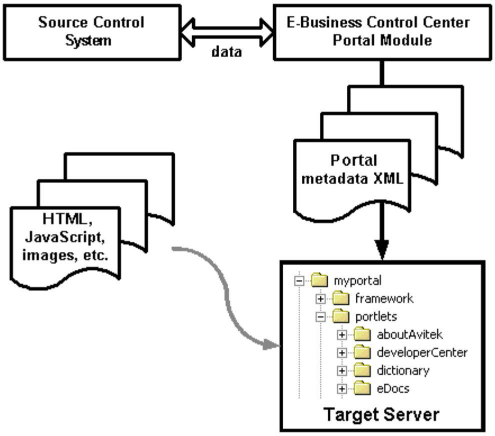
Task	Tool
Create JSPs, skins, portlets	Java IDE, HTML design tool, image editing package
Define portal metadata	E-Business Control Center
Create entitlement segments	E-Business Control Center
Deploy J2EE files	Filesystem
Deploy portal metadata	E-Business Control Center
Verify deployment	Web browser, data repository browser
Create user groups, users, group portals	WebLogic Portal Administration Tools
Create visitor entitlements	WebLogic Portal Administration Tools
Manage delegated administration settings	WebLogic Portal Administration Tools

Portal Metadata Versus Application Files

Before a Business Analyst can begin to define a new portal Web application using the E-Business Control Center, the J2EE application files must have been migrated by a J2EE Developer.

Additionally, when the Business Analyst has finished defining, troubleshooting and validating the new application, the Application Assembler/Deployer can take over and perform the deployment of the metadata. The metadata created by the E-Business Control Center is saved in XML files, and is deployed to the WebLogic Portal server.

Figure 2-5 Using the EBCC to Deploy Portal Metadata



Two important points arise from this fact:

- A version control process should be implemented to secure revisions of xml data files. This is important in cases where data synchronization may be occuring at the EBCC and server sides.
- Synchronizing Web application metadata does not deploy the web content that makes up your portal Web application. These files must be deployed outside the E-Business Control Center.

Procedures for configuring and synchronizing your E-Business Control Center are addressed in detail in the *Deployment Guide*.

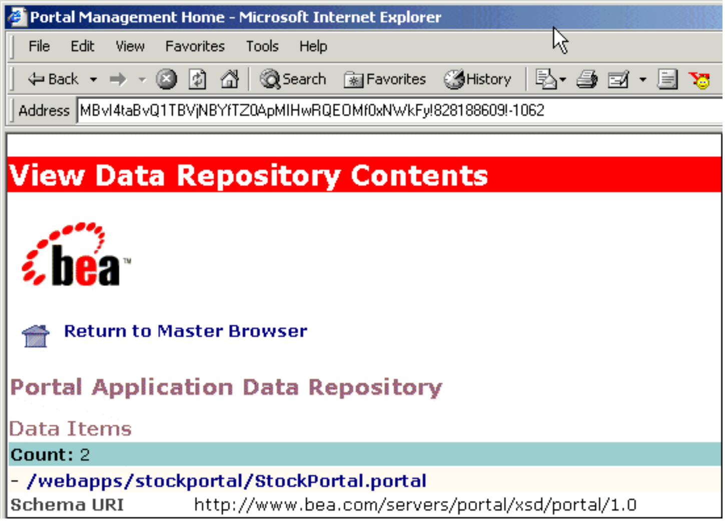
Verifying Your Synchronization

To view what applications the target server is able to see, open the data repository browser by entering this URL into your Web browser:

```
http://localhost:<port>/portalDataSync/index.html
```

This tool, shown in Figure 2-6, allows you to troubleshoot applications that seem to be deployed but do not show up on the server, for instance.

Figure 2-6 The Data Repository Browser



MBvl4taBvQ1TBVjNBYfTZ0ApMIHwRQEOMf0xNWkFy!828188609!-1062

View Data Repository Contents

Return to Master Browser

Portal Application Data Repository

Data Items

Count: 2

- /webapps/stockportal/StockPortal.portal

Schema URI http://www.bea.com/servers/portal/xsd/portal/1.0

From this browser, you can view information on the deployment descriptor for your portal Web application.

Note: For a detailed description of deployment descriptors in the WebLogic Portal platform, consult the *Security Guide*.

Delegated Administration

Delegated administration means assigning specific administrative privileges to individual administrators within a specified domain. In WebLogic Portal, these privileges can be assigned by one user as long as that user has more permissions than the assignee.

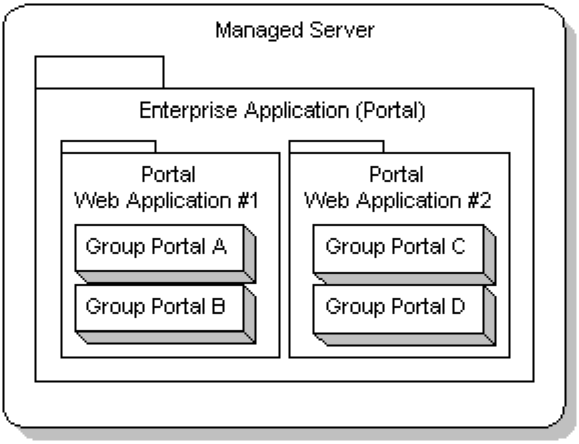
To manage the content of your portal applications at a centralized corporate office while delegating localization and design to regional offices, an elegant solution would be to create some administrative user accounts and then assign different sets of permissions to each user, based on the tasks you needed to hand out. WebLogic Portal now includes advanced delegated administration functionality to enable the creation of administrator roles with fine-grained administrative privileges.

Note: For details on the database objects used to manage these privileges, consult The RESOURCE_GROUP_ADMIN Database Table section in Chapter 7, “Portal Management Database Schema.”

Scoping Privileges

To revisit the administration model used by WebLogic Portal, the J2EE application scoping is also the basis for the three basic levels of administrator: System Administrator (SA), Portal Administrator (PA), and Group Administrator (GA). The scope of each of these roles can be explained in terms of the application scoping shown in Figure 2-7.

Figure 2-7 Scoping of Administrators



- A portal SA created within the enterprise application called Portal could perform all administrative tasks within any of the applications shown in Figure 2-7.
- A Portal PA created within portal Web application #1 could be granted privileges within that application only. (Portal PAs can be granted different sets of privileges within more than one portal Web application.)
- A Portal GA created within Group Portal C could be granted privileges within that group portal only. (Portal GAs can be granted different sets of privileges within more than one group portal.)

Visitor Entitlements

Visitor entitlements are created by associating existing entitlement segments with portal resources. The entitlement segments are created in the E-Business Control Center, and are explained in detail in the *Guide to Using the E-Business Control Center* documentation. The process of associating entitlement segments with portal resources is explained in Chapter 4, “Portal Administration Tools.”

Rule-Based Entitlements Versus Rule-Based Personalization

WebLogic Portal enables rule-based personalization. Generally, with rule-based personalization, a rules engine is used to dynamically determine whether a user is part of a segment based on profile attributes, request or session attributes, or a time element. Based on this decision, specific content may be shown to the user. A developer uses JSP tags provided with WebLogic Portal, such as a placeholder or the content selector tag, to specify where on the site the personalized content should be displayed.

On the other hand, rule-based entitlements represent a specific usage of rule-based personalization at the system level. Entitlements are applied to specific portal resources: portlets and portal pages, rather than arbitrary content and areas on the site. The entitlements are controlled completely from administration tools, and no HTML/JSP developer involvement is required.

In addition, rule-based entitlements are typically used to control access to portal content, whereas rule-based personalization is used to serve targeted content. Inevitably, there will be scenarios where the line between rule-based entitlements and rule-based personalization will be blurred. For example, rule-based entitlements may be used to show a portal page with recommended content to a segment of users.

Both rule-based personalization and rule-based entitlements rely on the same set of infrastructure components—the BEA rules engine, the user profile, and property sets.

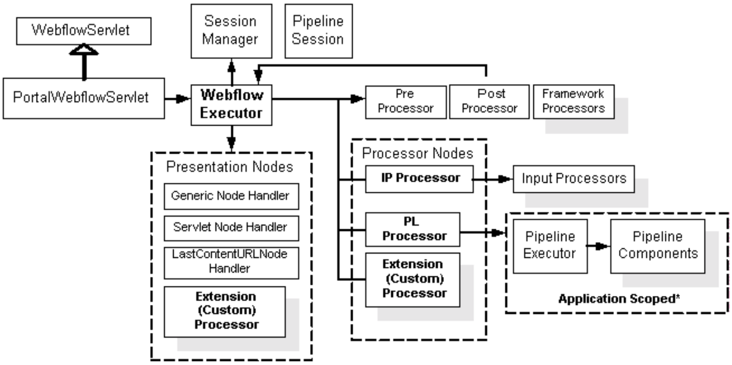
Portal Rendering

When an HTTP request for a portal arrives at an instance of WebLogic Portal, the request is sent to the `PortalWebflowServlet`, which inherits from the `WebflowServlet`. Figure 2-8 shows a high-level view of the objects used to respond to such a request. Based on the Webflow file defined for that portal, presentation nodes such as JSPs are invoked, providing the "wireframe" on which the portlets will be arranged.

Rendering Implementation

This rendering process involves determining what portlets go in what position on the page, what content objects the user is entitled to view, and what other application functionality should be exposed based on the group(s) to which this user belongs.

Figure 2-8 Portal High-Level Architecture



Note: In Figure 2-8, every object except for the Pipeline Executor and the Pipeline Components (see asterisk) are scoped to the Web application. This means that only the Pipeline Executor and Pipeline Components can be shared from one Web application to another within the same enterprise application.

What Is Rendered First?

The following factors are considered when a portal page is rendered:

- **Group Personalization**: What user group does the visitor belong to, and what group portal is this user group associated with?
- **Dynamic Personalization**: Based on entitlement segments applied to portal resources within the portal page being rendered, some portlets may be available, some may not. Also, any Commerce, Campaign or Personalization service logic is carried out here as well.

- **Visitor Customization**: If this visitor has never logged in, the default settings are applied to this page. If the visitor has already logged in to this portal and has selected Customize My Portal, then any settings such as skins, layouts, the visibility or placement of portlets on a given layout, and so on will be in effect. This is because any customization settings the visitor saves take precedence over those set using Group Personalization.

Default Rendering

Part of creating a portal using the E-Business Control Center involves selecting a layout, then setting the default placement of portlets on that layout. This setting is used when a given visitor requests that page, unless the visitor has overwritten these default settings by selecting and saving customization.

However, individual visitors can be entitled to see some portlets while being prohibited from seeing others. This poses a problem in deciding how to render a portal if the default settings are in conflict with those determined by visitor entitlements. What if, for instance, the default layout is three-column, and the center column contains a portlet to which visitor A does not have access? As previously explained, visitor A would view the portal with a huge gap running straight down the center of the page. Obviously, this is not the intended behavior.

The Auto Placer and “Homeless” Portlets

One aspect of the Portal Servlet involves efficient determination of the placement of portlets, especially when certain layout parameters are changed. It is possible for instance, for a portal visitor to modify the layout for a portal page such that some of the portlets become “homeless.”

A portlet can become homeless for a number of reasons. If default portlets are arranged on a three-column layout, but then the visitor changes the layout to two columns without rearranging the portlets, those in the center column become “homeless.” The next time that visitor goes to this page, whether within the same session or another, the auto placer mechanism responds to the presence of a homeless portlet. At this point, the auto placer’s custom logic makes certain decisions about the placement of the portlets in the center column, usually meaning the portlets are redistributed more or less equally within the layout. These decisions are saved to the database and consulted each time that page is visited.

Note: The auto placer only runs if the layout for a page has changed since the page was last saved, and if this change causes portlets to become homeless.

For information on creating custom layouts and using the placeholder JSP tag, consult “Create a Custom Layout,” in Chapter 5, “Customizing Portals and Portlets.”

Administrator Tasks

This section outlines the tasks required to administrate WebLogic Portal, and includes the following sections:

- Create Users and Groups
- Scope Administrator Users: Group or Portal?
- Create Group Portals
- Rudimentary Portal Personalization
- Replicate Sample Users in LDAP Service

Note: The sample entities used within the steps in this section are listed in Table 2-3.

Table 2-3 Sample Entities

Entity Type	Entity Names
Portal SA	TestSA1, TestSA2
Portal PA	TestPA1, TestPA2, TestPA3
Portal GA	TestGA1, TestGA2, TestGA3
Visitor	TestVisitor1, TestVisitor2
User Group	TestGroup1, TestGroup2, TestGroup3
Group Portal	GroupPortalA, GroupPortalB, GroupPortalC, GroupPortalD

Create Users and Groups

Navigate to the following URL and login as Administrator/<password>

`http://<hostname>:<port>/portalTools/index.jsp`

For the remainder of this topic, use this credential.

Go to the User Management page in the WebLogic Portal Administration Tools and click Create on the Users tab.

- Create users with the names in the first four rows of Table 2-3.
- Create user groups using the names in the fifth row of Table 2-3.

All the users you have created are automatically placed in the default "everyone" group, and granted no administrative privileges. You can leave the Visitor users alone for now. The administrative users, however, must be added to groups in order for them to have any administrative capability.

Warning: From this point on, avoid the User Management home page when performing administrative actions that can be performed in the Portal Management or Group Portal Management home pages. This page should only be used when creating new System Administrators, user groups, or when moving users into the AdminEligible user group. Instead, go into the Portal Management home page and manage users at the portal or group portal level.

Place GAs and PAs in AdminEligible Group

Go to the Group Management page in the WebLogic Portal Administration Tools and place all PAs and GAs in the AdminEligible user group. At this point, these users still have no administrator privileges, but they can now be granted privileges, as shown in Scope Administrator Users: Group or Portal?

Create SA Users

Go to the Group Management page in the User Management Home page, place TestSA1 and TestSA2 in the `SystemAdministrator` user group.

Scope Administrator Users: Group or Portal?

Once they have been placed into the AdminEligible user group, users can be granted administrative privileges at one of two levels:

- They can be made into PAs by following the steps in “Create and Delegate PAs” on page 2-20. In this case, the rights these users are given will affect all the group portals within the portal Web application in which they exist.
- They can be made into GAs by following the steps in “Create and Delegate GAs” on page 2-21. In this case, the rights these users are given will only affect the group portal(s) in which they are delegated.

Note: GAs may belong to more than one group portal. Also, GAs may be promoted to PAs by clicking on Edit Portal Administrators from within the Portal Management Home page.

Create and Delegate PAs

To turn users (that is, users already associated with the AdminEligible user group,) into PAs, take the following steps:

1. From the Portal Management Home page, select Edit Portal Administrators.
2. Click Create a New Administrator. The Add New Portal Administrator page appears.
3. Select TestPA1 from the results of the search the AdminEligible user group. The Delegate Administration page appears.
4. Click Grant under User Management, then click Save.

Repeat this procedure for the user TestPA2, but this time when you repeat step 4, click Grant under Portal Page Management, Portlet Management, and Skins Management. Click Save.

Create Group Portals

From the Portal Management Home page, create new group portals from the user group and group portal template associations as shown in Table 2-4.

Table 2-4 Creating Test Group Portals

Group Portal Display Name	User Group	Group Portal Template
GroupPortalA	TestGroup1	BEA Basic Portal
GroupPortalB	TestGroup2	BEA Basic Portal
GroupPortalC	TestGroup3	BEA Basic Portal

Note: Notice that each user group can only be associated with one group portal within a portal Web application.

Create and Delegate GAs

The users in the AdminEligible user group can now be associated with the group portals defined in Table 2-4.

1. From the Portal Management Home page, select GroupPortalA and click on the Edit Delegated Administration Settings for Group Administrators tab at the bottom of the page.
2. At the Choose Administrator page, click on Create New Administrator.
3. Select TestGA1 from the results of AdminEligible user group.
4. Click Select User. The Delegate Administration page appears.
5. Click Grant under User Management, then click Save.

Repeat this procedure for the user TestGA2, associating it with GroupPortalA, but this time when you repeat step 5, click Grant under Portal Page Management, Portlet Management, and Skins Management. Click Save.

Place Visitors in User Groups

The users in the **everyone** user group can now be associated with the user groups defined in Table 2-4. From the User Management Home page, search for the user group to which you want to add users from the Manage Groups page:

- Place TestVisitor1 into TestGroup1.
- Place TestVisitor2 into TestGroup2.
- Place TestVisitor1 and TestVisitor2 into TestGroup3.

Rudimentary Portal Personalization

To understand content personalization in WebLogic Portal, consider the following simple example:

1. Navigate to GroupPortalA. Under Attributes and Visitor Entitlements, click on Page and Portlet Attributes and Visitor Entitlements.
   - Click on the page called “Web”, set Set Page Entitlements for the Entitlement Segment called “EVERYONE” to Deny.
   - Click on the page called “Web”, set Set Page Entitlements for the Entitlement Segment called “DEVELOPER” to Grant.
2. From within GroupPortalA, Click on User Management and select TestVisitor1.
   - Select the Property Set “StockPortal” and click on “Developer (Boolean, Single, Restricted)” under the Properties Tab.
   - Set the value to True, click Save.
3. From within GroupPortalB, Click on User Management and select TestVisitor2.
   - Select the Property Set “StockPortal” and click on “Developer (Boolean, Single, Restricted)” under the Properties Tab.
   - Set the value to False, click Save.
4. Launch a new browser and navigate to the following URL:

```
http://<server>:7501/stockportal/
```

5. Login as TestVisitor1 and verify you can see all three pages: Home, Web, and Misc.

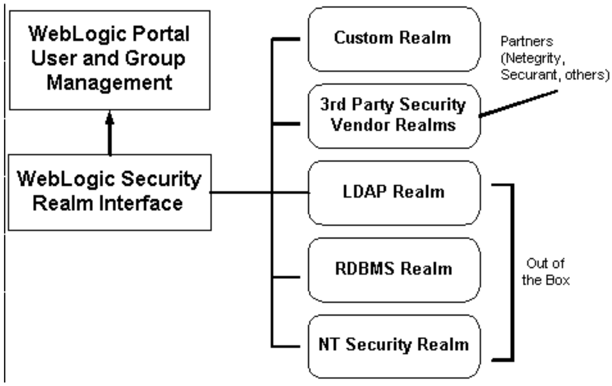
6. Logout. Login as TestVisitor2 and verify you can only see two pages: Home and Misc.

This is based on the StockPortal property set.

Replicate Sample Users in LDAP Service

The WebLogic Portal platform can use LDAP authentication as part of the Unified User Profile, but in this situation, the LDAP service is always the system of record, meaning that WebLogic Portal does not write to the LDAP directory. Figure 2-9 shows the relationship between the Unified User Profile user scheme used by WebLogic Portal, along with its relationship with LDAP and other services.

Figure 2-9 LDAP and Unified User Profile Authentication


WebLogic Portal User and Group Management
WebLogic Security Realm Interface
Custom Realm
3rd Party Security Vendor Realms
LDAP Realm
RDBMS Realm
NT Security Realm
Partners (Netegrity, Securant, others)
Out of the Box

If you use LDAP for authentication and want to use the sample accounts provided by WebLogic Portal, you must set up the accounts and user groups in your LDAP server. It is not necessary to enter the account's first and last name fields. The required fields are the username and password.

Note: For instructions on setting up LDAP services, consult the *Guide to Registering Customers and Managing Customer Services*.

1. Using your Netscape or iPlanet Directory Server console, create the following users in LDAP, as shown in Table 2-5.
2. Create the following user groups in LDAP, as shown in the left column of Table 2-5.
3. Add the users indicated in the right column of Table 2-3 to the user groups in the left column.

Table 2-5 Sample User Groups

User Group	Add Users
AdminEligible	demopa<n> and demoga<n>
DelegatedAdministrator	demopa<n> and demoga<n>
SystemAdministrator	administrator
Group1	visitor1 through visitor5
Group2	visitor6 through visitor10

Table 2-6 Sample User Accounts

Username	Note	Password
administrator		password
admin<n>	where n = 1 through 4	password
demopa<n>	where n = 1 and 2	password
demoga<n>	where n = 1 through 3	password
visitor<n>	where n = 1 through 10	password

You should now be able to work with the sample accounts installed with the product, even if your authentication system is LDAP.

3 Using the E-Business Control Center Portal Tool

The E-Business Control Center enables you to define the design, functionality and visitor attributes of portals and portlets with limited coding knowledge of Java, JavaServer Pages (JSPs) or HTML. Once you have defined the components of a portal or portlet, the definitions are communicated to a server and the new portal is available to visitors or to an administrator who may define its runtime components using the WebLogic Portal Administration Tools.

Components that can be defined via the E-Business Control Center include:

- Design components

  Includes assigning skins and layouts to portals.

- Functionality

  Includes assigning Web applications, Webflow mechanisms, and portlets.

- Visitor information and Guidelines

  Includes defining entitlement definitions and user profiles.

  Note: To use the portal design functionality in the E-Business Control Center, you must purchase the full WebLogic Portal License.

This topic includes the following sections:

- Using the E-Business Control Center

What you should know in order to make the best use of this application.

- E-Business Control Center Portal Concepts

  Basic concepts you should be familiar with when using the E-Business Control Center portal functionality.

- Getting Started

  How to get started using the E-Business Control Center.

- Portal Web Applications

  How to open a newly created portal Web application in the E-Business Control Center.

- Portals

  How to associate available Web components, such as skins, layouts, and the portlets and entitlements you have created in earlier sections, for the creation of a new portal.

- Portlets

  Step by step procedure for creating a new portlet.

- Entitlement Segments

  How to access the entitlement segment definition tool.

Using the E-Business Control Center

To effectively utilize the portal definition tools in the E-Business Control Center, you should have some knowledge of the fundamentals of building a Web portal, including a high-level understanding of JavaServer Pages (JSPs), page flows, servlets, and portlets. It is also helpful, but not necessary, to be familiar with the Portal framework, Webflow, and the WebLogic Portal Administration Tools.

E-Business Control Center Portal Concepts

The following terms are concepts often used in this document and in the E-Business Control Center.

- Portal Web Applications

  A portal Web application is a group of server-side Web resources that make up an interactive online portal application. The Web resources include Java servlets, JSPs, static documents (such as HTML documents), and applets that can be deployed in a client Web browser.

- Portals

  A Portal is the framework for the Web portal or application. Using the E-Business Control Center, you define what possible layouts, skins, pages, and portlets that will be available for the portals. Once these definitions are synchronized with the server, the new portal can be accessed by a visitor or can be edited by a designated user of the WebLogic Portal Administration Tools.

- Portlets

  A portlet is a Web component, often a small software application, that resides within a portal. It can be something as simple as a JSP `include` statement containing text, or as robust as a Web application, such as e-mail or a stock ticker. A well-designed portlet can be reused in different portals.

- Defaults

  When you define portal attributes, you set defaults to determine which of your selections will be seen initially at the front end. These defaults can also be set, or the EBCC defaults can be reset, using the WebLogic Portal Administration Tools portal at runtime.

- Entitlement Segments

  An entitlement segment is a visitor group based on common characteristics that allow a member of the group to view certain aspects of a portal. For example, if you create a portal that provides information about upcoming city council elections in Los Angeles, you might want to define an entitlement group for that portlet that consists of visitors who live in Los Angeles county and are of voting age.

Meeting an entitlement segment's criteria may also give the user certain privileges in a portal. In the above example, any visitor that lives in Los Angeles county and is of voting age might be given the ability to edit the presentation or color scheme of their portal.

You use the E-Business Control Center to create entitlement segment definitions. These segments are associated with pages and portals in the WebLogic Portal Administration Tools.

- Synchronization

  Once you have defined the portal attributes and defaults, you use the synchronization functionality to post this information to a server. Once you synchronize, the portal is generated or updated and becomes available to visitors and designated users of the WebLogic Portal Administration Tools. While other tools in the E-Business Control Center require a server connection, you can use the Portal tool offline until you are ready to synchronize to the server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

- Webflow and Pipeline

  Webflow allows the user to create a flow of actions and pages for a Web application. You use the E-Business Control Center to assign Webflow mechanisms to portals and portlets. For general information about Webflow and Pipeline, see the *Guide to Managing Presentation and Business Logic: Using Webflow and Pipeline* documentation. For information specific to portal Webflow, see the Webflow topics in “Customizing Portals and Portlets.”

- WebLogic Portal Administration Tools

  Once you have determined the desired Web components for the portal, the actual instance of the portal can be configured using the WebLogic Portal Administration Tools. For more information, see the *Getting Started with Portals and Portlets* documentation.

  Note: You cannot edit or delete the default `Administrator` and `everyone` groups in the WebLogic Portal Administration Tools.

Getting Started

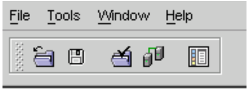
To start the E-Business Control Center:

Windows: Selecting Start → Programs → BEA WebLogic E-Business Platform → BEA E-Business Control Center 4.0 → E-Business Control Center.

UNIX/Linux: Run `$ EBCC_HOME/bin/ebcc.sh`

If you don't see the Explorer window along the left side of the E-Business Control Center, click the Open Explorer button in the main toolbar, as shown in Figure 3-1.

Figure 3-1 Explorer Window



- For information about managing E-Business Control Center files, see "Working With Files" in the *Guide to Using the E-Business Control Center* documentation.

Portal Web Applications

A portal Web application is a group of Web resources or components that are used in a portal application. The Web resources include Java servlets, JSPs, static documents (such as HTML documents), and applets that can be deployed in a client Web browser. Before you create a new portal, make sure resources exist on the WebLogic Portal server.

For information about creating a new Web application, see the *Getting Started with Portals and Portlets* documentation.

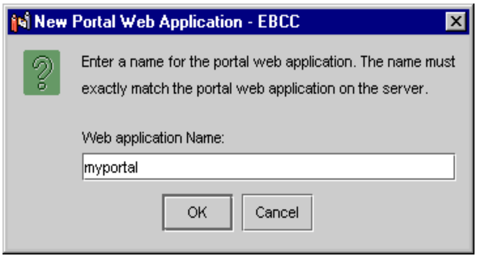
Importing Data for a New Portal Web Application

Use the following procedure to create a new portal Web application that exists on a server.

1. In the E-Business Control Center toolbar, select File → New → Portal Web Application.

   The New Portal Web Application window appears (Figure 3-2).

Figure 3-2 New Portal Web Application Window


New Portal Web Application - EBCC

Enter a name for the portal web application. The name must exactly match the portal web application on the server.

Web application Name: myportal

1. In the Web Application Name field, type the name of the portal Web application. You must type this exactly as it exists on the server.

2. Click OK. The portal Web application is now available for use and editing in the E-Business Control Center.

Portals

When you create a new portal definition, you associate metadata for elements such as skins, portlets, pages, layouts, and Webflow mechanisms. This section covers the following information:

- Designating a New Portal
  - Defining General Portal Elements
  - Creating Portal Pages
  - Setting Portal Defaults
- Editing a Portal
- Deleting a Portal

Designating a New Portal

The first step in creating a new portal is selecting the design elements, including banners, skins, headers, content, layout, as well as user profile information.

Defining General Portal Elements

To define general portal elements, use the following procedure:

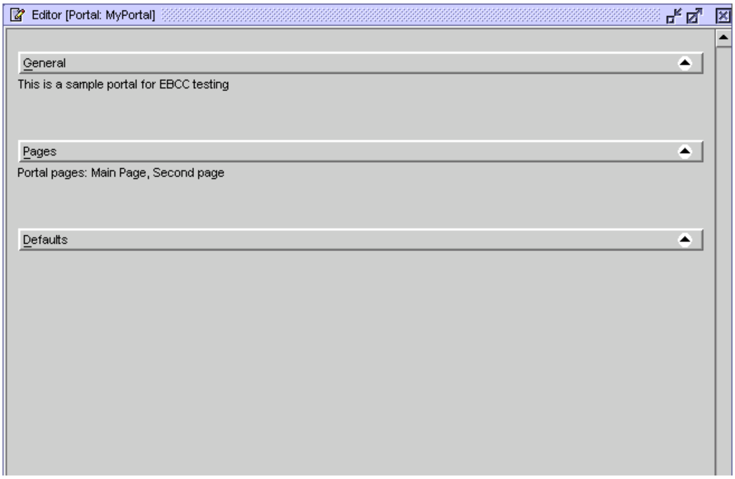
1. In the main toolbar, select File → New → Presentation → Portal.

   The Select Web Application window appears.

2. Select the desired Web application from the available list and click OK. For information about adding making new portal Web applications available for use via the E-Business Control Center, see “Portal Web Applications” on page 3-5.

   The Portal Editor appears (Figure 3-3).

Figure 3-3 Portal Editor

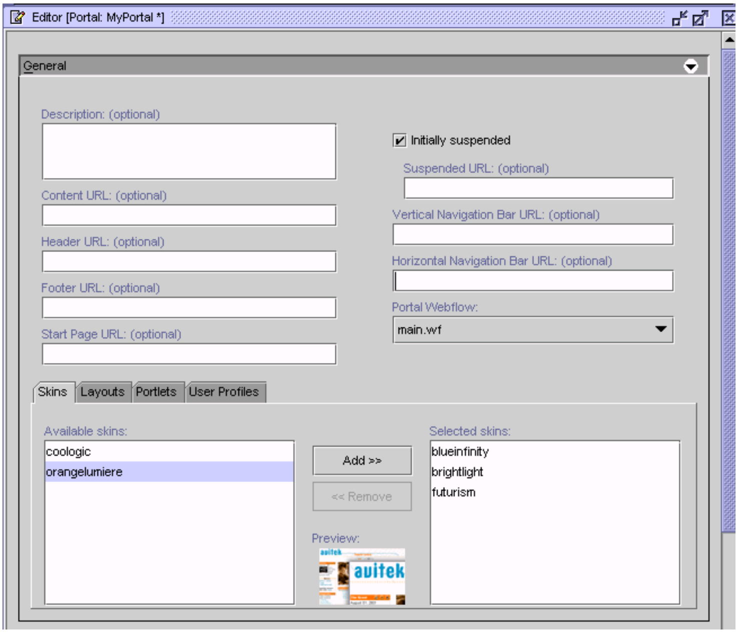


This is a sample portal for EBCC testing

Portal pages: Main Page, Second page

3. Click on the General bar to display the General portal pane (Figure 3-4).

Figure 3-4 General Portal Pane



Editor [Portal: MyPortal *]

General

Description: (optional)

Content URL: (optional)

Header URL: (optional)

Footer URL: (optional)

Start Page URL: (optional)

☑ Initially suspended

Suspended URL: (optional)

Vertical Navigation Bar URL: (optional)

Horizontal Navigation Bar URL: (optional)

Portal Webflow: main.wf

Available skins: coologic, orangelumiere

Selected skins: blueinfinity, brightlight, futurism

4. Provide the following information:

   a. Description (optional): Provide a description for the portal you are creating.

   b. Content URL: Provide a URL for the content that will appear in the portal.

   Note: The URLs should be the file system directory path to the JSPs that are located on the WebLogic Portal server. When you synchronize this data, the appropriate JSP is inserted as an `include` to the portlet. The location of these directories are determined by your organization.

   c. Header URL (optional): Provide a URL to the JSP that will be used as the portlet's header content.

   d. Footer URL (optional): Provide a URL to the JSP that will be used as the portlet's footer content.

   e. Vertical Navigation Bar URL: Provide a URL if you wish to display a vertical navigation bar for your portal.

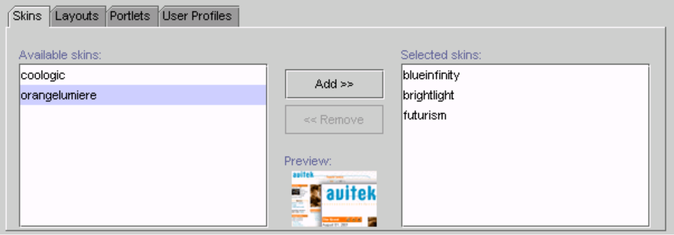
f. Horizontal Navigation Bar URL: Provide a URL if you wish to display a horizontal navigation bar for your portal.

g. Portal Webflow: When you create a Portal, you must associate the new Portal with a predefined Webflow; in most cases, the `portal` Webflow. For more information about the Webflow mechanism as it relates to portals and portlets, see "Customizing Portals and Portlets."

5. Click on the Skins tab at the bottom of the panel to select the possible skins for the portal (Figure 3-5).

   Note: When you select skins and layouts for your portal, you select from a library of metadata representing selected portal components. The E-Business Control Center ships with an existing library which can be modified or supplemented. For information about editing or adding skins and layouts, see "Developing Portals and Portlets" in the *Getting Started with Portals and Portlets* documentation.

Figure 3-5 Selecting Skins



Available skins:
coologic
orangelumiere

Selected skins:
blueinfinity
brightlight
futurism

   Use the Add and Remove buttons to select the desired skins. A thumbnail image will appear when you select a skin. You can select one to many skins.

6. Click on the Layouts tab to select the possible layouts for the portal.

   Use the Add/Remove buttons to select the desired layout(s); The preview thumbnail allows you to view the selected layout. You can select one to many layouts.

7. Click on the Portlets tab to select the portlets that will be included in the new portal.

   Use the Add/Remove buttons to select the desired portlets. If you select multiple portlets for your portal, visitors can select from these layouts when personalizing their portal.

8. Click the User Profiles tab to select the user profile set that will be enabled to view or use the portal.

   Use the Add/Remove buttons to select the desired user profiles.

   A **user profile** is a collection of information that is gathered, saved, and updated as customers visit your Web site, or is collected from other offline sources. When you select a user profile, you determine which information can be used or collected. These user profiles are defined in the Entitlement section of the E-Business Control Center. For more information, see “Using Customer Segments to Target High Value” in the *Guide to Using the E-Business Control Center*.

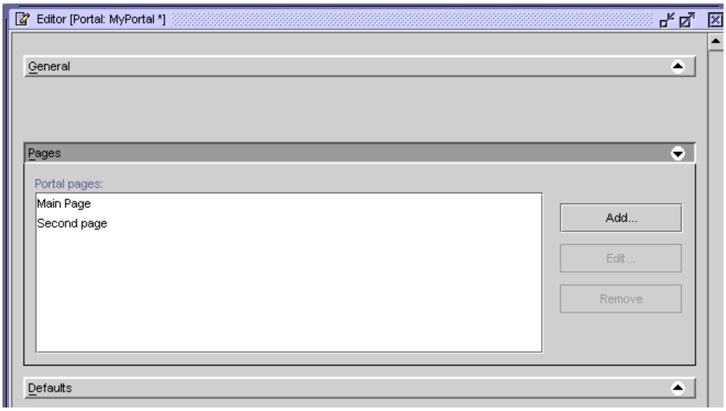
9. Click Save to save your changes. These changes will be available to the visitor or for modification via the WebLogic Portal Administration Tools once you have synchronized your changes to the appropriate server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Creating Portal Pages

When you create portal pages, you create or associate each page that can or will be used in the portal flow.

1. Click the Pages bar of the Portal Editor panel to expand the Pages pane (Figure 3-6).

Figure 3-6 Creating Portal Pages

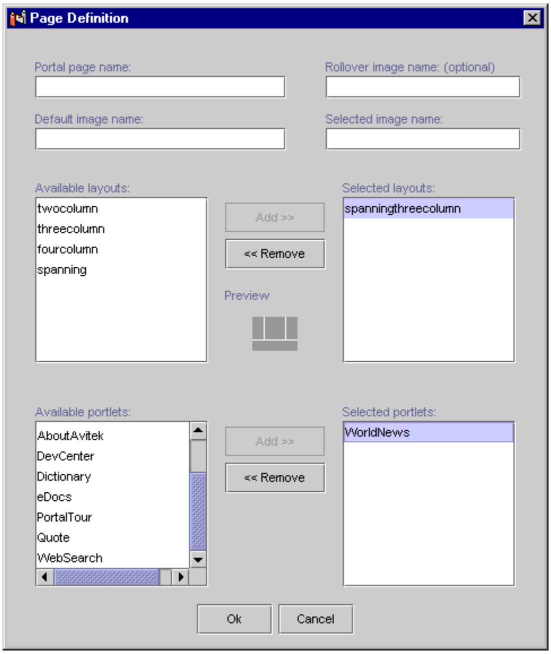


Portal pages:
Main Page
Second page

2. Click Add.

   The Page Definition editor displays (Figure 3-7).

Figure 3-7 Page Definition Editor



Available layouts:
twocolumn
threecolumn
fourcolumn
spanning

Selected layouts:
spanningthreecolumn

Available portlets:
AboutAvitek
DevCenter
Dictionary
eDocs
PortalTour
Quote
WebSearch

Selected portlets:
WorldNews

3. Provide a portal page name.

4. Supply the path to the graphics files that will be used as the default and selected images. A path to the rollover image is optional.

5. Select at least one page layout: use the Add/Remove buttons to select the desired layout. The page can have one to many associated layouts.

   Pages draw from the portal's layout and portlet lists to get their available lists. This means that the layouts in this window are limited to layouts that have been added to the portal. To make more layouts available for your pages, see "Designating a New Portal" on page 3-6.

6. Select at least one portlet: Use the Add/Remove buttons to select the desired portlet(s). The Portal page can have one to many portlets.

7. When you are finished, click OK.

   The new page appears in the Portal Pages window.

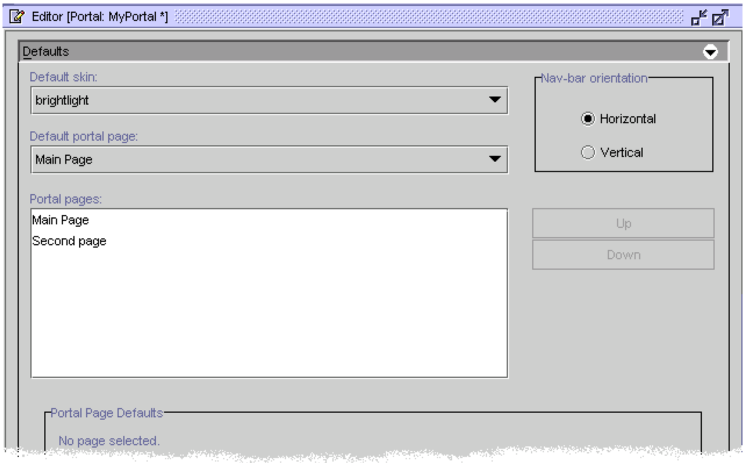
8. Click Save to save your changes. These changes will be available to the visitor or for modification via the WebLogic Portal Administration Tools once you have synchronized your changes to the appropriate server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Setting Portal Defaults

Before synchronizing your new portal definition to the server, you must set defaults. Setting defaults for your portal ensures that appropriate pages are displayed until an administrator has enabled the applications.

1. Click the Default Bar in the Portal Editor to expand the Defaults pane (Figure 3-8).

Figure 3-8 Setting Portal Defaults



Editor [Portal: MyPortal *]

Defaults

Default skin: brightlight

Default portal page: Main Page

Portal pages:
Main Page
Second page

Nav-bar orientation
● Horizontal
○ Vertical

Portal Page Defaults
No page selected.

2. Set a default display name for your portal.

3. Set a default skin: Use the pull-down menu to select one default from a list of your previously selected skins.

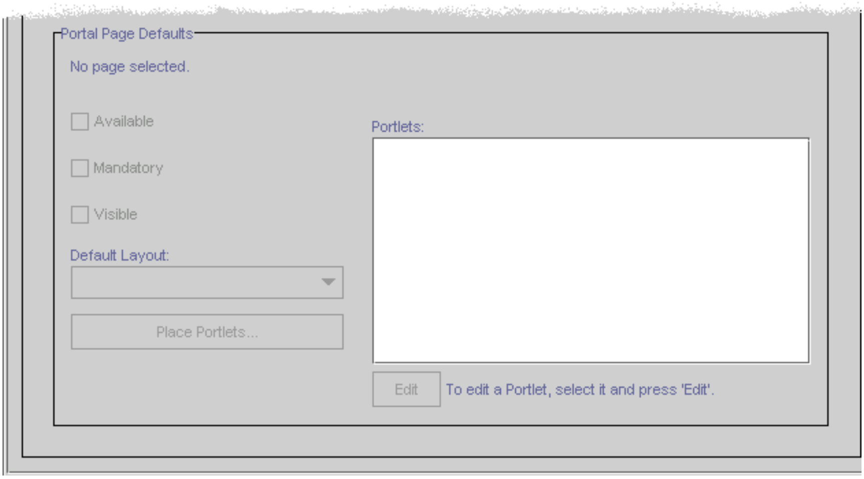
4. Set default pages: Use the pull-down menu to select the desired pages from a list of your previously selected pages.

5. Use the Up/Down buttons to set the order that the selected page tabs will appear in the portal.

6. Select either vertical or horizontal orientation for the Navigation bar.

7. Set portal page defaults (Figure 3-9).

Figure 3-9 Setting Portal Page Defaults



No page selected.

To edit a Portlet, select it and press 'Edit'.

a. Check the appropriate boxes if you wish the page to be Available, Mandatory, and/or Visible.

b. Select a default layout.

c. Click the Place Portlets button, the Page Layout window appears.

   To add a portlet to the layout template, highlight the desired portlet, click the radio button in the desired section, then click the Add button. The portlet will be added to the desired section of the layout.

   To remove a portlet from the layout template, click on the desired portlet in the template, then click the Remove button. The portlet will be removed from the template and added back to the portlet list.

   To move a portlet, click on the desired portlet in the template, then click the Up or Down buttons.

   When you have finished arranging your portlets in the template, click OK.

d. You can also edit a portlet's default settings from the Portal Page Defaults section. Select the desired portlet in the Portlets window, then click the “Edit” button to launch a Portlet Defaults editor. For a detailed description of each default setting, see “Editing a Portlet” on page 3-19.

8. Click Save to save your changes. These changes will be available to the visitor or for modification via the WebLogic Portal Administration Tools once you have synchronized your changes to the appropriate server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Editing a Portal

1. In the E-Business Control Center Explorer window, click the Presentation tab, then the Portal icon.

   A list of existing portals will appear.

2. Click on the portal you wish to edit.

   The Portal Editor will display the information for that portlet.

3. You may edit any of the fields in Portal Editor. For more information about portal attributes that can be modified in the editor, see the previous section of this topic.

4. Click Save to save your changes. These changes will be available to the visitor or for modification via the WebLogic Portal Administration Tools once you have synchronized your changes to the appropriate server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Deleting a Portal

1. In the E-Business Control Center Explorer window, click the Presentation tab, then the Portal icon.

   A list of existing portals appears.

2. Click on the portal you wish to delete and click the Delete button.

   The portal is removed from the explorer window.

3. When you synchronize to the server, the portal will be deleted from the server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Portlets

Portlets reside within a portal application and can be anything from a robust Web application (such as e-mail) to a simple data view (for example, a link to a weather page). You can define portlets and add them to a portal at any time during or after the portal development process.

When you create a new portlet definition, you associate metadata for elements such as title bars, headers, helpfiles, and Webflow mechanisms. This section covers the following information:

- Defining a New Portlet
- Editing a Portlet
- Deleting a Portlet

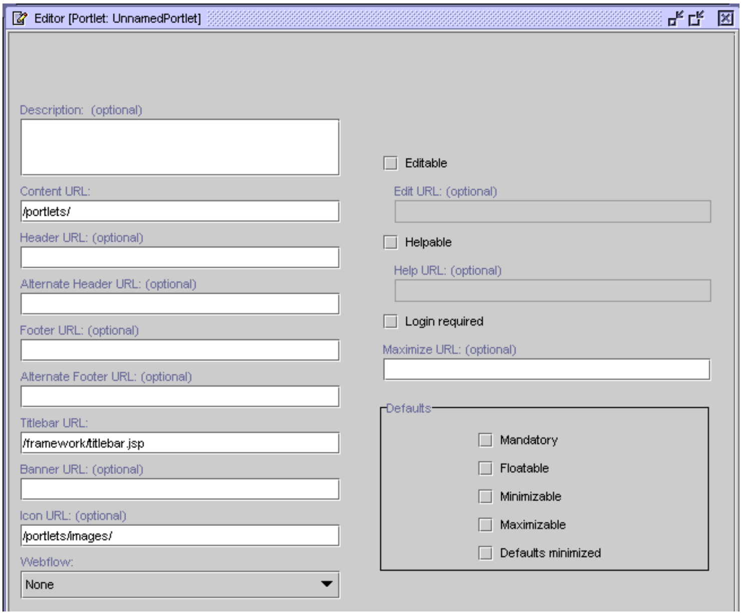
Defining a New Portlet

Use the following procedure to create a new portlet definition.

1. In the E-Business Control Center toolbar, select File → New → Presentation → Portlet.

   The Portlet Editor appears (Figure 3-10).

Figure 3-10 Portlet Editor



Editor [Portlet: UnnamedPortlet]

Description: (optional)

Content URL: /portlets/

Header URL: (optional)

Alternate Header URL: (optional)

Footer URL: (optional)

Alternate Footer URL: (optional)

Titlebar URL: /framework/titlebar.jsp

Banner URL: (optional)

Icon URL: (optional) /portlets/images/

Webflow: None

□ Editable

Edit URL: (optional)

□ Helpable

Help URL: (optional)

□ Login required

Maximize URL: (optional)

Defaults

□ Mandatory

□ Floatable

□ Minimizable

□ Maximizable

□ Defaults minimized

2. In the Portlet Editor, provide the following information:

   a. Description (optional): Provide a description for the portlet you are creating.

   b. Content URL: Provide a URL for the main body of content that will appear in the portlet.

   Note: The URLs should be the file system directory path to the JSPs that are located on the WebLogic Portal server. When you synchronize this data, the appropriate JSP is inserted as an `include` to the portlet. The location of these directories are determined by your organization.

   c. Header URL/Alternate Header URL (optional): Provide a URL for the JSP that will serve as an `include` for the portlet header; You can also provide an alternative header in the event that the primary header is missing or corrupt at runtime.

   d. Footer URL/Alternate Footer URL (optional): Provide a URL for the footer. If desired, provide an alternate footer.

   e. Titlebar URL: Provide a URL for the titlebar.

f. Banner URL (optional): Provide a URL for a banner, if you wish one to appear in your portlet.

g. Icon URL (optional): Provide a path to an icon that will represent the portlet.

h. Portlet Webflow (optional): Select a defined Webflow if your portlet requires Webflow functionality. Although portlets do not require a Webflow association, it is highly recommended. For more information, see "Customizing Portals and Portlets."

i. Editable checkbox/Edit URL (optional): If your portlet enables editing by visitors, check the Editable box and provide the URL for the JSP page they will use to personalize the portlet.

j. Helpable checkbox/Help URL (optional): If the portlet is accompanied by Online Help, check the box and provide a path to the Help files.

k. Login Required (optional): Check this box if the visitor must login with a password to access the portlet.

l. Maximize URL (optional): If the portlet is maximizable, provide the URL for the maximized JSP page.

3. Determine the portlet defaults. Default selections determine how the portlet will appear in a browser if it has not been configured using the WebLogic Portal Administration Tools.

   a. Mandatory: Mandates that the visitor will always see this portlet; That is, they cannot delete it from their personalized portal.

   b. Floatable: If this box is checked, the portlet will appear in a popup window.

   c. Minimizable: Allows the visitor to minimize the portlet. If this box is not checked, the portlet cannot be minimized within the portal.

   Note: Portlet defaults can be set differently via the portal editor and override the original defaults for that instance of the portlet. For more information, see "Setting Portal Defaults" on page 3-13.

4. Click Save to save your changes. These changes will be available to the visitor or for modification via the WebLogic Portal Administration Tools once you have synchronized your changes to the appropriate server. See "Synchronizing Application Data" in the *Deployment Guide* for more information.

Editing a Portlet

To edit a portlet:

1. In the E-Business Control Center Explorer window, click the Presentation tab, then click the Portlet icon.

   A list of existing portlets appears in the E-Business Control Center Explorer window.

2. Click the portlet you wish to edit.

   The Portlet Editor displays the information for that portlet.

3. You may edit any of the fields in the Portlet Editor.

4. Click Save to save your changes. These changes will be available for use by the WebLogic Portal Administration Tools when you have synchronized your changes to the appropriate server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Deleting a Portlet

1. In the E-Business Control Center Explorer window, click the Presentation tab, then click the Portlet icon.

   A list of existing portlets appears.

2. Select the portlet you wish to delete and click the Delete button.

   The portal is removed from the Explorer window.

3. When you synchronize to the server, the deletion will be updated in the server. See “Synchronizing Application Data” in the *Deployment Guide* for more information.

Entitlement Segments

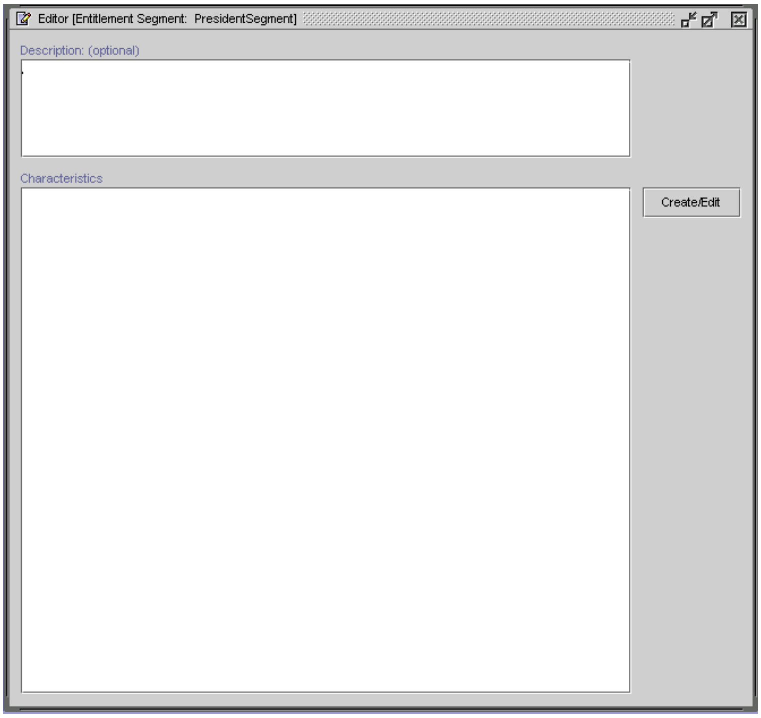
The E-Business Control Center Portal Tools allow you to create multiple entitlement segment definitions with multiple conditions and values. You use the E-Business Control Center to create entitlement segment definitions. These segments are associated with pages and portals in the WebLogic Portal Administration Tools.

Accessing the Entitlement Segment Definition Tool

In the main toolbar, select File → New → Presentation → Entitlement Segment.

The Entitlement Editor appears (Figure 3-11). Defining entitlement rules involves setting up conditions and queries. See “Building Conditions and Queries in the BEA E-Business Control Center” in the *Guide to Using the E-Business Control Center*.

Figure 3-11 Entitlement Editor



CHAPTER

4 Portal Administration Tools

This topic contains instructions on using the WebLogic Portal Administration Tools to manage resources for Portal Web Applications that are already deployed on a server.

This topic contains the following sections:

- About the Portal Administration Tools
  - Getting Started with the Portal Administration Tools
- User Creation and Management
  - Creating Users
  - Creating a System Administrator
  - Creating AdminEligible Members
  - Creating Portal Administrators
  - Creating Group Administrators
  - Editing Delegated Administration Properties for Group Administrators
  - Managing Existing Users
  - Managing Existing User Groups
- Portal Management
  - Creating a New Group Portal
  - Modifying the Appearance and Content of Existing Portals

About the Portal Administration Tools

The WebLogic Portal Administration Tools allow you to:

- Create users and assign them to groups with specific authority:
  - You can create new users and place them in a group that is associated with a portal application as a visitor group. Those visitors, depending on entitlement attributes, can view, modify, or remove pages, skins, and layouts in their personalized view of the portal.
  - You can create users and give them certain administrative abilities in the WebLogic Portal Administration Tools by placing them in role-based groups associated with the portal and specifying those administrative tasks.
- Manage runtime attributes of portals, pages and portlets. Although portals, pages and portlets are defined in the E-Business Control Center, you can modify certain runtime attributes:
  - Set portal page layout and skin defaults.
  - Change the order that pages appear in a group portal.
  - Create group portals by associating visitor user groups with a portal.

Getting Started with the Portal Administration Tools

Use the following procedure to log in:

1. Start the WebLogic Portal Server. For instructions, see the *WebLogic Portal Architectural Overview* documentation.
2. In your Start menu, select Programs→BEA WebLogic E-Business Platform→WebLogic Portal 4.0→Start Administration Tools.

   or

   Open a browser and type

   `http://server:port/portalTools/index.jsp`

   Note: Using the out of the box application, the server:port is localhost:7501

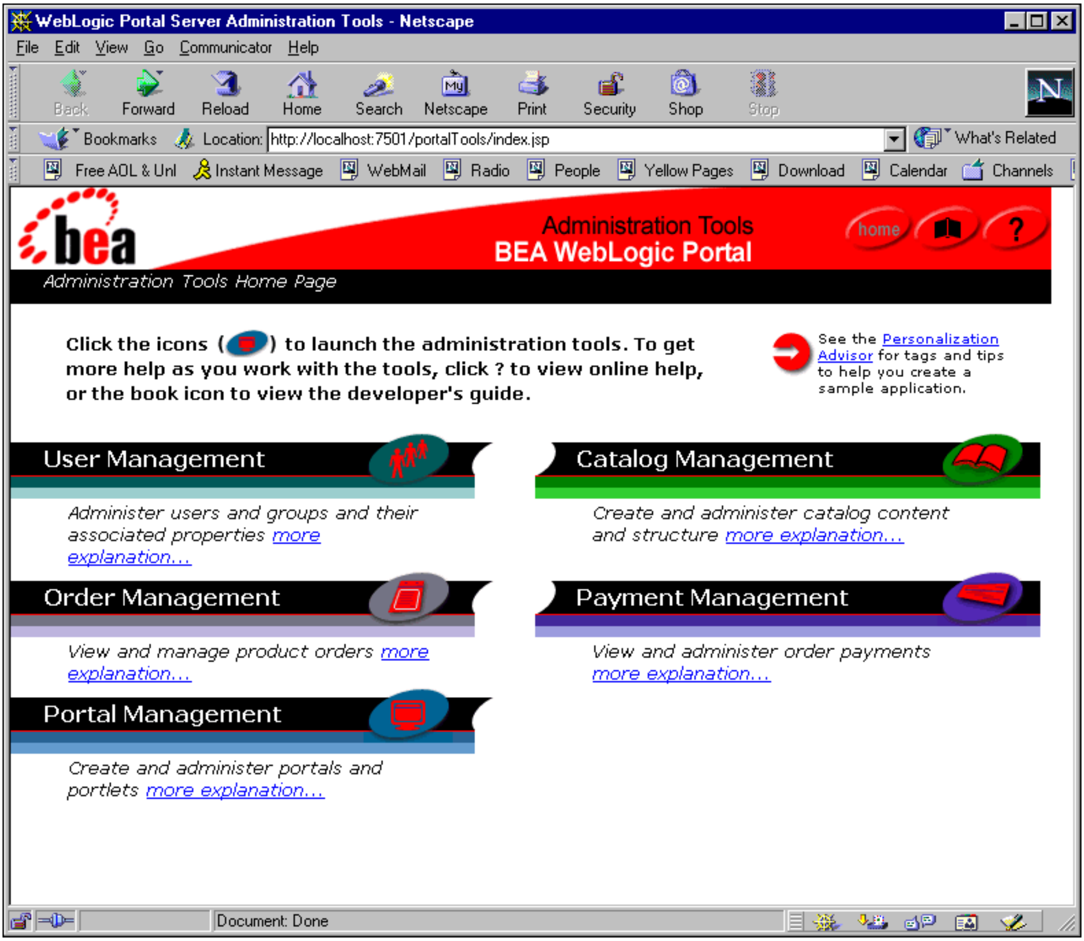
3. Log in using the user name and password provided by your administrator, or use the initial login:

   - User name: administrator
   - Password: password

   Note: This user name/password combination grants System Administrator (SA) authority and should be disabled once the appropriate administrators have been created.

4. The WebLogic Portal Administration Tools home page appears (Figure 4-1).

Figure 4-1 Administration Tools Home Page for System Administrators


WebLogic Portal Server Administration Tools - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks Location: http://localhost:7501/portalTools/index.jsp What's Related
Free AOL & Unl Instant Message WebMail Radio People Yellow Pages Download Calendar Channels
bea
Administration Tools
BEA WebLogic Portal
home
?
Administration Tools Home Page
Document: Done

Click the icons ( ) to launch the administration tools. To get more help as you work with the tools, click ? to view online help, or the book icon to view the developer's guide.

See the Personalization Advisor for tags and tips to help you create a sample application.

User Management

Administer users and groups and their associated properties more explanation...

Catalog Management

Create and administer catalog content and structure more explanation...

Order Management

View and manage product orders more explanation...

Payment Management

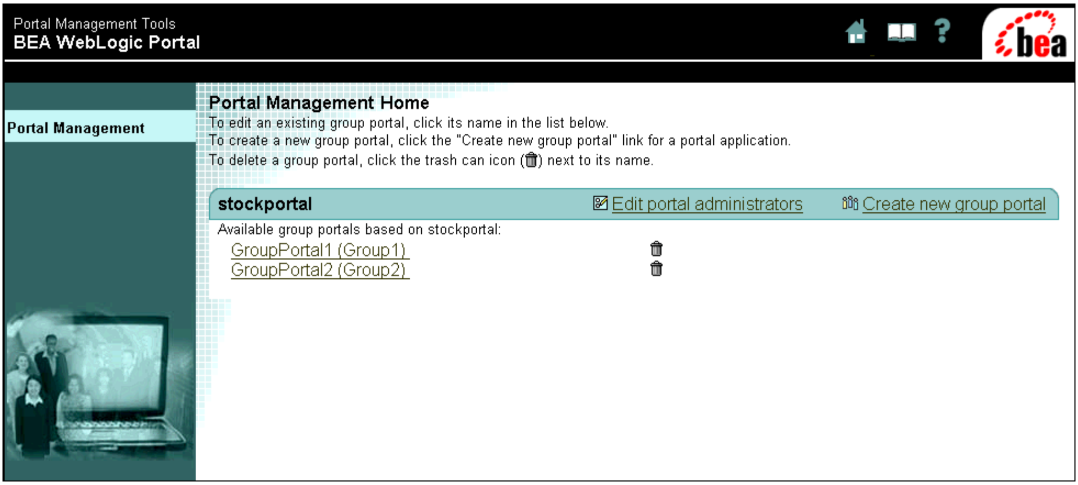
View and administer order payments more explanation...

Portal Management

Create and administer portals and portlets more explanation...

5. All procedures in this section originate from the WebLogic Portal Administration Tools home page. Users without System Administrator (SA) authority will automatically begin in the Portal Management Home page (Figure 4-2), which is the second page in each procedure. Clicking on the Home icon at the top of any page returns you to your respective home page.

Figure 4-2 Portal Management Home page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Portal Management Home

To edit an existing group portal, click its name in the list below.
To create a new group portal, click the "Create new group portal" link for a portal application.
To delete a group portal, click the trash can icon (🗑) next to its name.

stockportal — Edit portal administrators — Create new group portal

Available group portals based on stockportal:

GroupPortal1 (Group1) 🗑
GroupPortal2 (Group2) 🗑

6. Depending on your level of administrative authority, you should begin at the following sections of this chapter:
   - If you are a System Administrator, begin with “Creating Users” on page 4-6.
   - If you are a Portal Administrator, begin with “Creating Portal Administrators” on page 4-10.
   - If you are a Group Administrator, begin with “Creating Group Administrators” on page 4-15.

User Creation and Management

Using the WebLogic Portal Administration Tools, you can create a number of users with different authority. By assigning a person to a role-based group, you give them the authority that is assigned to that group. You can create the following types of users:

- Users: Users are individuals that exist in the system with user names and passwords. Only System Administrators can create users. Users are not part of a particular group.

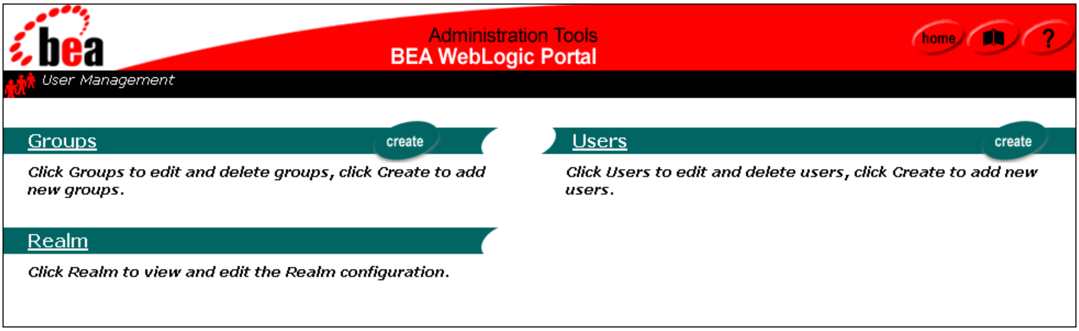
- System Administrators (SAs): SAs are the superusers of the tools and can manage any aspect of any portal. They can only be created by other SAs.
- AdminEligible users: AdminEligible group members are eligible for administrative abilities. Users can be added to the AdminEligible group by SAs.
- Portal Administrators (PAs): PAs are AdminEligible users that have been assigned administrative authority for a particular portal application. PAs can be created by SAs and other PAs.
- Group Administrators (GAs): GAs are AdminEligible users who are given limited administrative authority for specific group portals. GAs can be created by SAs, PAs, and other GAs.

Creating Users

In order to assign someone administrative abilities of any kind, they must first be present in the system as a user. A user can be an administrator in the system or a visitor to a portal. Adding someone as a user gives them a user name and a password, and enables them to be selected for addition to other role-based user groups associated with portals.

1. Click the User Management icon in the home page. The User Management home page appears (Figure 4-3).

Figure 4-3 User Management Home Page



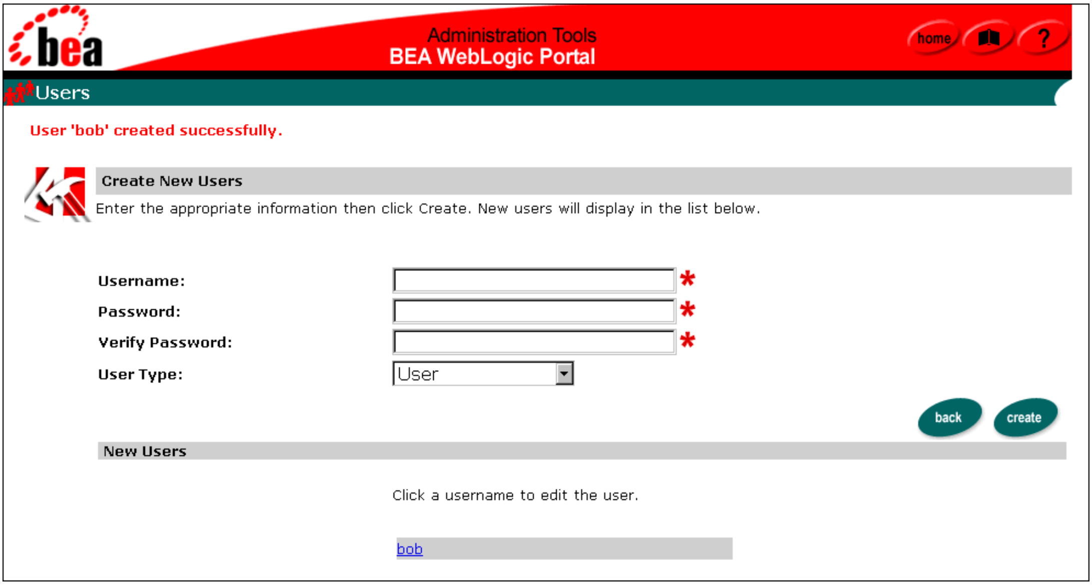
Click Groups to edit and delete groups, click Create to add new groups.

Click Users to edit and delete users, click Create to add new users.

Click Realm to view and edit the Realm configuration.

2. Click the Create icon in the Users bar. The Create New Users page appears (Figure 4-4).

Figure 4-4 Create New Users page



User 'bob' created successfully.

Create New Users

Enter the appropriate information then click Create. New users will display in the list below.

Username:	*
Password:	*
Verify Password:	*
User Type:	User

New Users

Click a username to edit the user.

bob

3. In the Create New User page, enter the following information:
   a. Name
   b. Password
   c. Confirm password
   d. User type
4. Click Create. The new user will appear at the bottom of the page.

Editing the Newly Created User

Use this procedure to assign property sets to the new user. For more information about property sets, see the *Personalization Server Tour* documentation:

1. In the Create New Users page, click the name of the new user. A user management page for your new user appears.
2. Use the drop-down menu to select a property set for the user then click Search.

3. Click Finished when you are done.

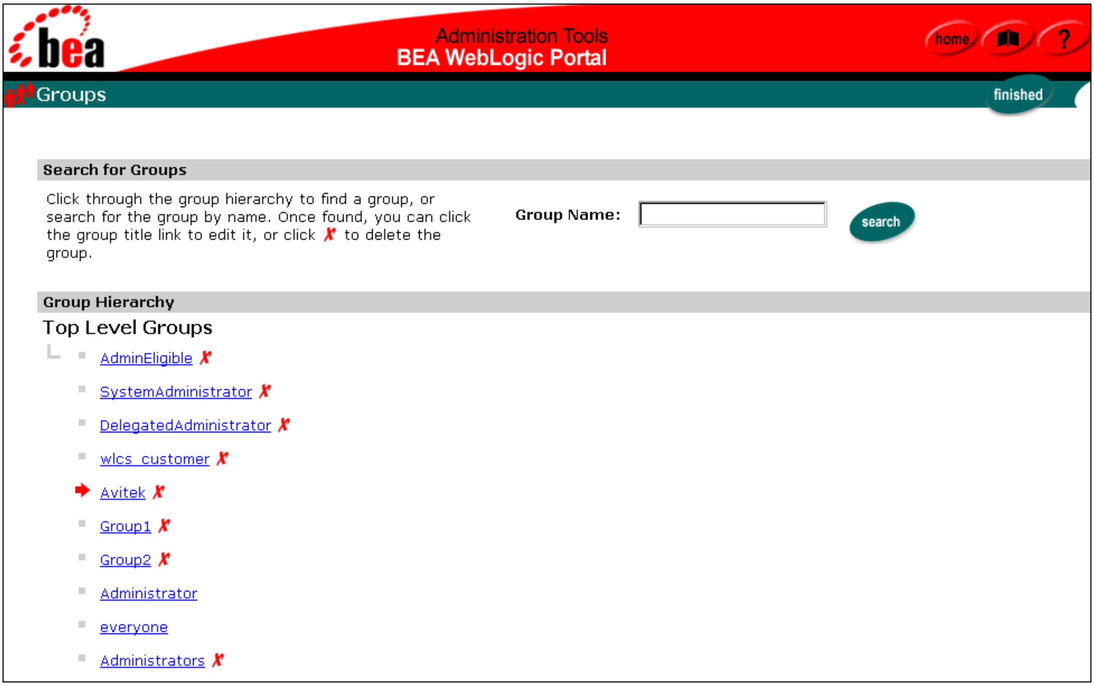
Creating a System Administrator

In order to assign someone to a System Administrator group, they must already be created as a user. To create a new user, see “Creating Users” on page 4-6.

System administrators (SAs) can be created by other SAs. You create an SA by adding a user to the system administrator group. Use the following procedure to add an SA.

1. Click the User Management icon in the Administration Tools Home page. The User Management page appears.
2. Click the Groups icon in the User Management page. The Group Hierarchy page appears (Figure 4-5).

Figure 4-5 Group Hierarchy page



Click through the group hierarchy to find a group, or search for the group by name. Once found, you can click the group title link to edit it, or click ✗ to delete the group.

Top Level Groups

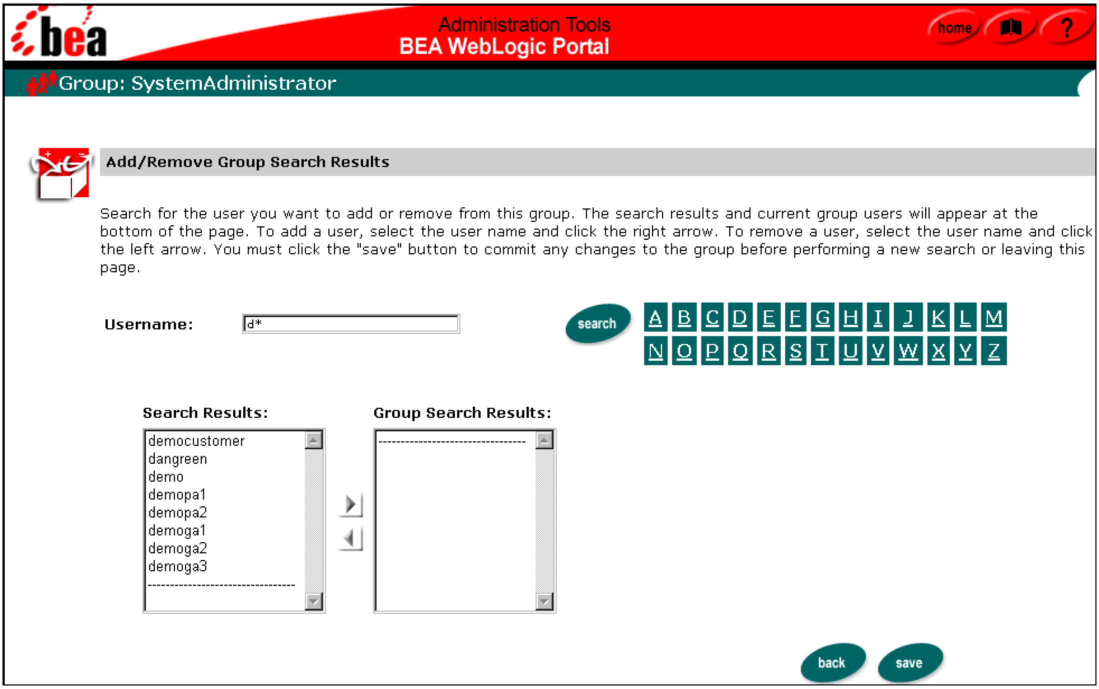
- AdminEligible ✗
- SystemAdministrator ✗
- DelegatedAdministrator ✗
- wlcs_customer ✗
- Avitek ✗
- Group1 ✗
- Group2 ✗
- Administrator
- everyone
- Administrators ✗

3. In the Group Hierarchy page, click “SystemAdministrator”.

4. The Edit Groups page displays information for the SystemAdministrator group.

5. Click the add (+/-) button. The Add/Remove page appears.

Figure 4-6 Add/Remove page



Group: SystemAdministrator

Add/Remove Group Search Results

Search for the user you want to add or remove from this group. The search results and current group users will appear at the bottom of the page. To add a user, select the user name and click the right arrow. To remove a user, select the user name and click the left arrow. You must click the "save" button to commit any changes to the group before performing a new search or leaving this page.

Username: d*

Search Results:

democustomer
dangreen
demo
demopa1
demopa2
demoga1
demoga2
demoga3

Group Search Results:

----

6. Search for the user you want to add to the SystemAdministrator group.

7. Highlight the desired user and click the right arrow to add them to the Group Search Results list.

8. Click Save. The selected user is added to the System Administrator group, granting that user all SA privileges.

Creating AdminEligible Members

Before you can give users portal administration or group administration authority, you must first add them to the AdminEligible group. AdminEligible members must first be created as users and can only be added to the AdminEligible group by SAs.

1. Click the User Management icon in the home page. The User Management page appears.
2. Click the Groups icon in the User Management page. The Group Hierarchy page appears.
3. In the Group Hierarchy page, click the AdminEligible link.
4. Click the add/remove icon (+/-) in the AdminEligible page.
5. Search for the user you want to add to the AdminEligible group.
6. Highlight the desired user and click the right arrow to add them to the Group Search Results list.
7. Click Save. The selected user is added to the AdminEligible group, making the user eligible for portal and group portal administration authority. To assign administrative authority, see “Creating Portal Administrators” on page 4-10 and “Creating Group Administrators” on page 4-15.

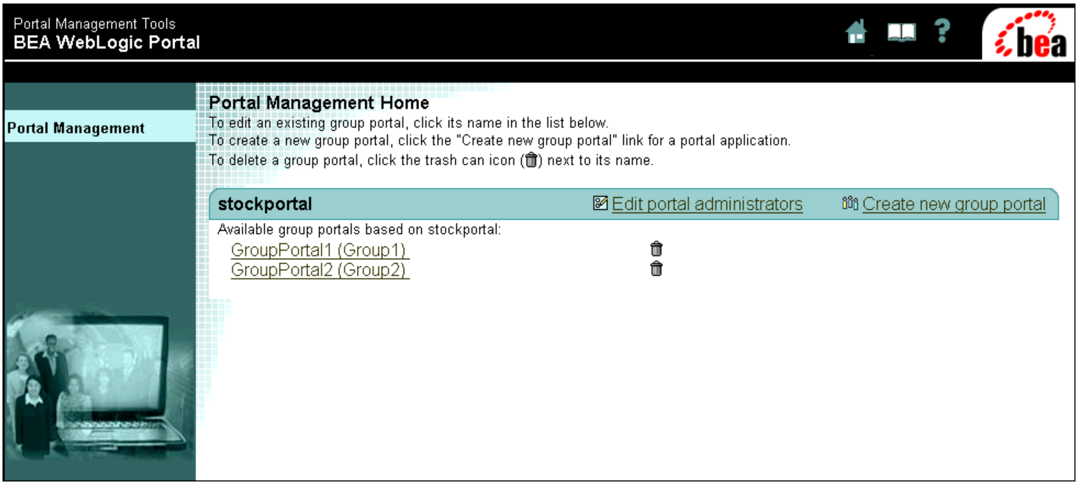
Creating Portal Administrators

Portal Administrators (PAs) have authority to manage aspects of their associated portal. A PA is created by adding a member of the AdminEligible group to the Portal Administrator group and specifying their privileges. To add a user to the Portal Administrator group, an SA must first add them to the AdminEligible group. See “Creating AdminEligible Members” on page 4-10 for the procedure.

SAs and PAs can create a new portal administrator from an existing AdminEligible user. A PA may have authority in multiple portals. Use the following procedure to create a new portal administrator:

1. Click the Portal Management icon in the home page. The Portal Management home page appears, listing all group portals within the portal application (Figure 4-7).

Figure 4-7 Portal Management Home page



Portal Management Home

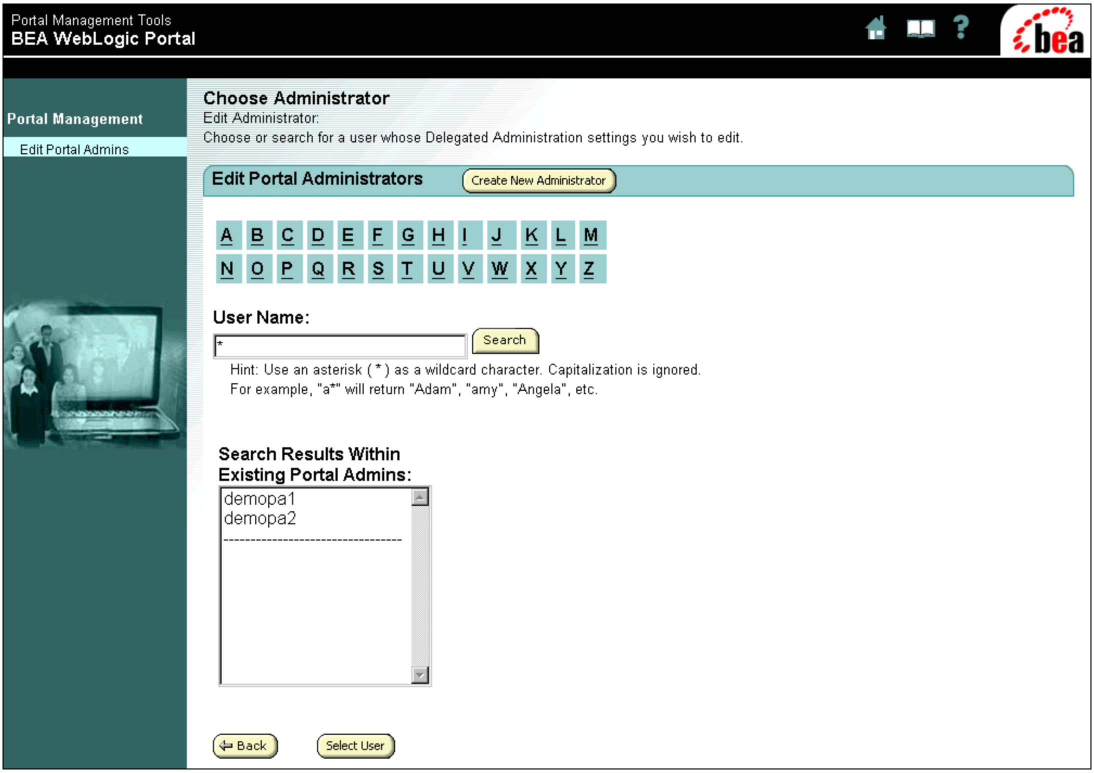
To edit an existing group portal, click its name in the list below.
To create a new group portal, click the "Create new group portal" link for a portal application.
To delete a group portal, click the trash can icon (🗑) next to its name.

Available group portals based on stockportal:

GroupPortal1 (Group1) 🗑
GroupPortal2 (Group2) 🗑

2. Click the Edit Portal Administrators link. The Edit Portal Administrators page appears (Figure 4-8).

Figure 4-8 Edit Portal Administrators page



Portal Management Tools
BEA WebLogic Portal

Portal Management
Edit Portal Admins

Choose Administrator
Edit Administrator:
Choose or search for a user whose Delegated Administration settings you wish to edit.

Edit Portal Administrators

User Name: *

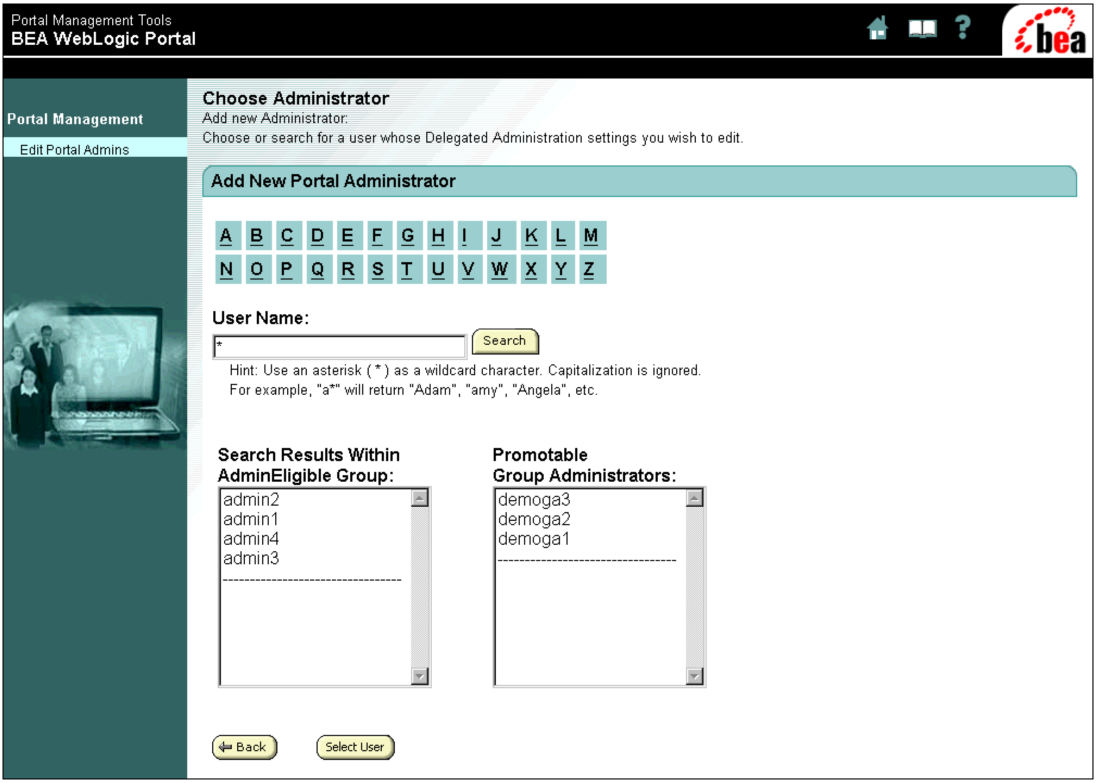
Hint: Use an asterisk (*) as a wildcard character. Capitalization is ignored.
For example, "a*" will return "Adam", "amy", "Angela", etc.

Search Results Within Existing Portal Admins:

demopa1
demopa2

3.In the Edit Portal Administrator page, click Create New Administrator. The Add New Portal Administrator page appears (Figure 4-9).

Figure 4-9 Add New Portal Administrator page



Portal Management Tools
BEA WebLogic Portal

Choose Administrator
Add new Administrator:
Choose or search for a user whose Delegated Administration settings you wish to edit.

Add New Portal Administrator

User Name: *

Hint: Use an asterisk (*) as a wildcard character. Capitalization is ignored.
For example, "a*" will return "Adam", "amy", "Angela", etc.

Search Results Within AdminEligible Group:
admin2
admin1
admin4
admin3

Promotable Group Administrators:
demoga3
demoga2
demoga1
----

4. To select a user to add to the Portal Administrator group, you can do any of the following:

   a. Search for the user you want to add as a PA. You can search for and add any AdminEligible user or existing GA.

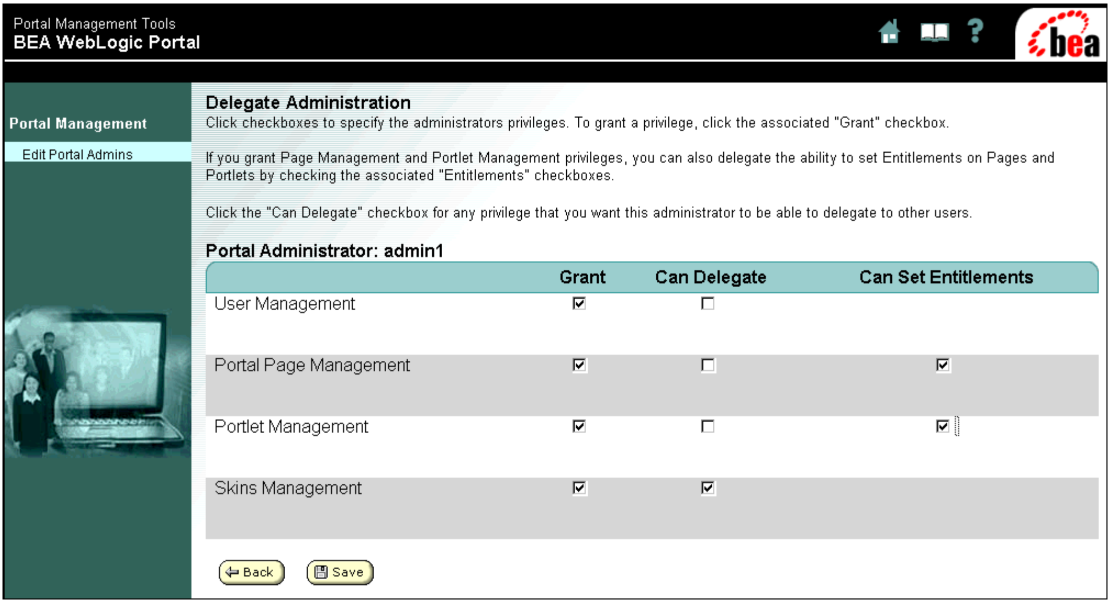
   b. Select a user from the AdminEligible list: Any user in the AdminEligible group can be added to the Portal Administrator group.

   c. Select a User from the Promotable Group Administrator list.

5. Once you have selected the user, click Select User.

6. The Delegate Administration page appears (Figure 4-10).

Figure 4-10 Delegate Administration page



Delegate Administration

Click checkboxes to specify the administrators privileges. To grant a privilege, click the associated "Grant" checkbox.

If you grant Page Management and Portlet Management privileges, you can also delegate the ability to set Entitlements on Pages and Portlets by checking the associated "Entitlements" checkboxes.

Click the "Can Delegate" checkbox for any privilege that you want this administrator to be able to delegate to other users.

Portal Administrator: admin1

	Grant	Can Delegate	Can Set Entitlements
User Management	☑	□	
Portal Page Management	☑	□	☑
Portlet Management	☑	□	☑
Skins Management	☑	☑	

7. Determine what administration permissions the PA will have. Checking each box allows the new PA to perform that function on any group portal within the associated portal Web application.

   a. User Management:

      Grant: The new PA can create, add, remove, delete, and edit the properties of visitors.

      Can delegate: The PA can choose to assign or remove user management authority of other PAs or GAs associated with this portal application.

   b. Portal Page Management:

      Grant: The PA can perform portal page administration. Portal page administration is discussed in detail in “Creating a New Group Portal” on page 4-27.

      Can delegate: The PA can choose to give or remove portal page management authority of other PAs or GAs associated with this portal application.

Can set entitlements: PAs can control visitor portal page capabilities by associating entitlement segments with portal pages. For more information about entitlements, see the “Overview of Portal Administration” chapter of this guide.

c. Portlet Management:

Grant: PA can perform portlet administration. For information about managing portlets, see “Setting Portlet Attributes” on page 4-39.

Can delegate: The PA can choose to give or remove portal management authority of other PAs or GAs associated with this portal application.

Can set entitlements: PA can control visitor group capabilities by associating entitlement segments with portlets. For more information about entitlements, see the “Overview of Portal Administration” chapter of this guide.

d. Skins Management:

Grant: PA can perform skins management. Skins management is discussed in detail in “Selecting Skins” on page 4-31.

Can delegate: The PA can choose to give or remove skins management authority of other PAs or GAs associated with this portal application.

Creating Group Administrators

Group Administrators (GAs) are users in the AdminEligible group who are given limited administrative authority for specific group portals. A GA can have management authority in multiple group portals. This section explains how to give an AdminEligible user authority in an existing group portal. To create a new group portal, see “Creating a New Group Portal” on page 4-27.

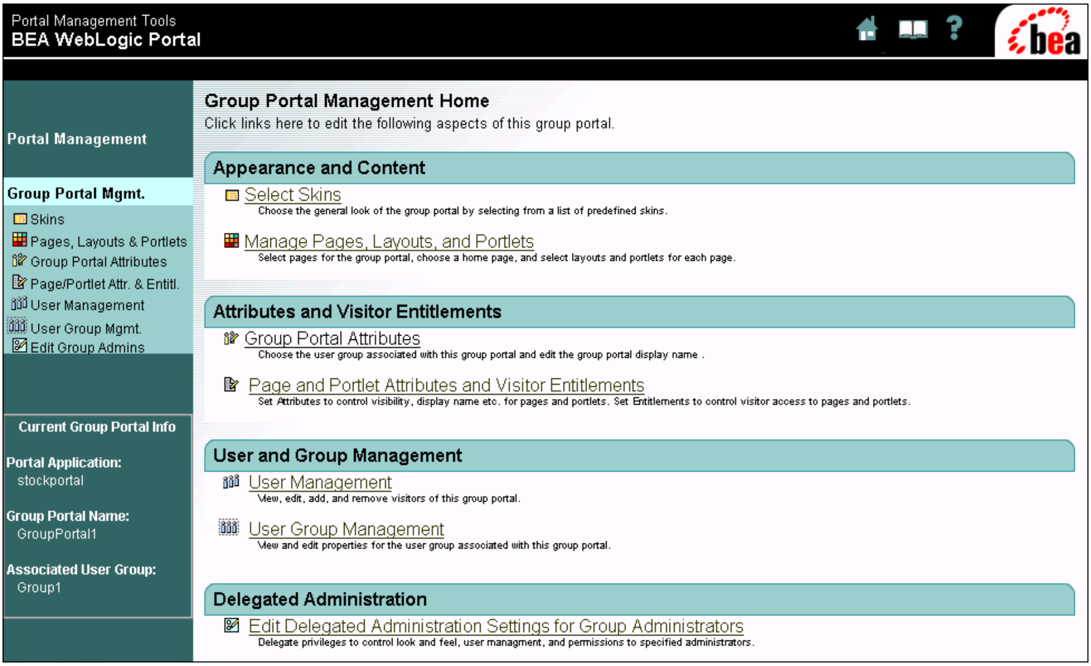
For a user to be a GA, they must first be placed in the AdminEligible group by an SA. See “Creating AdminEligible Members” on page 4-10.

Use the following procedure to create GAs:

1. Click the Portal Management icon in the Portal Administrator Home page. The Portal Management Home page appears.

2. Click the group portal for which you wish to add a GA. The Group Portal Management Home page appears (Figure 4-11).

Figure 4-11 Group Portal Management Home page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.

- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Group Portal Management Home

Click links here to edit the following aspects of this group portal.

Appearance and Content

Select Skins
Choose the general look of the group portal by selecting from a list of predefined skins.

Manage Pages, Layouts, and Portlets
Select pages for the group portal, choose a home page, and select layouts and portlets for each page.

Attributes and Visitor Entitlements

Group Portal Attributes
Choose the user group associated with this group portal and edit the group portal display name .

Page and Portlet Attributes and Visitor Entitlements
Set Attributes to control visibility, display name etc. for pages and portlets. Set Entitlements to control visitor access to pages and portlets.

User and Group Management

User Management
View, edit, add, and remove visitors of this group portal.

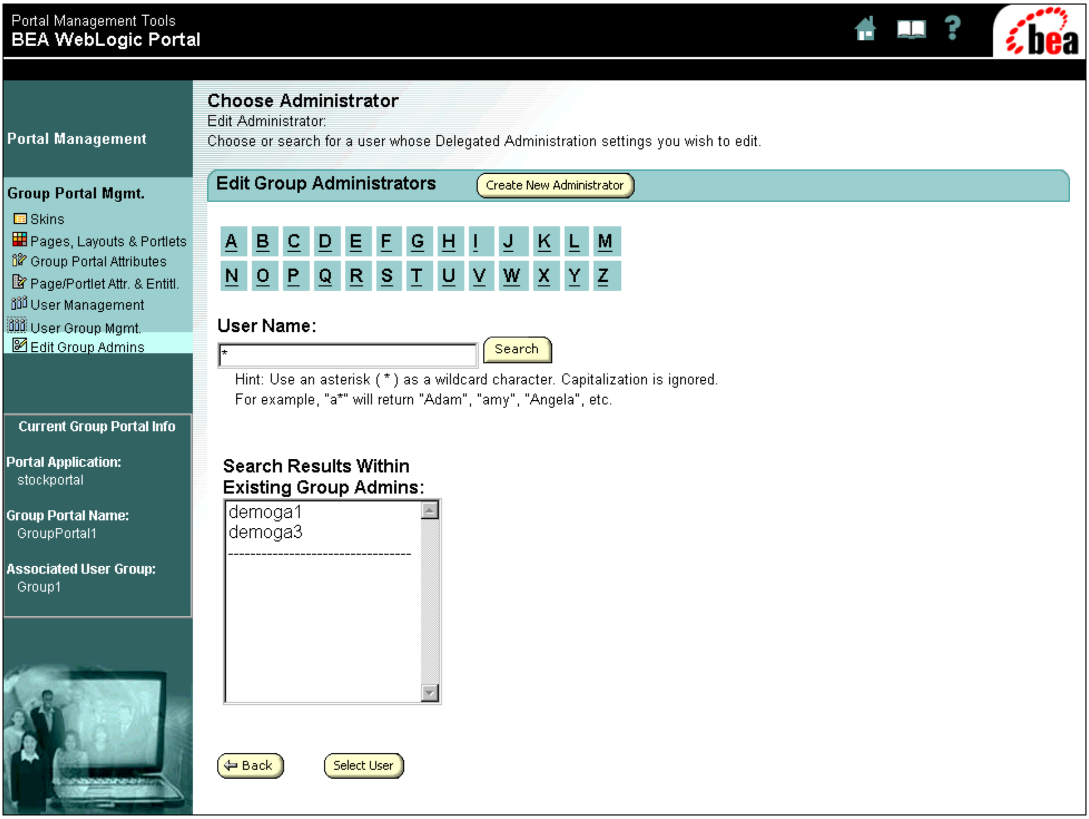
User Group Management
View and edit properties for the user group associated with this group portal.

Delegated Administration

Edit Delegated Administration Settings for Group Administrators
Delegate privileges to control look and feel, user managment, and permissions to specified administrators.

3. Click “Edit Delegated Administration Settings for Group Administrators”. The Edit Group Administrators page appears (Figure 4-12).

Figure 4-12 Edit Group Administrators page



Choose Administrator

Edit Administrator:

Choose or search for a user whose Delegated Administration settings you wish to edit.

Edit Group Administrators

User Name: *

Hint: Use an asterisk (*) as a wildcard character. Capitalization is ignored.
For example, "a*" will return "Adam", "amy", "Angela", etc.

Search Results Within
Existing Group Admins:

demoga1
demoga3

Current Group Portal Info

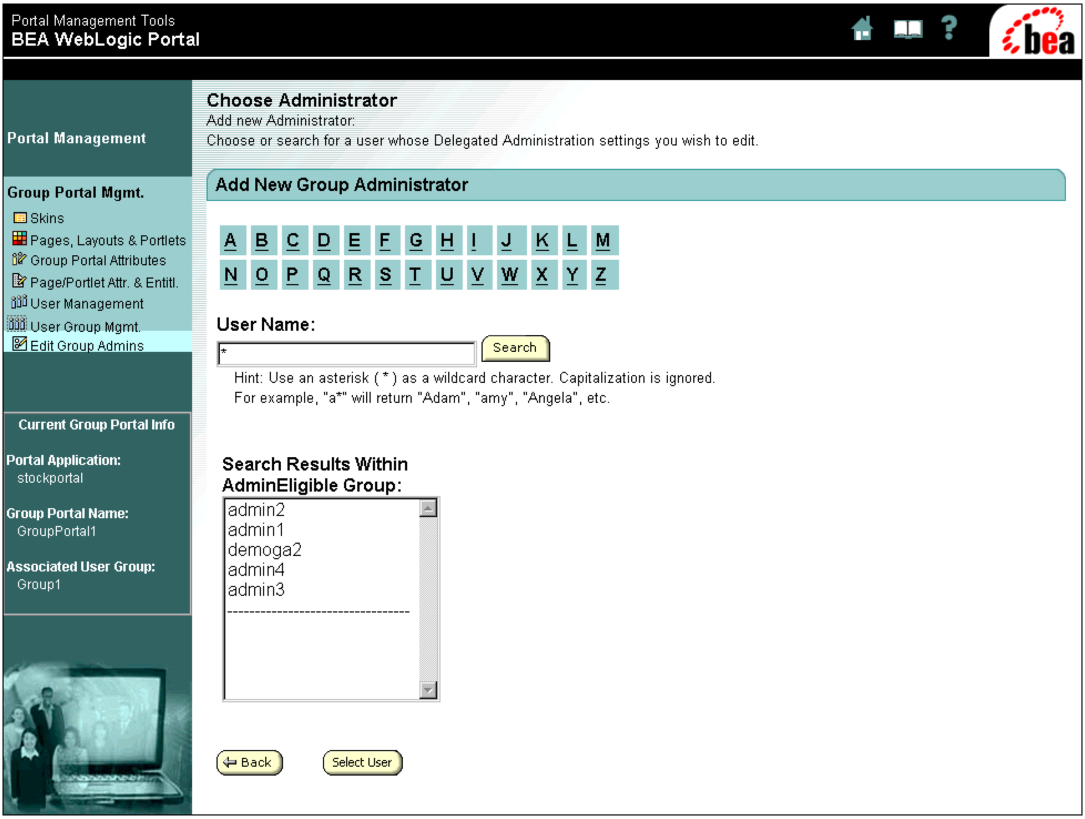
Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

4. Click Create New Administrator. The Choose Administrator page appears (Figure 4-13).

Figure 4-13 Choose Administrator page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Choose Administrator
Add new Administrator:
Choose or search for a user whose Delegated Administration settings you wish to edit.

Add New Group Administrator

A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z

User Name: * Search

Hint: Use an asterisk (*) as a wildcard character. Capitalization is ignored.
For example, "a*" will return "Adam", "amy", "Angela", etc.

Search Results Within
AdminEligible Group:

admin2
admin1
demoga2
admin4
admin3
----

5. Search for the user you want to add to the group.

6. Highlight the desired user and click Select User.

7. The Delegate Administration page appears (Figure 4-14).

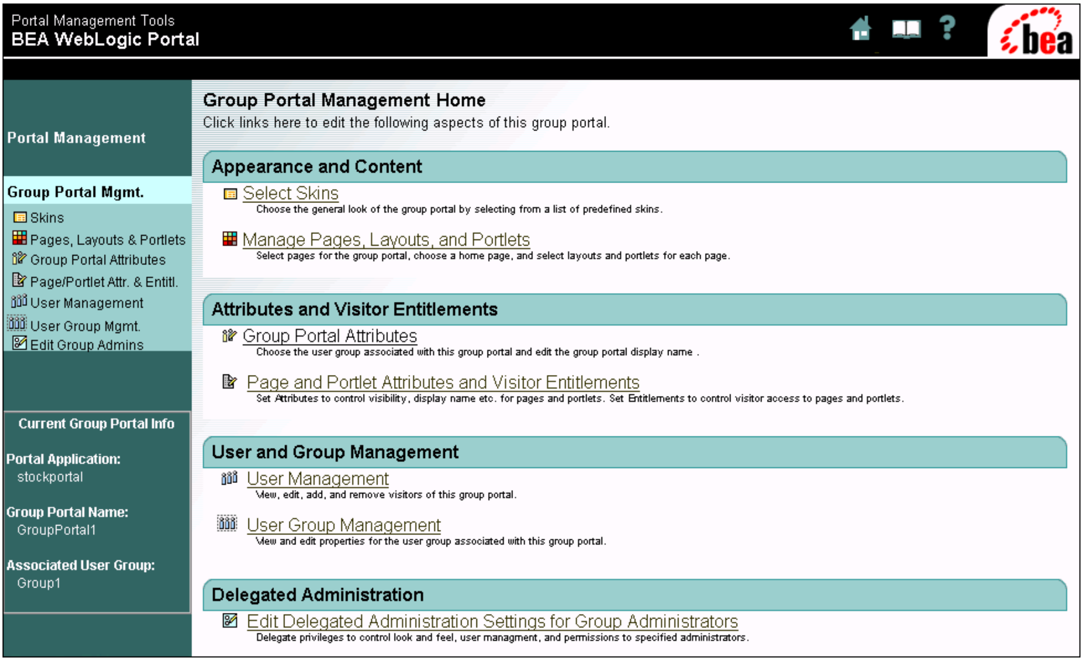
8. In the Delegate Administration page, determine what authority the GA will have. Checking each button allows the new GA to perform that function for this group portal only. See step 7. in “Editing Delegated Administration Properties for Group Administrators” for more information.

Editing Delegated Administration Properties for Group Administrators

Once a Group Administrator exists, you can modify their Administration authority via the WebLogic Portal Administration Tools.

1. Click the Portal Management icon in the Portal Administrator Home page. The Portal Management Home page appears.
2. Click the group portal for which you wish to add a GA. The Group Portal Home page appears (Figure 4-14).

Figure 4-14 Group Portal Management Home page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Group Portal Management Home

Click links here to edit the following aspects of this group portal.

Appearance and Content

Select Skins
Choose the general look of the group portal by selecting from a list of predefined skins.

Manage Pages, Layouts, and Portlets
Select pages for the group portal, choose a home page, and select layouts and portlets for each page.

Attributes and Visitor Entitlements

Group Portal Attributes
Choose the user group associated with this group portal and edit the group portal display name .

Page and Portlet Attributes and Visitor Entitlements
Set Attributes to control visibility, display name etc. for pages and portlets. Set Entitlements to control visitor access to pages and portlets.

User and Group Management

User Management
View, edit, add, and remove visitors of this group portal.

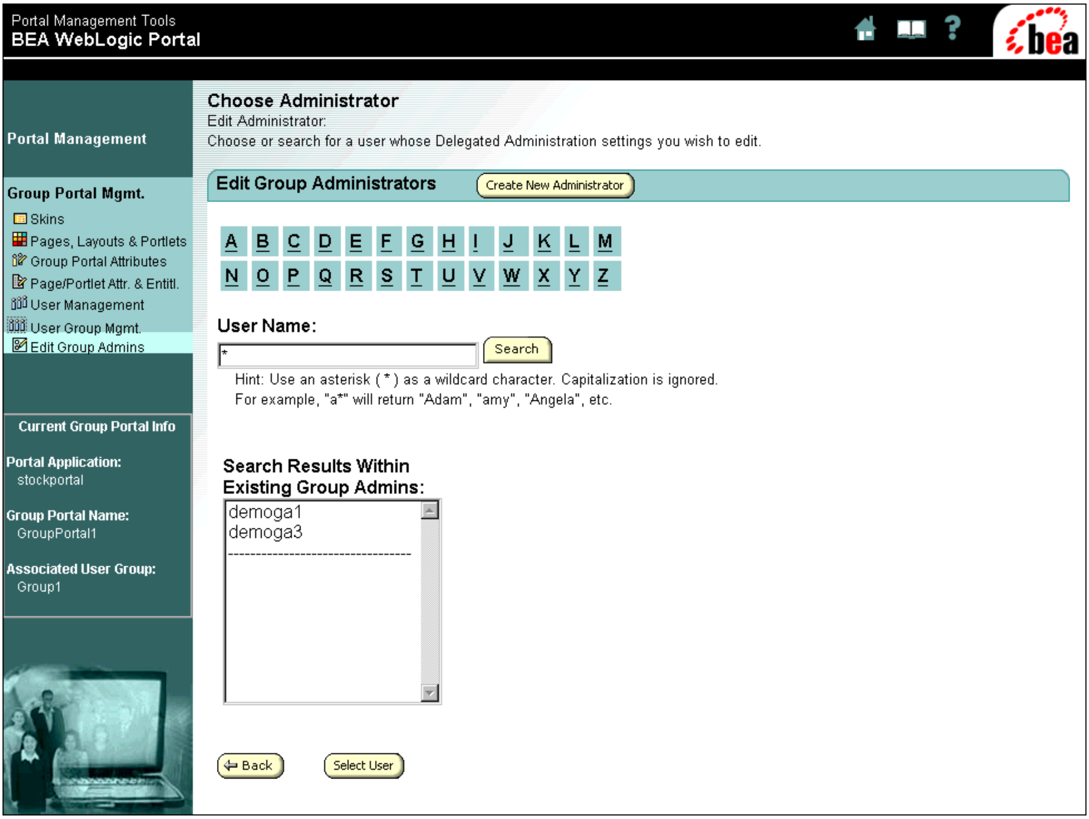
User Group Management
View and edit properties for the user group associated with this group portal.

Delegated Administration

Edit Delegated Administration Settings for Group Administrators
Delegate privileges to control look and feel, user managment, and permissions to specified administrators.

3. Click “Edit Delegated Administration Settings for Group Administrators”. The Choose Administrators page appears (Figure 4-15).

Figure 4-15 Choose Administrators page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

Choose Administrator

Edit Administrator:

Choose or search for a user whose Delegated Administration settings you wish to edit.

Edit Group Administrators

User Name: *

Hint: Use an asterisk (*) as a wildcard character. Capitalization is ignored.
For example, "a*" will return "Adam", "amy", "Angela", etc.

Search Results Within Existing Group Admins:

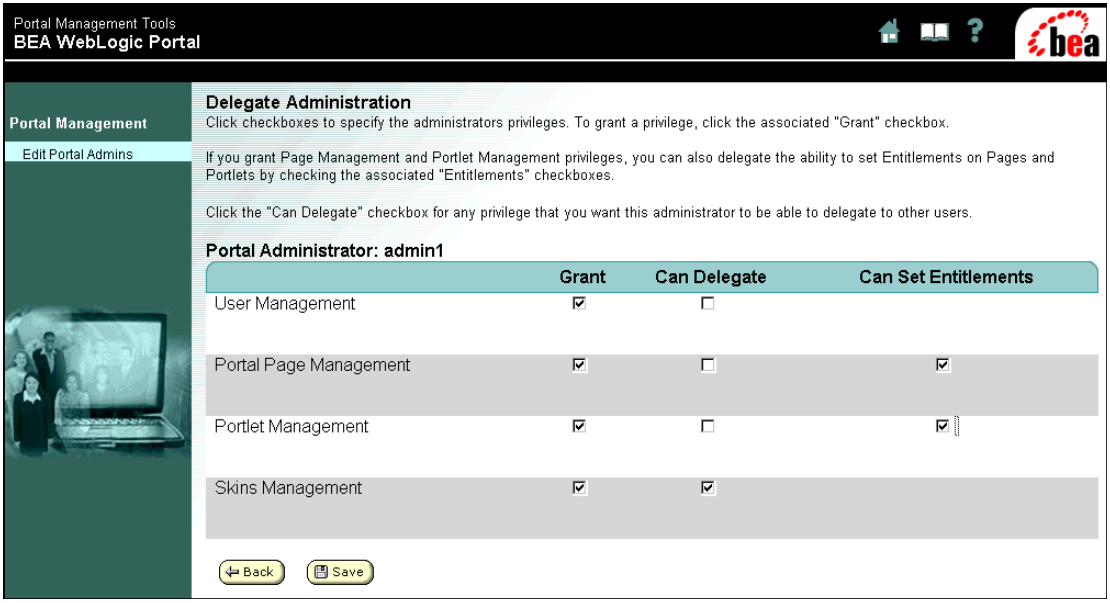
demoga1
demoga3

4. You can search for the Group Administrator for whom you want to modify administrative privileges and select from the Existing Group Administrators list.

5. Select the desired user and click the Select User button.

6. The Delegate Administration page appears (Figure 4-16).

Figure 4-16 Delegate Administration page



Delegate Administration

Click checkboxes to specify the administrators privileges. To grant a privilege, click the associated "Grant" checkbox.

If you grant Page Management and Portlet Management privileges, you can also delegate the ability to set Entitlements on Pages and Portlets by checking the associated "Entitlements" checkboxes.

Click the "Can Delegate" checkbox for any privilege that you want this administrator to be able to delegate to other users.

Portal Administrator: admin1

	Grant	Can Delegate	Can Set Entitlements
User Management	☑	□	
Portal Page Management	☑	□	☑
Portlet Management	☑	□	☑
Skins Management	☑	☑	

7. Determine what administration permissions the GA will have. Checking each box allows the new GA to perform that function in the within the associated group portal. See “Creating Group Administrators” on page 4-15 for information about creating a GA associated with a specific group portal.

   a. User Management:

      Grant: The new GA can create, add, remove, delete, and edit the properties of visitors.

      Can delegate: The GA can choose to assign or remove user management authority of other GAs associated with this portal application.

   b. Portal Page Management:

      Grant: The GA can perform portal page administration. Portal page administration is discussed in detail in “Creating a New Group Portal” on page 4-27.

Can delegate: The GA can choose to give or remove portal page management authority of other GAs associated with the associated group portal.

Can set entitlements: GA can control visitor portal page capabilities by associating entitlement segments with portal pages. For more information about entitlements, see the “Overview of Portal Administration” chapter of this guide.

c. Portlet Management:

Grant: GA can perform portlet administration. For information about managing portlets, see “Setting Portlet Attributes” on page 4-39.

Can delegate: GA can delegate portlet management to other GAs associated with this portal application.

Can set entitlements: GA can control writer visitor portlet capabilities by associating entitlement segments with portlets. For more information about entitlements, see the “Overview of Portal Administration” chapter of this guide.

d. Skins Management:

Grant: GA can perform skins management. Skins management is discussed in detail in “Selecting Skins” on page 4-31.

Can delegate: GA can delegate skins management authority to other GAs associated with this portal application.

8. Click Save when you are finished.

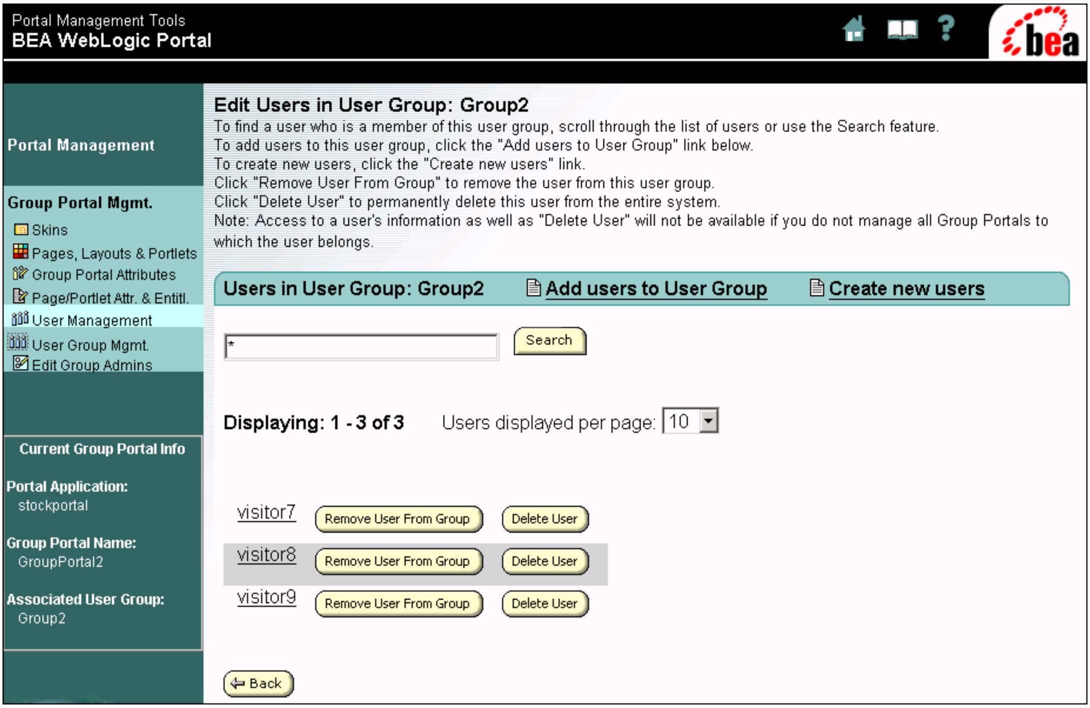
Managing Existing Users

You can use the WebLogic Portal Administration Tools to manage existing users and groups associated with a portal.

1. Click The Portal Management icon in the home page. The Group Portal page appears.
2. Click on the Group Portal you wish to edit in the Portal Management Home page, the Portal Management Home page appears.

3. In the User and Group Management section, click User Management. The Edit Users in Group page displays a list of users for the selected group (Figure 4-17).

Figure 4-17 Edit Users in Group page



Portal Application: stockportal

Group Portal Name: GroupPortal2

Associated User Group: Group2

Edit Users in User Group: Group2

To find a user who is a member of this user group, scroll through the list of users or use the Search feature.
To add users to this user group, click the "Add users to User Group" link below.
To create new users, click the "Create new users" link.
Click "Remove User From Group" to remove the user from this user group.
Click "Delete User" to permanently delete this user from the entire system.
Note: Access to a user's information as well as "Delete User" will not be available if you do not manage all Group Portals to which the user belongs.

Users in User Group: Group2 | Add users to User Group | Create new users

*

Displaying: 1 - 3 of 3   Users displayed per page: 10

visitor7
visitor8
visitor9

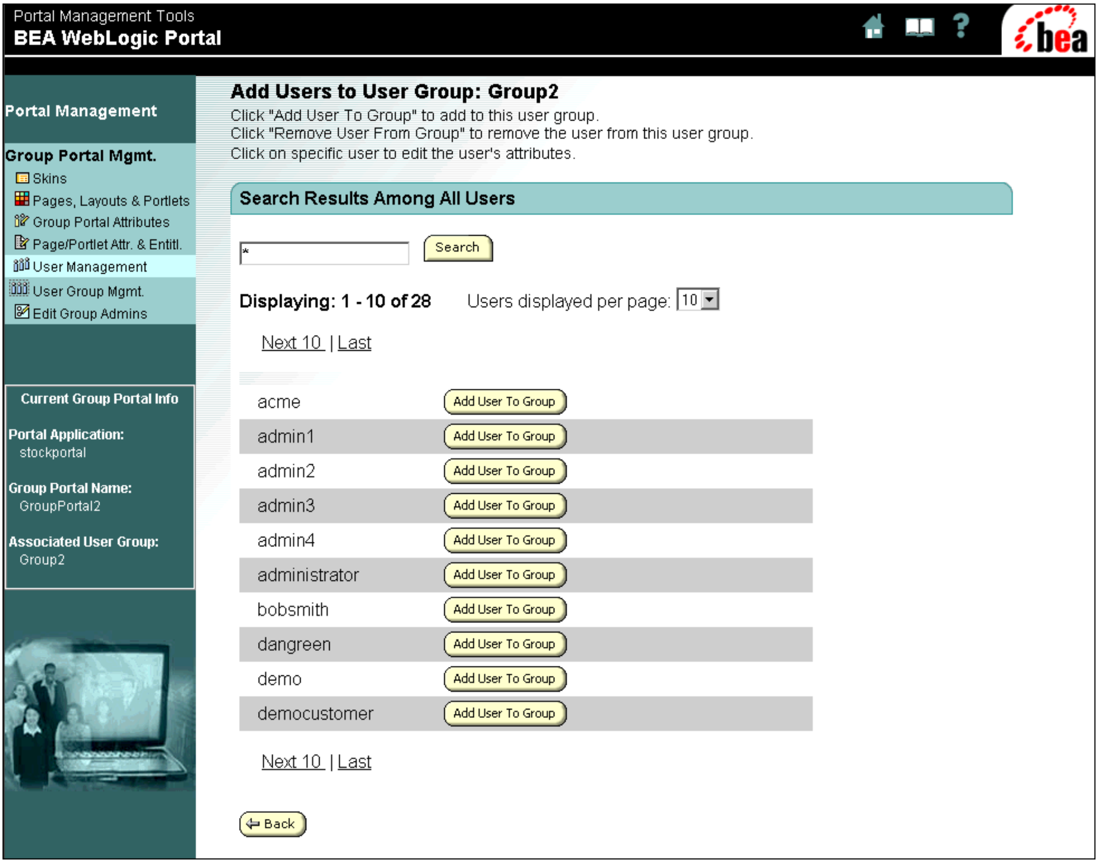
a. To remove a user from the group, select the desired user and click the Remove User From Group button. The user will be removed from the group and will no longer be displayed in the list.

b. To delete the user from the system, select the desired user and click the Delete button. The user is deleted from the system and is removed from the display list.

c. To create new users, click Create New Users and see “Creating Users” on page 4-6 for more information.

d. To add users to the group, click Add Users to the Group; The Add users to Group page displays a list of available users (Figure 4-18).

Figure 4-18 Add Users to Group page



Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal2

Associated User Group: Group2

Add Users to User Group: Group2

Click "Add User To Group" to add to this user group.
Click "Remove User From Group" to remove the user from this user group.
Click on specific user to edit the user's attributes.

Search Results Among All Users

*

Displaying: 1 - 10 of 28   Users displayed per page: 10

acme	Add User To Group
admin1	Add User To Group
admin2	Add User To Group
admin3	Add User To Group
admin4	Add User To Group
administrator	Add User To Group
bobsmith	Add User To Group
dangreen	Add User To Group
demo	Add User To Group
democustomer	Add User To Group

e. Select the user you want to add to the group and click the Add User to Group button.

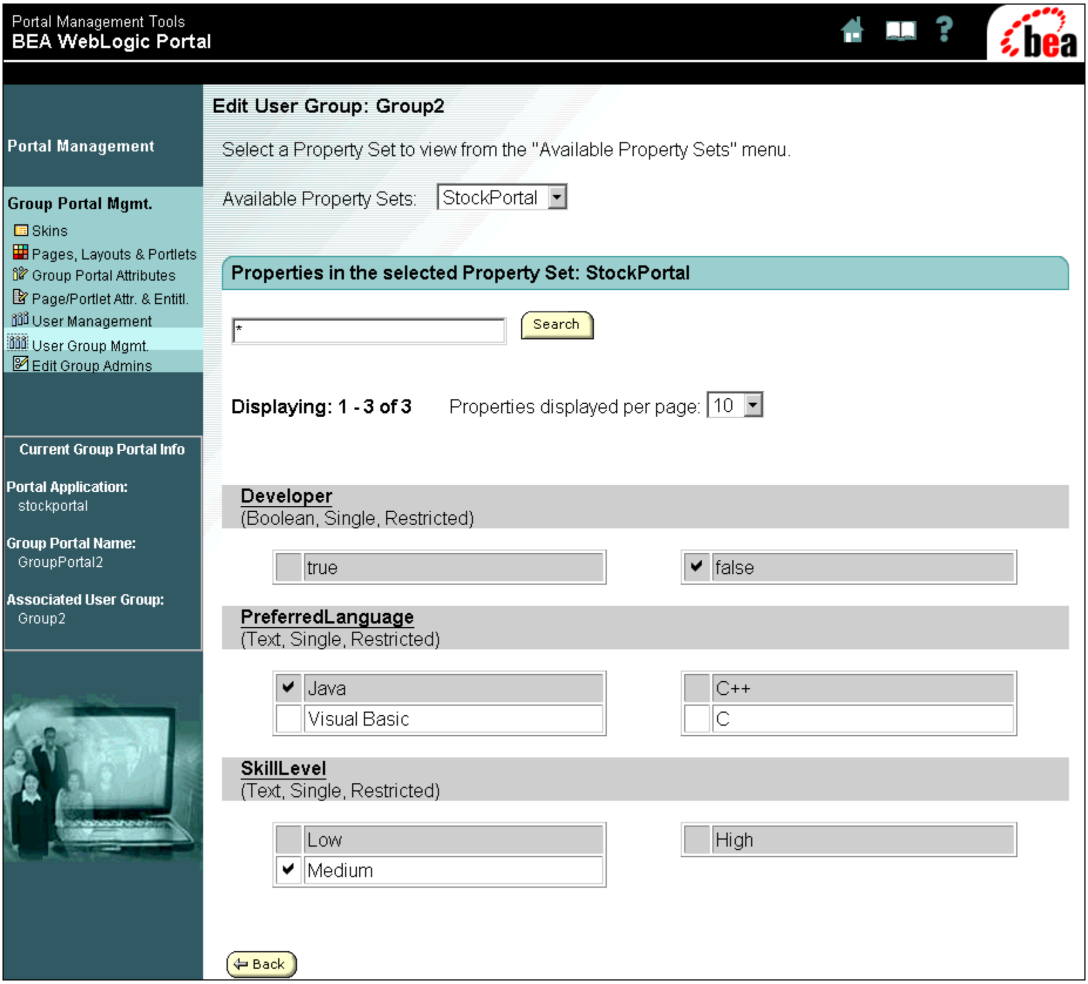
4. Click Back to return to the Edit Users in Group page.

5. Click Back in the Edit Users in Group page when you are finished.

Managing Existing User Groups

1. Click The Portal Management icon in the home page. The Group Portal page appears.
2. Click on the Group Portal you wish to edit, the Portal Management Home page appears.
3. In the User and Group Management section, click User Group Management. The Edit User Group page displays a list of attributes for the selected user group (Figure 4-19).

Figure 4-19 Edit User Group page



Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal2

Associated User Group: Group2

Edit User Group: Group2

Select a Property Set to view from the "Available Property Sets" menu.

Available Property Sets: StockPortal

Properties in the selected Property Set: StockPortal

*

Displaying: 1 - 3 of 3 Properties displayed per page: 10

Developer
(Boolean, Single, Restricted)

☐ true ☑ false

PreferredLanguage
(Text, Single, Restricted)

☑ Java ☐ C++
☐ Visual Basic ☐ C

SkillLevel
(Text, Single, Restricted)

☐ Low ☐ High
☑ Medium

4. Select the desired property set for the selected group portal, then edit any of the following information:

   Developer: Check true if this property set requires that members be developers

   Preferred language: Indicate the preferred coding language if the members are developers

   Skill Level: Indicate a skill level if the member is a developer

5. Click Back when you are finished.

Portal Management

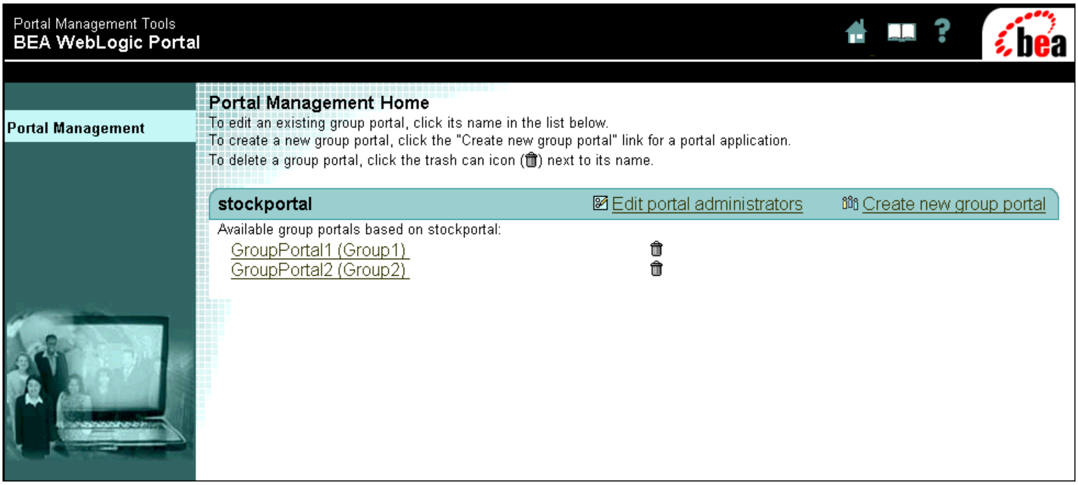
Although portals, pages and portlets are defined in the E-Business Control Center, you can modify certain runtime attributes using the WebLogic Portal Administration Tools.

Creating a New Group Portal

SAs and PAs can use the WebLogic Portal Administration Tools to create a new group portal within a portal application.

1. Click The Portal Management icon in the home page. The Portal Management Home page appears (Figure 4-20).

Figure 4-20 Portal Management Home page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Portal Management Home

To edit an existing group portal, click its name in the list below.
To create a new group portal, click the "Create new group portal" link for a portal application.
To delete a group portal, click the trash can icon (🗑) next to its name.

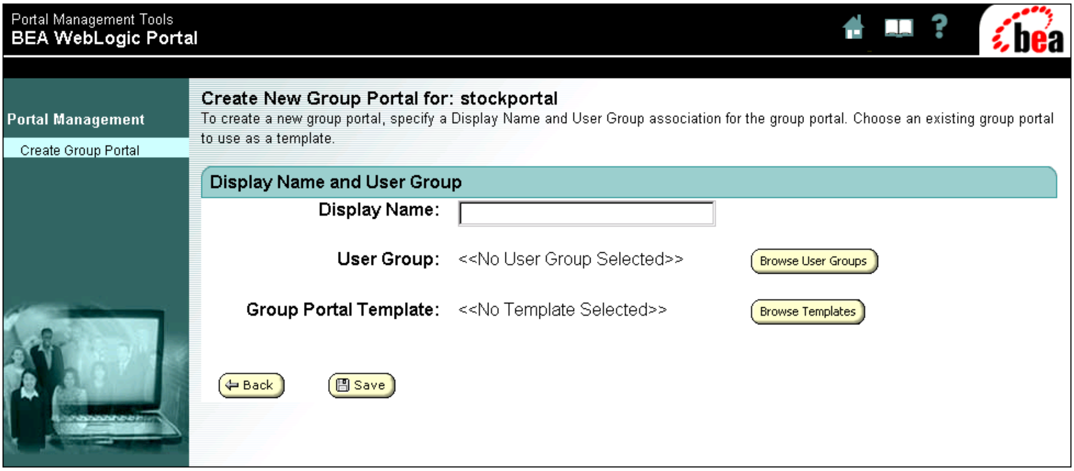
stockportal — Edit portal administrators — Create new group portal

Available group portals based on stockportal:

GroupPortal1 (Group1) 🗑
GroupPortal2 (Group2) 🗑

2. Click “Create a New Group Portal” in the Portal Management Home page. The New Group Portal page appears (Figure 4-21).

Figure 4-21 New Group Portal page



Portal Management Tools
BEA WebLogic Portal

Create New Group Portal for: stockportal

To create a new group portal, specify a Display Name and User Group association for the group portal. Choose an existing group portal to use as a template.

Display Name and User Group

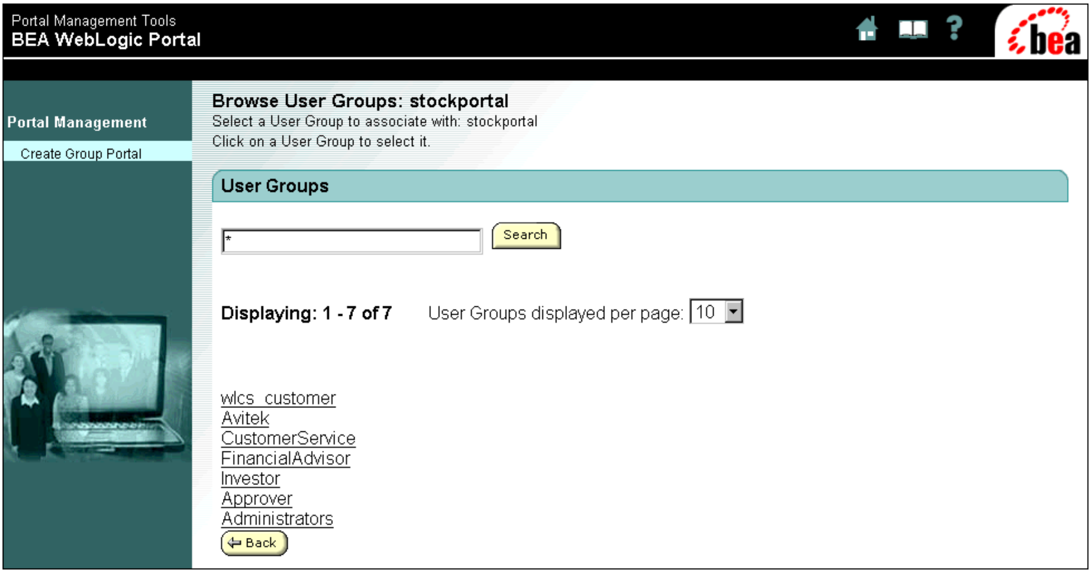
Display Name:

User Group: <<No User Group Selected>>

Group Portal Template: <<No Template Selected>>

3. Enter a Display name for the group portal.
4. Select a user group to associate with this group portal.
    a. Click Browse Groups; The Browse User Groups page appears (Figure 4-22).

Figure 4-22 Browse User Groups page



Portal Management Tools
BEA WebLogic Portal

Portal Management
Create Group Portal

Browse User Groups: stockportal
Select a User Group to associate with: stockportal
Click on a User Group to select it.

User Groups

*

Search

Displaying: 1 - 7 of 7   User Groups displayed per page: 10

wlcs_customer
Avitek
CustomerService
FinancialAdvisor
Investor
Approver
Administrators

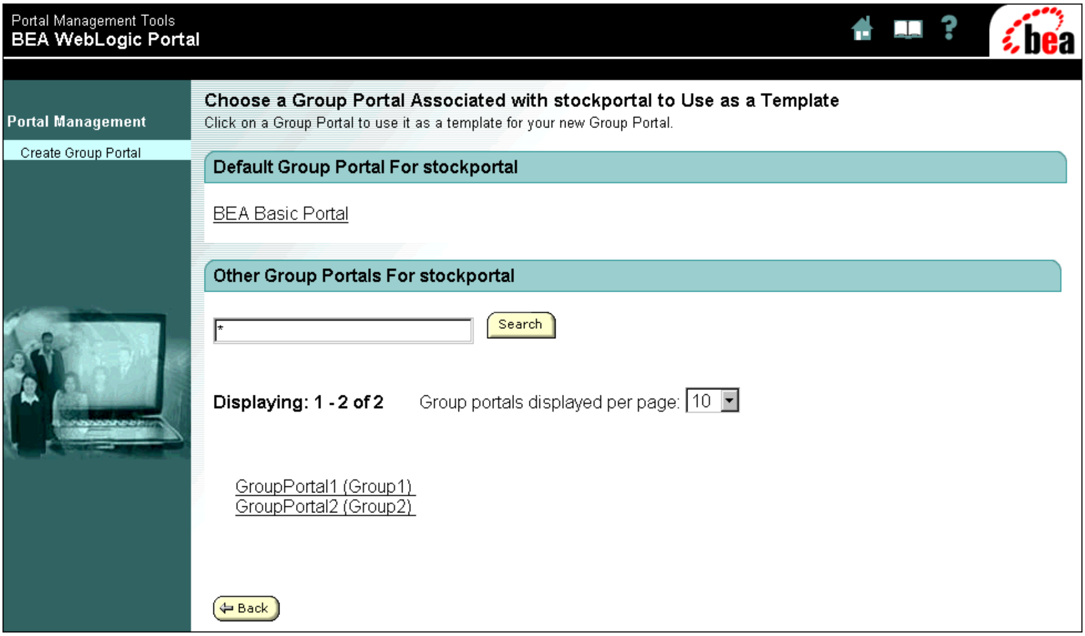
Back

b. Click on a user group from those available for the portal. The Create New Group Portal page displays the chosen User group.

5. Select a template for the new group portal.

a. Click Browse Templates; A list of existing group portals is listed in the Choose Group Portal page (Figure 4-23).

Figure 4-23 Choose Group Portal page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Create Group Portal

Choose a Group Portal Associated with stockportal to Use as a Template

Click on a Group Portal to use it as a template for your new Group Portal.

Default Group Portal For stockportal

BEA Basic Portal

Other Group Portals For stockportal

*

Displaying: 1 - 2 of 2 Group portals displayed per page: 10

GroupPortal1 (Group1)
GroupPortal2 (Group2)

b. Click on the group portal you would like to use as a template. The Create New Group Portal page displays the selected template.

6. In the Create New Group Portal page, determine the following information:

a. Copy Entitlements: You can copy existing entitlements from the template group portal and keep them as is and/or edit them later. You can also choose not to copy the existing entitlements and create new entitlements. The procedure for creating new entitlements is described in greater detail later in this document: “Setting Page Entitlements” on page 4-37 and “Setting Portlet Attributes” on page 4-39.

b. Copy Group Administrators: When you copy GAs, the same GAs will have the same authority in this group portal as those in the template group portal you have selected. If you choose not to copy the GAs you can assign your own. If you choose to copy the existing GAs you can add to or remove them later. To add GAs, see “Creating Group Administrators” on page 4-15.

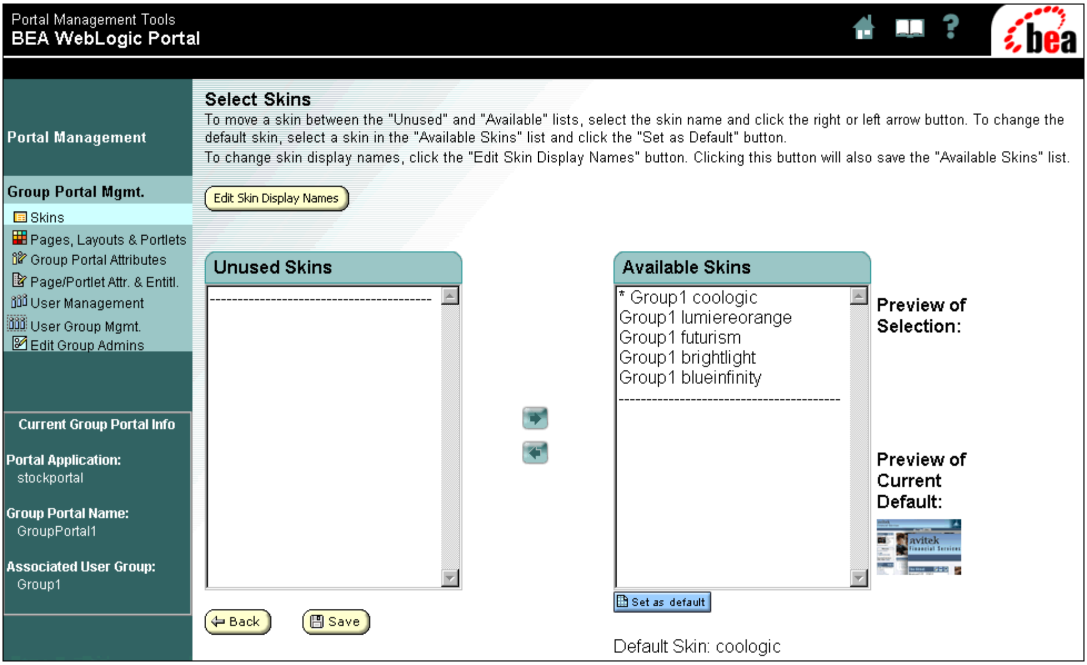
Modifying the Appearance and Content of Existing Portals

Portals are defined using the E-Business Control Center. You can use the WebLogic Portal Administration Tools to change the content and appearance of group portals, portal pages, and portlets. For more information about defining portals and portlets, see the “Using the E-Business Control Center” chapter in this document.

Selecting Skins

1. Click The Portal Management icon in the home page. The Group Portal page appears.
2. Click on the Group Portal you wish to edit in the Portal Management Home page, the Portal Management Home page appears.
3. Under Appearance and Content section, click Select Skins. The Select Skins page displays a list of unused and available skins (Figure 4-24). The default skin is indicated by an asterisk (*).

Figure 4-24 Select Skins page



Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

Select Skins

To move a skin between the "Unused" and "Available" lists, select the skin name and click the right or left arrow button. To change the default skin, select a skin in the "Available Skins" list and click the "Set as Default" button.

To change skin display names, click the "Edit Skin Display Names" button. Clicking this button will also save the "Available Skins" list.

* Group1 coologic
Group1 lumiereorange
Group1 futurism
Group1 brightlight
Group1 blueinfinity

Default Skin: coologic

4. To view a thumbnail of a skin, highlight the desired skin. A thumbnail of that skin will appear under the Preview Skin heading.
5. To set a new default, highlight the desired skin and click Set as Default; The new default skin is marked with an asterisk (*).
6. You can move skins between the Available and Unused lists by selecting the skin and clicking the left and right arrows. Making a skin available means that visitors can select that skin when personalizing their portal.
7. Click Save when you are done.

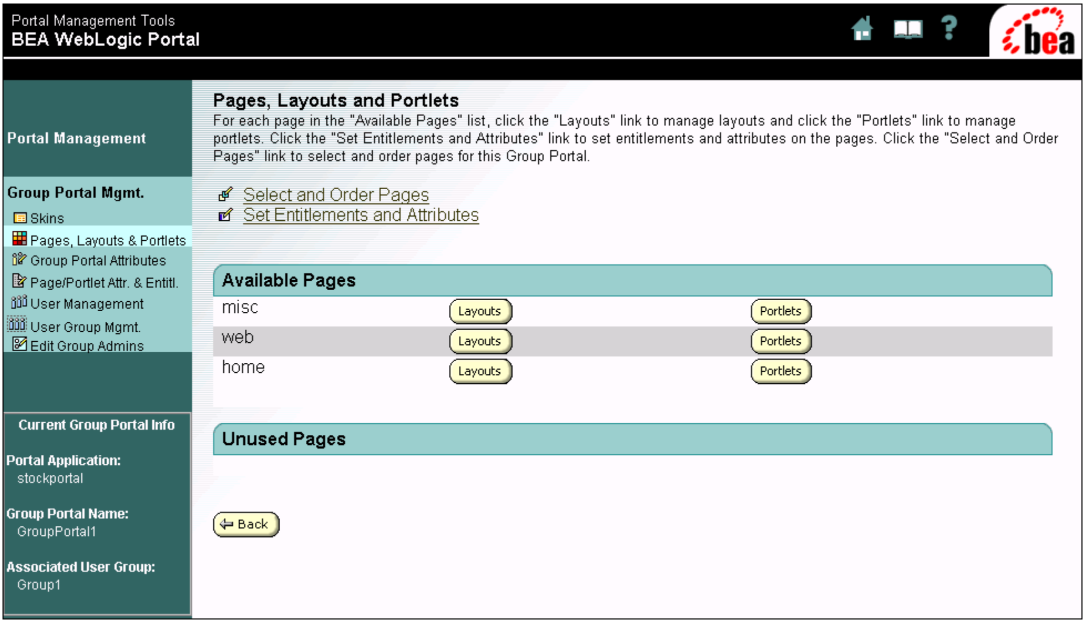
Selecting and Ordering Pages:

1. Click The Portal Management icon in the home page. The Portal Management Home page appears.

2. Click on the Group Portal you wish to edit. The Group Portal Management home page appears.

3. In the Group Portal Management Home page, click Manage Portlets, Pages and Layouts. The Pages, Layouts, and Portlets page appears (Figure 4-25).

Figure 4-25 Pages, Layouts, and Portlets page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Pages, Layouts and Portlets

For each page in the "Available Pages" list, click the "Layouts" link to manage layouts and click the "Portlets" link to manage portlets. Click the "Set Entitlements and Attributes" link to set entitlements and attributes on the pages. Click the "Select and Order Pages" link to select and order pages for this Group Portal.

Select and Order Pages
Set Entitlements and Attributes

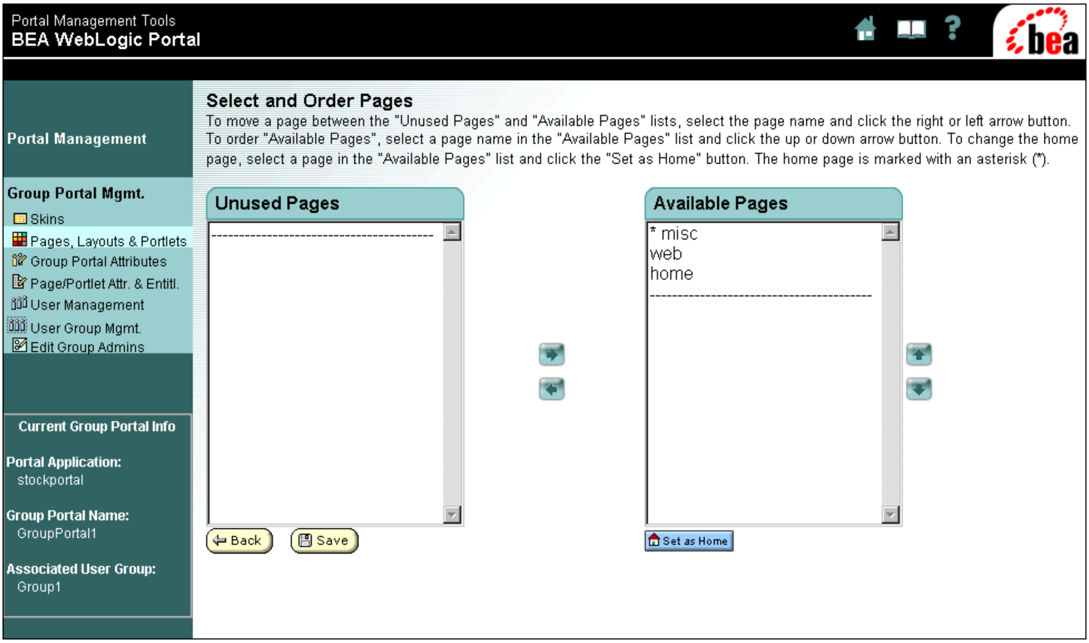
Available Pages

misc	Layouts	Portlets
web	Layouts	Portlets
home	Layouts	Portlets

Unused Pages

a. Click the Select and Order Pages link. The Select and Order Pages page displays a list of available and unused pages (Figure 4-26). The Home (default) page is indicated by an asterisk (*).

Figure 4-26 Select and Order Pages page



Select and Order Pages

To move a page between the "Unused Pages" and "Available Pages" lists, select the page name and click the right or left arrow button. To order "Available Pages", select a page name in the "Available Pages" list and click the up or down arrow button. To change the home page, select a page in the "Available Pages" list and click the "Set as Home" button. The home page is marked with an asterisk (*).

Unused Pages

Available Pages

* misc
web
home

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

b. To reset the Home page, highlight the desired page and click Set as Home; The new default page is indicated by an asterisk (*).

c. You can move pages between the Available Pages and Unused Pages lists by selecting the page name and clicking the Left and Right arrows. Making a page available means that visitors may choose to display the page as a tab when personalizing their portal.

d. You can reorder pages by selecting the desired page in the Available Pages list and using the Up and Down arrows. This determines the order in which the page tabs will be displayed.

e. Click Save when you are finished.

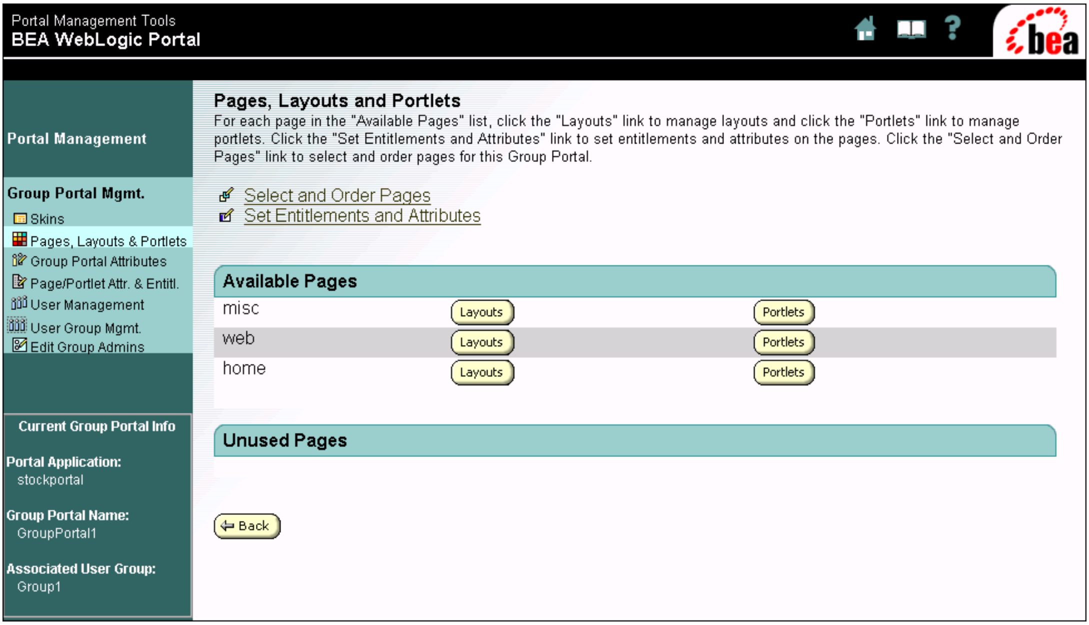
Setting Page Attributes

1. Click the Portal Management icon in the home page. The Portal Management Home page appears.

2. Click on the Group Portal you wish to edit. The Group Portal Management home page appears.

3. In the Group Portal Management Home page, click “Manage Portlets, Pages and Layouts”. The Pages, Layouts, and Portlets page appears (Figure 4-27).

Figure 4-27 Pages, Layouts, and Portlets page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Pages, Layouts and Portlets

For each page in the "Available Pages" list, click the "Layouts" link to manage layouts and click the "Portlets" link to manage portlets. Click the "Set Entitlements and Attributes" link to set entitlements and attributes on the pages. Click the "Select and Order Pages" link to select and order pages for this Group Portal.

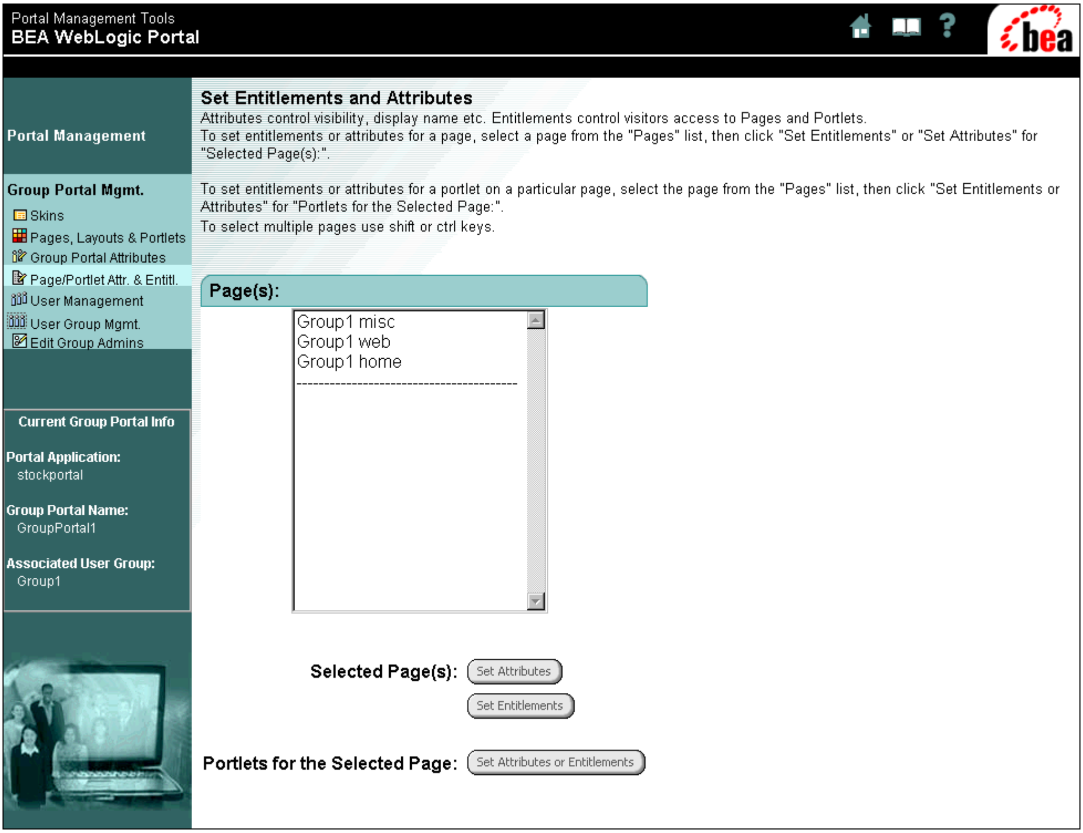
Select and Order Pages
Set Entitlements and Attributes

Available Pages		
misc	Layouts	Portlets
web	Layouts	Portlets
home	Layouts	Portlets

Unused Pages

a. In the Pages, Layouts and Portlets page, click “Set Entitlements and Attributes” link. The Set Entitlements and Attributes page appears (Figure 4-28).

Figure 4-28 Set Entitlements and Attributes page



Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

Set Entitlements and Attributes

Attributes control visibility, display name etc. Entitlements control visitors access to Pages and Portlets.
To set entitlements or attributes for a page, select a page from the "Pages" list, then click "Set Entitlements" or "Set Attributes" for "Selected Page(s):".

To set entitlements or attributes for a portlet on a particular page, select the page from the "Pages" list, then click "Set Entitlements or Attributes" for "Portlets for the Selected Page:".
To select multiple pages use shift or ctrl keys.

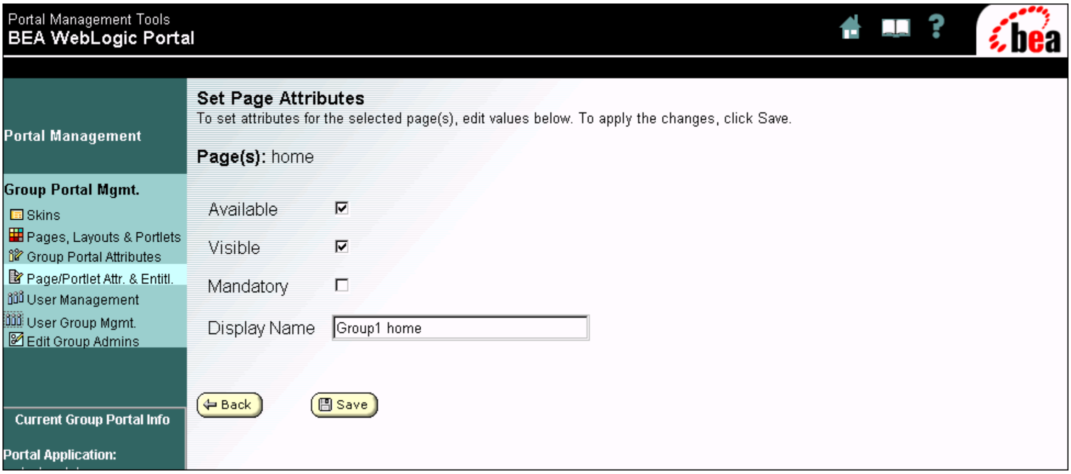
Page(s):

Group1 misc
Group1 web
Group1 home

Selected Page(s): Set Attributes
Set Entitlements
Portlets for the Selected Page: Set Attributes or Entitlements

b. Select the page for which you wish to set the attributes and click Set Attributes. The Set Page Attributes page appears (Figure 4-29).

Figure 4-29 Set Page Attributes page



Set Page Attributes

To set attributes for the selected page(s), edit values below. To apply the changes, click Save.

Page(s): home

Available ☑

Visible ☑

Mandatory □

Display Name Group1 home

c. You can modify the following:

   Available: The page will be available to the visitor.

   Visible: The page should be visible to the visitor default.

   Mandatory: The visitor will always see this page. That is, they cannot remove it from their personalized portal.

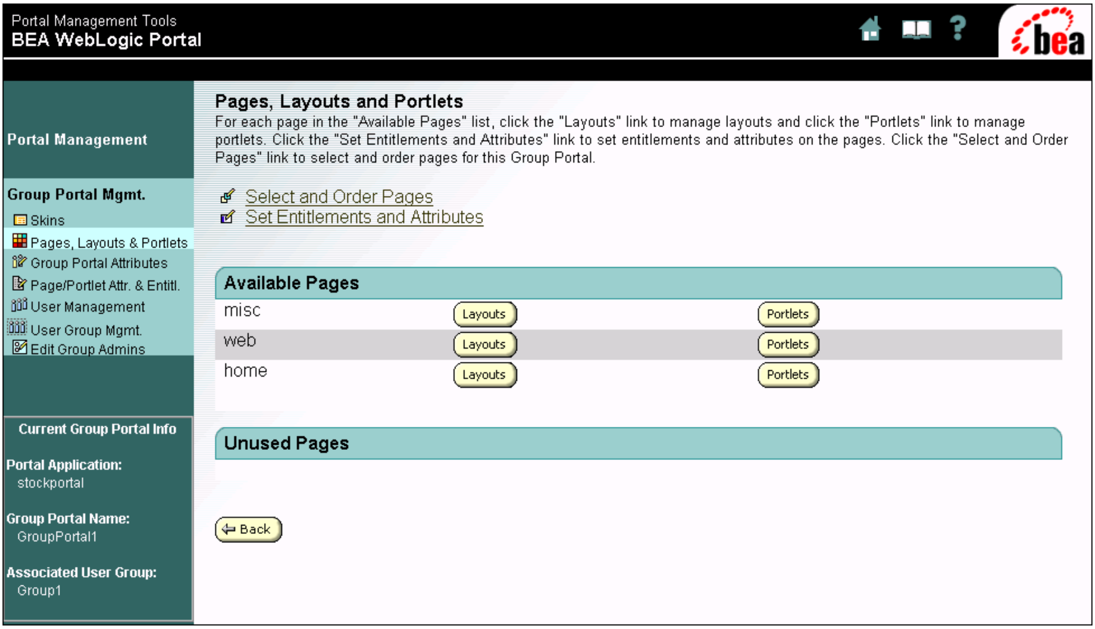
   Display Name: The display name for the page, this is the name that site visitors will see.

d. Click Save when you are finished.

Setting Page Entitlements

1. Click the Portal Management icon in the home page. The Portal Management Home page appears.
2. Click on the Group Portal you wish to edit. The Group Portal Management home page appears.
3. In the Group Portal Management Home page, click “Manage Portlets, Pages and Layouts”. The Pages, Layouts, and Portlets page appears (Figure 4-30).

Figure 4-30 Pages, Layouts, and Portlets page



Pages, Layouts and Portlets

For each page in the "Available Pages" list, click the "Layouts" link to manage layouts and click the "Portlets" link to manage portlets. Click the "Set Entitlements and Attributes" link to set entitlements and attributes on the pages. Click the "Select and Order Pages" link to select and order pages for this Group Portal.

Select and Order Pages
Set Entitlements and Attributes

Available Pages

misc	Layouts	Portlets
web	Layouts	Portlets
home	Layouts	Portlets

Unused Pages

Current Group Portal Info

Portal Application:
stockportal

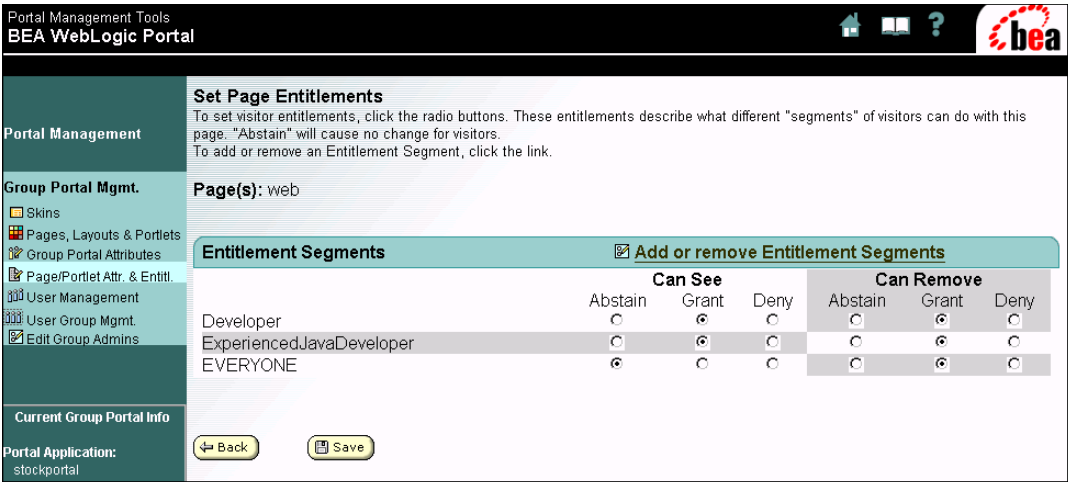
Group Portal Name:
GroupPortal1

Associated User Group:
Group1

4. In the Pages, Layouts and Portlets page, click “Set Entitlements and Attributes” link. The Set Entitlements and Attributes page appears.

5. Select the page for which you wish to set the entitlements and click the Set Entitlements button. The Set Page Entitlements page appears (Figure 4-31).

Figure 4-31 Set Page Entitlements page



Portal Application: stockportal

Set Page Entitlements

To set visitor entitlements, click the radio buttons. These entitlements describe what different "segments" of visitors can do with this page. "Abstain" will cause no change for visitors.
To add or remove an Entitlement Segment, click the link.

Page(s): web

Entitlement Segments Add or remove Entitlement Segments

	Can See			Can Remove		
	Abstain	Grant	Deny	Abstain	Grant	Deny
Developer	○	●	○	○	●	○
ExperiencedJavaDeveloper	○	●	○	○	●	○
EVERYONE	●	○	○	○	●	○

a. Can See: Check the radio buttons to grant or deny availability of this page to the entitlement segment members.

b. Can Remove: Check the radio buttons to grant or deny the entitlement segment members the ability to remove the portal page from their view.

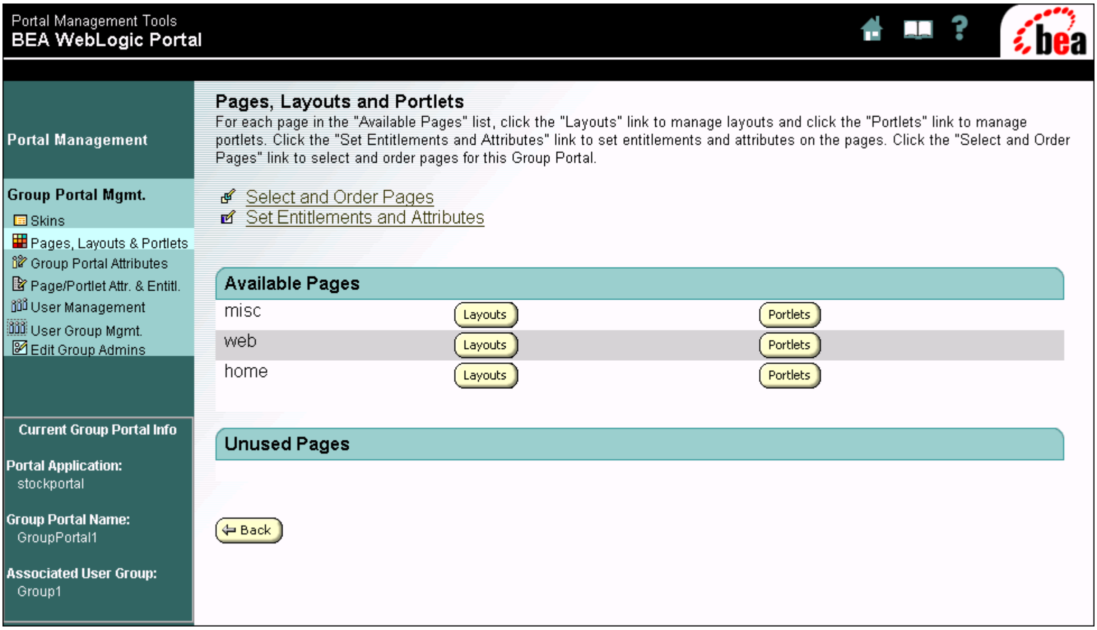
Note: The entitlement segment "Everyone" represents every user in the system.

6. Click Save when you are finished.

Setting Portlet Attributes

1. Click the Portal Management icon in the home page. The Portal Management Home page appears.
2. Click on the Group Portal you wish to edit. The Group Portal Management home page appears.
3. In the Group Portal Management Home page, click Manage Portlets, Pages and Layouts. The Pages, Layouts, and Portlets page appears (Figure 4-32).

Figure 4-32 Pages, Layouts, and Portlets page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.

- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Pages, Layouts and Portlets

For each page in the "Available Pages" list, click the "Layouts" link to manage layouts and click the "Portlets" link to manage portlets. Click the "Set Entitlements and Attributes" link to set entitlements and attributes on the pages. Click the "Select and Order Pages" link to select and order pages for this Group Portal.

Select and Order Pages
Set Entitlements and Attributes

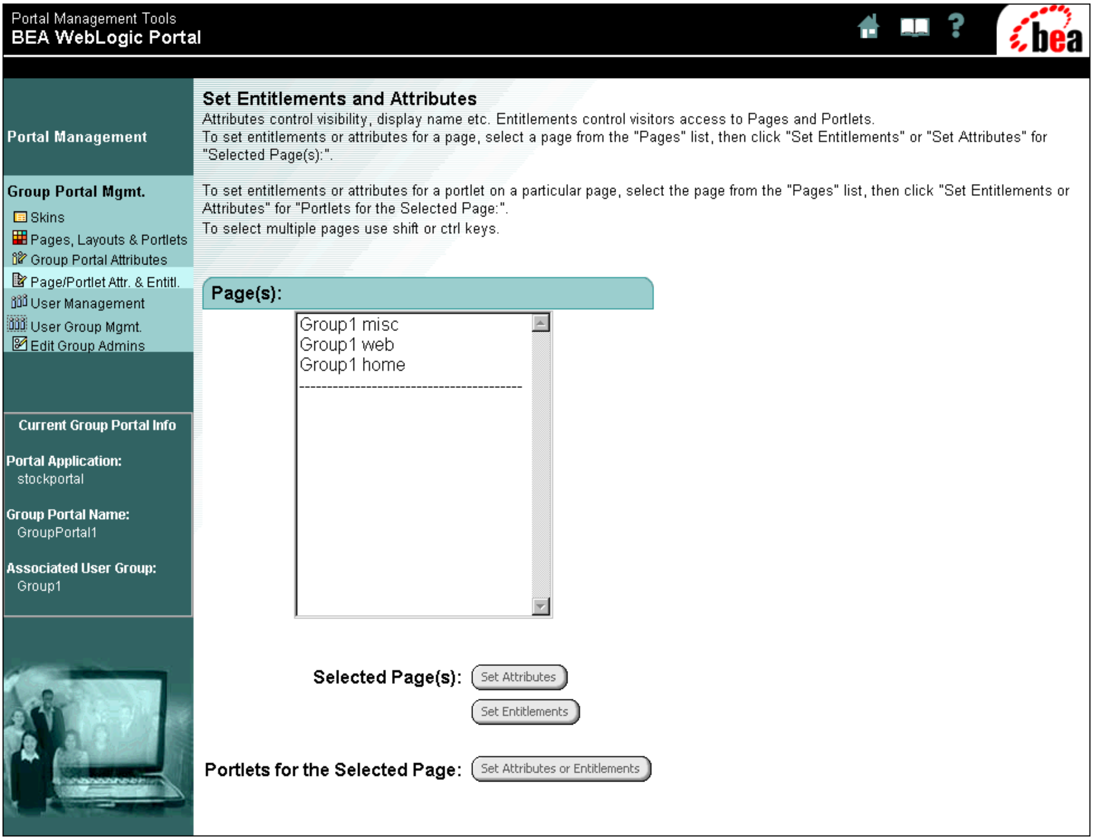
Available Pages

misc	Layouts	Portlets
web	Layouts	Portlets
home	Layouts	Portlets

Unused Pages

4. In the Pages, Layouts and Portlets page, click Set Entitlements and Attributes link. The Set Entitlements and Attributes page appears (Figure 4-33).

Figure 4-33 Set Entitlements and Attributes page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.

- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Set Entitlements and Attributes

Attributes control visibility, display name etc. Entitlements control visitors access to Pages and Portlets.
To set entitlements or attributes for a page, select a page from the "Pages" list, then click "Set Entitlements" or "Set Attributes" for "Selected Page(s):".

To set entitlements or attributes for a portlet on a particular page, select the page from the "Pages" list, then click "Set Entitlements or Attributes" for "Portlets for the Selected Page:".
To select multiple pages use shift or ctrl keys.

Page(s):

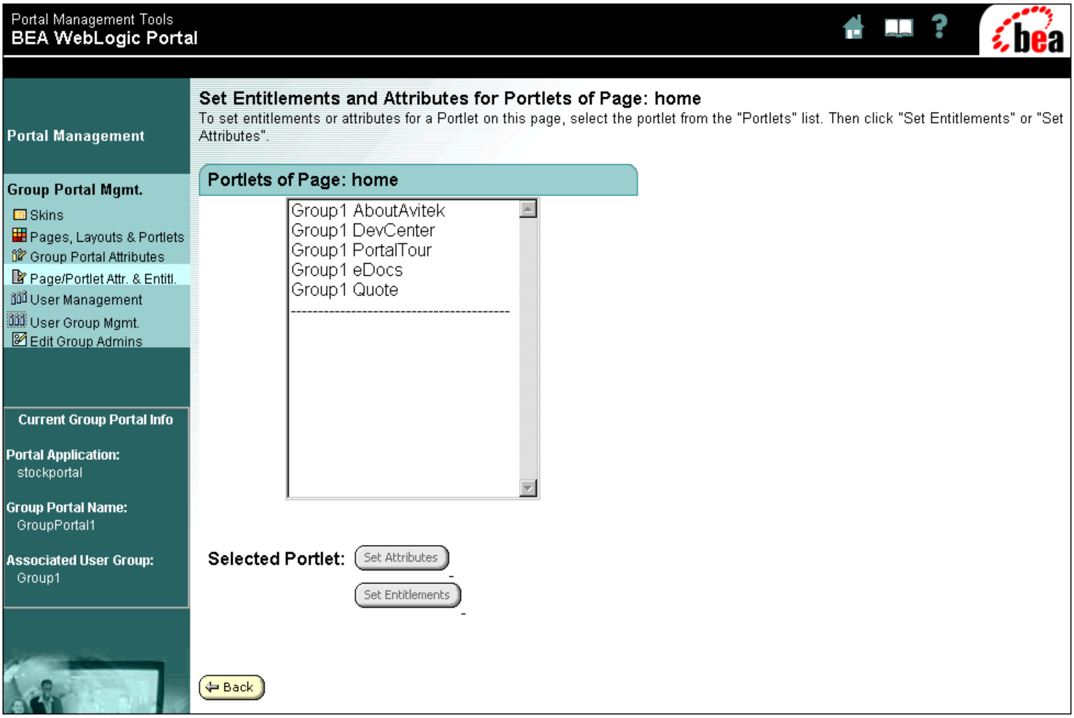
Group1 misc
Group1 web
Group1 home

Selected Page(s): Set Attributes
Set Entitlements

Portlets for the Selected Page: Set Attributes or Entitlements

5. Select the page for which you wish to set the portlet attributes and click Portlets for Selected Page: Set Entitlements and Attributes. The Set Entitlements and Attributes for Portlets page appears (Figure 4-34).

Figure 4-34 Set Entitlements and Attributes for Portlets page



Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

Set Entitlements and Attributes for Portlets of Page: home

To set entitlements or attributes for a Portlet on this page, select the portlet from the "Portlets" list. Then click "Set Entitlements" or "Set Attributes".

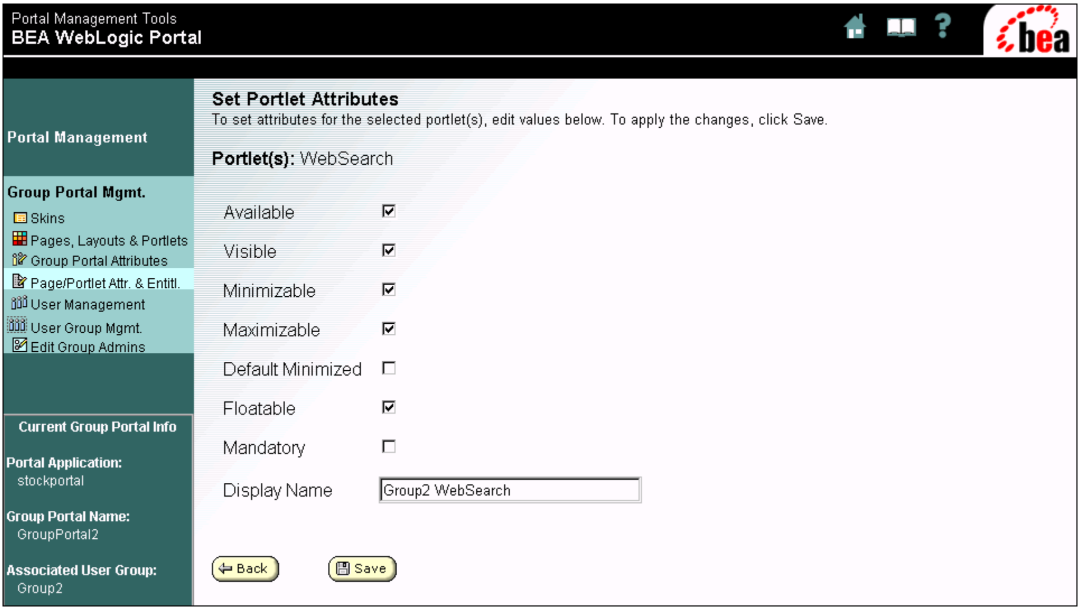
Portlets of Page: home

```
Group1 AboutAvitek
Group1 DevCenter
Group1 PortalTour
Group1 eDocs
Group1 Quote
----------------------------------------
```

Selected Portlet:

6. Select the portlet you wish to edit and click Set Attributes. The Set Portlet Attributes page appears (Figure 4-35).

Figure 4-35 Set Portlet Attributes page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal2

Associated User Group:
Group2

Set Portlet Attributes
To set attributes for the selected portlet(s), edit values below. To apply the changes, click Save.

Portlet(s): WebSearch

Available	☑
Visible	☑
Minimizable	☑
Maximizable	☑
Default Minimized	☐
Floatable	☑
Mandatory	☐
Display Name	Group2 WebSearch

7. You can set any of the following attributes:
   a. Available: The portlet will be available to the visitor.
   b. Visible: The portlet should be visible to the visitor default.
   c. Minimizable: The portlet can be minimized by the visitor.
   d. Maximizable: The portlet can be maximized by the visitor.
   e. Default Minimized: The portlet is minimized in the visitor default.
   f. Floatable: The portlet can be opened in a new browser.
   g. Mandatory: The visitor will always see this page. That is, they cannot remove it from their personalized portal.
   h. Display Name: The display name for the page, this is the name that site visitors will see.
8. Click Save when you are done.

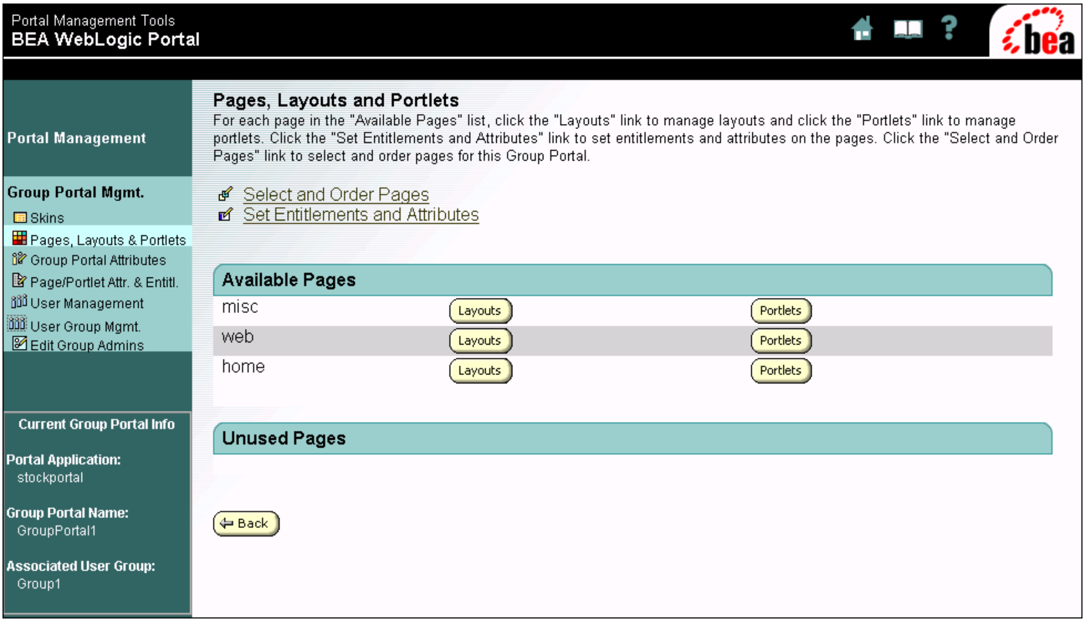
Setting Portlet Entitlements

1. Click the Portal Management icon in the home page. The Portal Management Home page appears.

2. Click on the Group Portal you wish to edit. The Group Portal Management home page appears.

3. In the Group Portal Management Home page, click Manage Portlets, Pages and Layouts. The Pages, Layouts, and Portlets page appears (Figure 4-36).

Figure 4-36 Pages, Layouts, and Portlets page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.
- Skins
- Pages, Layouts & Portlets
- Group Portal Attributes
- Page/Portlet Attr. & Entitl.
- User Management
- User Group Mgmt.
- Edit Group Admins

Current Group Portal Info

Portal Application:
stockportal

Group Portal Name:
GroupPortal1

Associated User Group:
Group1

Pages, Layouts and Portlets

For each page in the "Available Pages" list, click the "Layouts" link to manage layouts and click the "Portlets" link to manage portlets. Click the "Set Entitlements and Attributes" link to set entitlements and attributes on the pages. Click the "Select and Order Pages" link to select and order pages for this Group Portal.

Select and Order Pages
Set Entitlements and Attributes

Available Pages

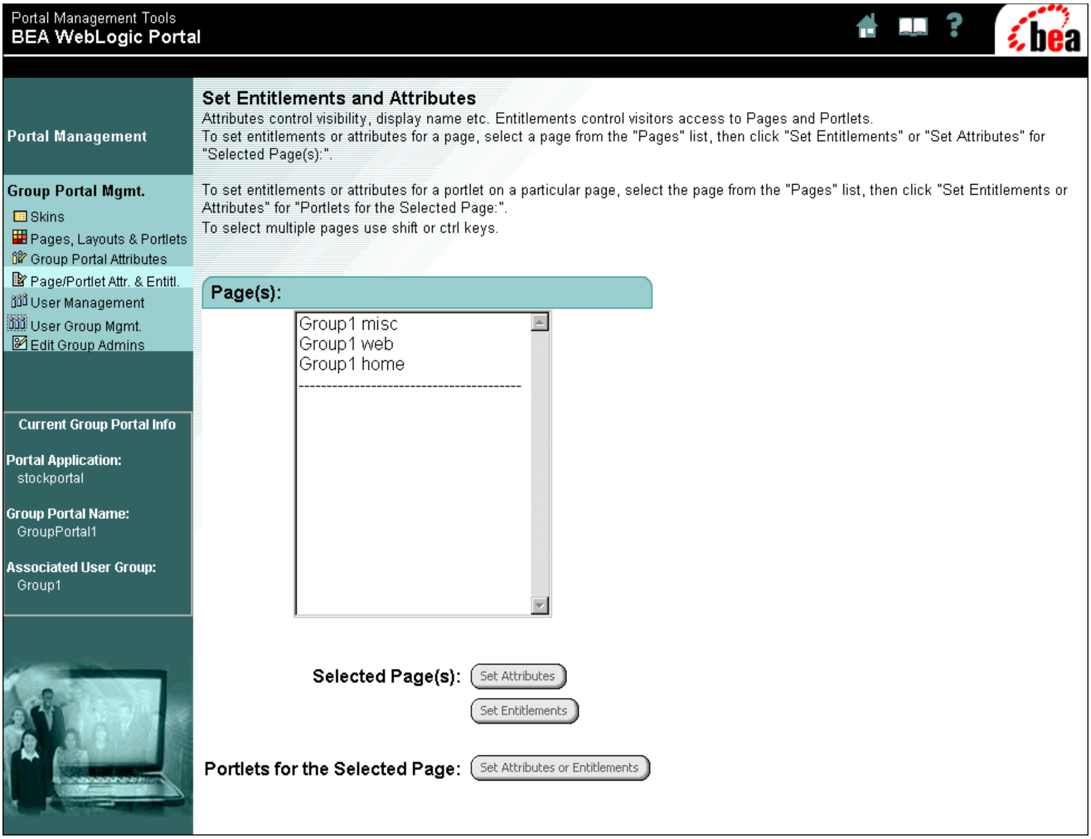
misc	Layouts	Portlets
web	Layouts	Portlets
home	Layouts	Portlets

Unused Pages

Back

4. In the Pages, Layouts and Portlets page, click Set Entitlements and Attributes. The Set Entitlements and Attributes page appears (Figure 4-37).

Figure 4-37 Set Entitlements and Attributes page



Portal Management Tools
BEA WebLogic Portal

Set Entitlements and Attributes

Attributes control visibility, display name etc. Entitlements control visitors access to Pages and Portlets.
To set entitlements or attributes for a page, select a page from the "Pages" list, then click "Set Entitlements" or "Set Attributes" for "Selected Page(s):".

To set entitlements or attributes for a portlet on a particular page, select the page from the "Pages" list, then click "Set Entitlements or Attributes" for "Portlets for the Selected Page:".
To select multiple pages use shift or ctrl keys.

Page(s):

Group1 misc
Group1 web
Group1 home

Current Group Portal Info

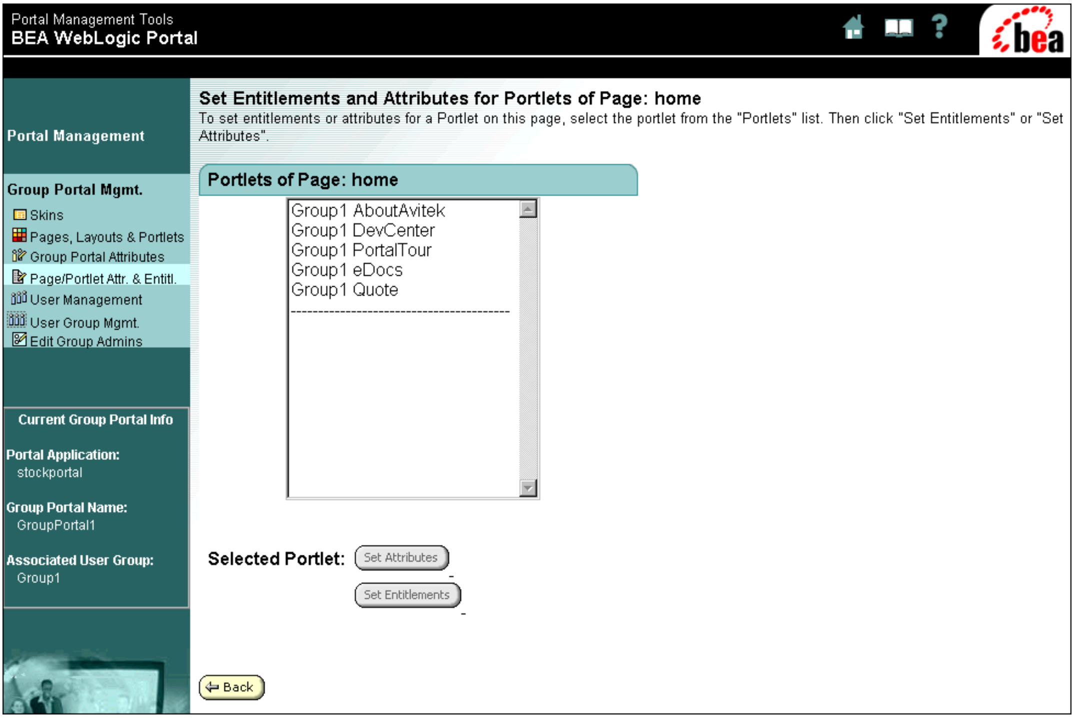
Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

5. Select the page for which you wish to set the portlet entitlements and click Portlets for Selected Page: Set Entitlements and Attributes. The Set Entitlements and Attributes for Portlets page appears (Figure 4-38).

Figure 4-38 Set Entitlements and Attributes for Portlets page



Portal Management Tools
BEA WebLogic Portal

Portal Management

Group Portal Mgmt.

Current Group Portal Info

Portal Application: stockportal

Group Portal Name: GroupPortal1

Associated User Group: Group1

Set Entitlements and Attributes for Portlets of Page: home

To set entitlements or attributes for a Portlet on this page, select the portlet from the "Portlets" list. Then click "Set Entitlements" or "Set Attributes".

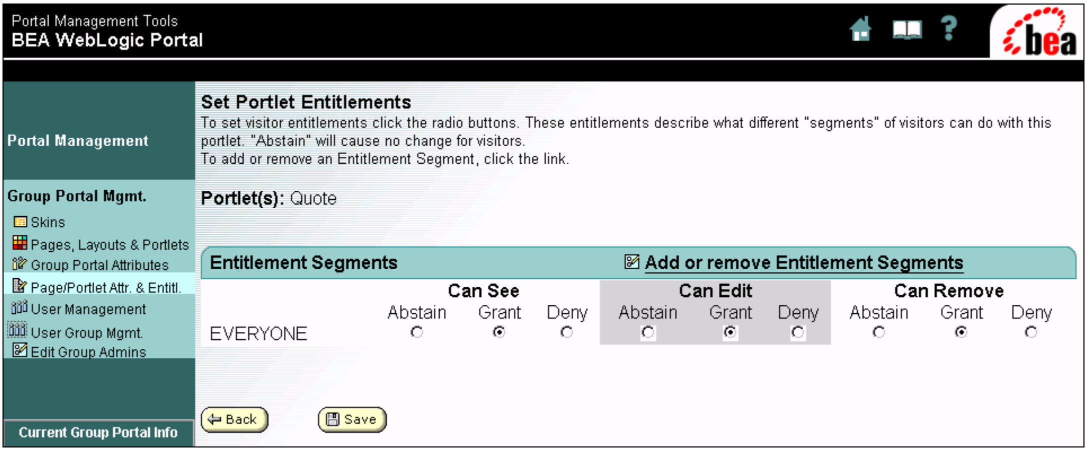
Portlets of Page: home

Group1 AboutAvitek
Group1 DevCenter
Group1 PortalTour
Group1 eDocs
Group1 Quote

Selected Portlet:

6. Select the portlet you wish to edit and click Set Entitlements. The Set Portal Entitlements page appears (Figure 4-39).

Figure 4-39 Set Portlet Entitlements



Set Portlet Entitlements

To set visitor entitlements click the radio buttons. These entitlements describe what different "segments" of visitors can do with this portlet. "Abstain" will cause no change for visitors.

To add or remove an Entitlement Segment, click the link.

Portlet(s): Quote

Entitlement Segments — **Add or remove Entitlement Segments**

	Can See			**Can Edit**			**Can Remove**		
	Abstain	Grant	Deny	Abstain	Grant	Deny	Abstain	Grant	Deny
EVERYONE	○	●	○	○	●	○	○	●	○

7. You can set any of the following attributes:

   a. Can See: Check the radio buttons to grant or deny availability of the portlet to the entitlement segment member.

   b. Can Edit: Check the radio buttons to grant or deny the entitlement segment members the ability to edit the portlet.

   c. Can Remove: Check the radio buttons to grant or deny the entitlement segment members the ability to remove the portlet from their view.

   Note: The entitlement segment “Everyone” represents every user in the system.

8. Click Save when you are done.

CHAPTER

5 Customizing Portals and Portlets

This topic begins with some basic developer tasks, and then introduces some more advanced developer-orientated concepts specific to the WebLogic Portal platform.

This topic includes the following sections:

- Creating a New Portal Web Application
- Creating Your Own Portlet
- In What Ways Can Portal be Customized?
- Look and Feel Customization
- Webflow and the Portal Platform

Creating a New Portal Web Application

This section explains using a copy of the sample portal web application as a template to create your own portal web application. This task requires working with the native file system, the WebLogic Server Console, and the E-Business Control Center.

- Copy and Rename the Web Application Files
- Deploy the Application in the WLP Server
- Edit Portal Metadata Using the EBCC

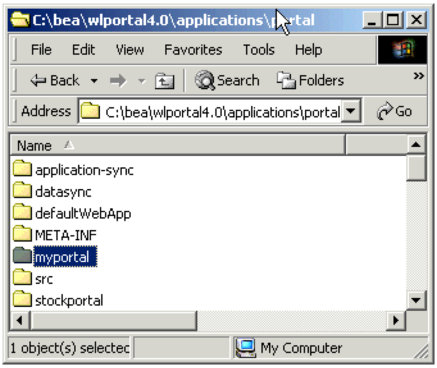
- Synchronize the Metadata Using the EBCC
- Verify Your Deployment

Copy and Rename the Web Application Files

1. Within the native file system, navigate to the following directory: `\\bea\wlportal4.0\applications\portal\`
2. Make a copy of the `stockportal` directory, placing it in the current directory.
3. Rename this new directory after the new portal web application you want to create. In this example, the new directory is called *`myportal`*.

   The new, renamed directory now contains all the J2EE application elements required of a web application, as shown in Figure 5-1.

Figure 5-1 The myportal Directory



C:\bea\wlportal4.0\applications\portal

Address C:\bea\wlportal4.0\applications\portal

application-sync
datasync
defaultWebApp
META-INF
myportal
src
stockportal

1 object(s) selectec

4. Copy the metadata files in Listing 5-1 to the following directory:

```
\bea\ebcc4.0\applications\portal\application-sync\webapps\myPortal\
```

Listing 5-1 Metadata Files

```
<BEA_HOME>\ebcc4.o\applications\portal\application-sync\webapps\stockportal\

portlet2.wf
```

`\portlet3.wf`

`\worldnews.wf`

Deploy the Application in the WLP Server

1. Start the WebLogic Portal server.
2. Navigate to the Console: `http://<hostname>:7501/console`

   `username: system`

   `password: <password you entered to boot server>`
3. From the Console, navigate to the following directory:

   `Deployments > Applications > Portal`

Configure a new Web Module

1. Right click on Portal and select “Edit Application Descriptor”
2. From the Web Server Console popup window, select WebModules
3. Right click WebModules and select “Configure a New WebModule”
4. From the Create a New WebModule Screen, enter the following information:

```
Module URI: myportal
Context URI: myportal
```

5. Click Create, then Apply
6. Navigate back to the Root node, StockPortal
7. With StockPortal node selected, click the Persist button
8. Close this editor window.

Deploy the New Web Application

1. Within WebLogic Console, navigate to the following directory:

```
Deployments > Applications > Portal
```

2. Select Portal
3. De-select the Deployed checkbox and click on Apply
4. Re-check the Deployed checkbox and click on Apply
5. Deployment will take several minutes. Once it has completed, refresh the browser
6. The **`myportal`** web application should now appear in the following directory:

`Deployments > Applications > Portal >` **`myportal`**

Target the Application to Your Server

1. Within WebLogic Console, navigate to the following directory:

`Deployments > Applications > portal > myportal >` **`Targets tab`**

2. From the Targets tab, move portalserver to the Chosen listbox and click Apply.

The J2EE web application deployment is complete.

Edit Portal Metadata Using the EBCC

This process uses the E-Business Control Center to edit metadata on the J2EE portions of your new portal web application.

Open Existing Portal Application

1. Start the E-Business Control Center and select `File > Open Application`
2. Navigate to the following directory:

`//EBCC4.0/applications/portal`

3. Open the portal application

Define a New Portlet

1. From the bottom left-hand corner of the E-Business Control Center, select the Presentation tab

2. Select the Portlet editor. A list of portlets should appear to the right
3. From the File menu at the top of the E-Business Control Center, select File > New > Presentation > Portlet
4. When the Portlet Editor appears, follow the rest of these instructions to reuse existing JSPs and image files for the new portlet.
5. Locate the `portlet3.wf` in the `PORTAL_HOME/applications/portal-project/application-sync/webapps/stockportal` directory.
6. Copy this file to the `PORTAL_HOME/applications/portal-project/application-sync/webapps/myportal` directory.
7. Sync the new data.
8. Define the Portlet as follows:
   - In the Content URL Field, enter the following URL:

   `/portlets/portlet3/welcome.jsp`

   - In the Titlebar URL Field, enter the following URL:

   `/framework/titlebar.jsp`

   - In the Icon URL Field, enter the following URL:

   `/portlets/dictionary/images/pt_dictionary.gif`

   - In the Description Field, enter the following URL:

   `My Sample Portlet`
9. Click on the dropmenu in the Webflow field and set this field to Portal3.
10. Disable the Login Required Checkbox. (If this checkbox is not disabled, the user will be required to log in order to view the portlet.)
11. Close the Portlet Editor Screen and save changes. Save the portlet with the following file name:

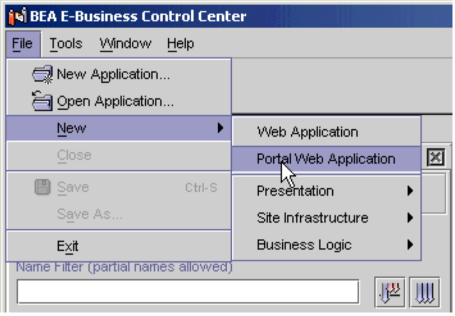
    `MyPortlet`
12. The new portlet should appear in the list of portlets as `MyPortlet`.

Define a New Web Application

This task creates new metadata for the new portal web application.

1. From the bottom left-hand corner of the E-Business Control Center, select the Presentation tab.
2. Select the Portal editor. A list of portals should appear to the right.
3. From the File menu at the top of the E-Business Control Center, select File > New > **Portal Web Application,** as shown in **Figure 5-2**.
4. Enter the name myportal and click OK.

Figure 5-2 Create a New Portal Web Application



Create a New Portal

This task edits the metadata for the new portal web application.

1. From the File menu at the top of the E-Business Control Center, select File > New > Presentation > Portal
2. When the Select Web Application dialog box appears, select the *`myportal`* web application and click OK
3. The Portal Editor will appear to the right
4. Expand the General tab and enter **`MyPortal`** in the Description field
5. Enter the following URLs:
   - In the Content URL field, enter the following URL:

/framework/page.jsp

- In the Header URL field, enter the following URL:

/framework/header.jsp

- In the Footer URL field, enter the following URL:

/framework/footer.jsp

- In the Vertical Navigation Bar URL field, enter the following URL:

/framework/vnav_bar.jsp

- In the Horizontal Navigation Bar URL field, enter the following URL:

`/framework/hnav_bar.jsp`

6. From the Portal Webflow dropmenu, select the Webflow called “portal”
7. From the Skins tab, move all available skins to the Selected Skins column
8. From the Layouts tab, move `twocolumn` to the Selected Layouts column
9. From the Portlets tab, move `MyPortlet` to the Selected Portlets Column
10. From the UserProfiles tab, move `StockPortal` to the Selected User Profiles column, and close the General tab
11. Expand the Pages tab and click **Add**
12. In the Portal Page Name Field, enter **`MyPortalPage`**
13. In the Default Image field, enter **home.gif**.
14. Move the layout and the portlets into their respective Selected listboxes.
15. Click OK. The new portal page should now appear in the Portal pages listbox in the E-Business Control Center.
16. Close the Pages tab.

Set Defaults on New Portal

1. Expand the Defaults tab, and select the **`Futurism`** skin.
2. From the Default portal pages listbox, highlight MyPortalPage.
3. From the Portlets listbox, highlight MyPortlet.

4. From the Default Layout dropmenu, select TwoColumn.
5. Click the Place Portlets button.
6. The Page Layout screen appears.
7. Enable the radio button in the second segment of the layout.
8. Highlight MyPortlet and click Add.
9. MyPortlet should now appear in the fourth segment of the layout.
10. Click OK.
11. Close the Defaults tab.
12. Click the Close(X) icon at the top right of the Portal editor.
13. From the “Portal is modified. Save it?” dialog box, click Yes.
14. In the Save As dialog box, enter the text `MyPortal.`
15. MyPortal and StockPortal should appear side-by-side in the E-Business Control Center list of portals.

Synchronize the Metadata Using the EBCC

Now that the metadata is edited, it must be moved to the server using a special Synchronization process.

Configure Connection Settings

The E-Business Control Center allows you to save different synchronization profiles as Connection Settings. This procedure shows how to configure one set.

1. From the Tools menu of the E-Business Control Center, select Connection Settings
2. From the Connection Settings popup window, navigate to the Connections tab and click the New button
3. Enter the following information:
   - Set the display name to `PortalDataSyncTest`

- Set Application Name Dropmenu to `Portal`
- Set the Server to `http://<hostname>:7501`

4. From the Details menu click OK
5. From, the Settings menu click OK

Synchronize Data

1. From the Tools menu of the E-Business Control Center, select Synchronize
2. In the login window that appears, enter the following credential:

   `username: system`

   `password: <password you entered to boot server>`
3. Click on Connect
4. When the Task Bar reaches 100% complete, click on Close

Modify Cookie Parameters in Deployment Descriptor

To prevent the new portal from using the generic cookie for authentication, take the following steps:

1. Open the `weblogic.xml` for the newly created portal. This file is in the `...\wlportal4.0\applications\portal\myportal\WEB-INF` directory.
2. Make the following edit to the deployment descriptor:

Listing 5-2 Editing Cookie Parameters for Deployment Descriptor

Replace this:

```
<session-param>
   <param-name>CookieName</param-name>
   <param-value>JSESSIONID_STOCKPORTAL</param-value>
</session-param>
<session-param>
```

```
   <param-name>CookiePath</param-name>
   <param-value>/stockportal</param-value>
</session-param>
```

with this:

```
<session-param>
   <param-name>CookieName</param-name>
   <param-value>JSESSIONID_webapp name</param-value>
</session-param>
<session-param>
   <param-name>CookiePath</param-name>
   <param-value>/webapp directory name</param-value>
</session-param>
```

Providing a unique pathname to which the browser sends cookies prevents other portals deployed in the same domain from using the same autologin cookies. After making this edit and re-starting WebLogic Portal, the new portal should use its own authentication cookie.

Verify Your Deployment

WebLogic Portal includes two ways to verify the deployment was successful - one is the obvious method of navigating to the URL and viewing the portal web application from the point of view of a Visitor. The other is to use the Data Repository Browser, shown below.

Verify Application Using Browser

You should see your newly-deployed portal web application if you navigate to

```
http://<hostname>:7501/myportal
```

Verify Data Synchronization Using Data Repository Browser

1. With the WebLogic Portal running, navigate to the following URL:

`http://<hostname>:7501/portalDataSync/index.html`

2. Enter the following credential:

   `username: system`

   `password: <password you entered to boot server>`

3. Click the Data Repository Browser link.
4. Click the Master Data Repository View link.
5. Verify that the myportal web application is visible within the Items list.
6. Verify that the newly created Portlet is visible.
7. Examine the XML file.

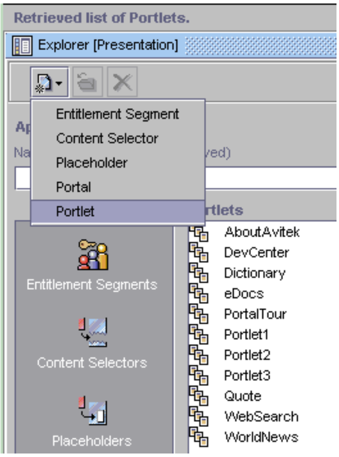
Creating Your Own Portlet

This section outlines how to create a new portlet and position it inside an existing portal.

Define a New Portlet

1. Start E-Business Control Center and open the portal application, as shown in Figure 5-1. The Default location is `BEA_HOME\wlportal4.0\applications\portal-project.`
2. Select the Presentation tab at the bottom right corner of the E-Business Control Center.
3. Click on Portlets, and then click on the Create button, as shown inFigure 5-3.

Figure 5-3 Create New Portlet Button



4. When the Portlet Editor window appears on the right, enter the values listed in Table 5-1.

5. Click the Save button, shown in Figure 5-4, using the name TestPortlet when prompted.

Table 5-1 New Portlet Settings

Field Name	Purpose	Example Value
Description	Description of Portlet	Test Portlet
Content URL	Default body JSP	/portlets/test/portlet.jsp
Header URL	Default header JSP	/portlets/test/header.jsp
Alternate Header URL	Used for Maximized state	/portlets/test/altheader.jsp
Footer URL	Default footer JSP	/portlets/test/footer.jsp
Alternate Footer URL	Used for Maximized state	/portlets/test/altfooter.jsp
Titlebar URL	Default titlebar JSP*	/framework/titlebar.jsp
Banner URL	Banner JSP	/portlets/test/banner.jsp
Icon URL	Image for titlebar icon	/portlets/test/images/icon.gif

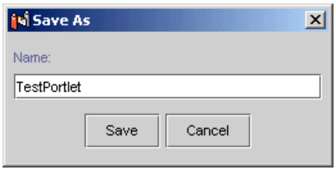
Table 5-1 New Portlet Settings

Field Name	Purpose	Example Value
Edit URL	Portlet editor JSP	/portlets/test/edit.jsp
Help URL	Portlet help JSP	/portlets/test/help/help.jsp
Maximize URL	Maximized body JSP	/portlets/test/max.jsp
Webflow	Portlet Webflow	None
Editable	Editor is available	True
Helpable	Help is available	True
Login Required	Only displayed if user logs in	False
Mandatory	Portlet must appear	False
Floatable	May be opened in separate window	True
Minimizable	Portlet may be minimized	True
Maximizable	Portlet may be maximized	True
Defaults minimized	Appears minimized by default	False

Note: The following clarifications apply to Table 5-1:

- Notice that the `titlebar.jsp` is kept in the `/framework` directory because it detects the portlet state at runtime, and includes one of the following JSPs: `normal_titlebar.jsp`, `edit_titlebar.jsp`, `maximize_titlebar.jsp`, or minimize_titlebar.jsp. These also reside in the `/framework` directory.
- For simple portlets the optional fields may be ignored. In this example all of them are used for demonstration purposes.
- The paths are relative to the web application directory, and by convention the portlets are placed under the /portlets directory, with subdirectories for each portlet.

Figure 5-4 Save as TestPortlet



Name: TestPortlet

Place New Portlet into Existing Portal Definition

In order to see the new portlet, references to it need to be placed in the following sections of the portal definition:

- General Setting for the Portal
- Specific Pages in the Portal
- Default Settings for the Portal

General Setting for the Portal

1. Select the Presentation tab at the bottom right corner of the E-Business Control Center.
2. Click on Portals, and then double-click on StockPortal to open the editor.
3. Click on the General pane to expand it and click on the Portlets tab.
4. Move TestPortlet from the Available to the Selected column using the Add >> button.
5. Click on OK and then close the General pane by clicking on it.

Specific Pages in the Portal

1. Click on the Pages pane.
2. Select the Home page and click on Edit.

3. Move TestPortlet from the Available to the Selected column using the Add >> button.
4. Click on OK and then close the Pages pane by clicking on it.

Default Settings for the Portal

1. Click on the Defaults pane.
2. Select the Home page from the list and click on Place Portlets at the bottom of the window.
3. In the Place Portlets dialog, select the middle column.
4. Click on Add to move the TestPortlet from the list on the left to the middle column.
5. Click on OK and then close the Pages pane by clicking on it.

Deploy the New Portlet

The metadata for the new portlet has now been edited. Next, deploy the metadata as well as the J2EE application files.

Synchronize the Metadata to the Server

Use the E-Business Control Center to synchronize this metadata to the server. This section outlines the basic steps to perform this task. For detailed instructions on operating the E-Business Control Center, consult the *Guide to Using the E-Business Control Center*.

Note: Make sure that in your Synchronization Settings, the realm is set to weblogic, and that the application is set to portal.

Install the J2EE Application Files for New Portlet

At this point in the procedure, there is metadata on the server that defines a portal called myportal. Since it mostly reuses existing J2EE files, most parts of this new portal web application should already be visible, with one exception: The portlet we have defined and named TestPortlet has no J2EE application files yet.

For this example, we'll copy an existing portlet directory and alter the existing files to construct the TestPortlet.

1. Navigate to the following directory:

   `BEA_HOME\wlportal4.0\applications\portal\stockportal\portlets`

2. Make a copy of the portlet1 directory and paste it into the portlets directory.

3. Rename the directory `test`.

4. Open the new directory: the filenames should match those defined in the EBCC for the TestPortlet.

Verify the Portlet is Deployed

Navigate to the following directory:

`http://<host>:7501/stockportal/`

The new test portlet should appear below the E-Docs portlet in the center column of the Home page of the sample portal.

Note: To make this new portlet available for authenticated visitors, the Group Portal attributes must be set so the portlet is available and visible to the user. The simplest way to do this is to use static personalization by setting entitlements on the portal to be available to any member of the group to which the user belongs. For instructions on setting entitlements, consult the section called "Setting Entitlements" in the Portal Administration Tools chapter.

Edit the Copied Portlet Files

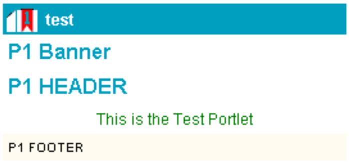
Now that the JSPs are installed, and the metadata has been edited and deployed so that the framework can see your portlet, changing the content is a matter of editing the files that make up your portlet. The JSPs in this example are very simple, so you can edit them with something like Notepad. Also, notice that we're editing JSPs from within the server's native file system.

Note: While performing these exercises, leave your browser open to the stockportal page between editing the JSPs. When you re-save the edited portlet files, you can simply re-load the page using the same session.

Edit Static HTML

The JSP files that make up your new portlet include some simple HTML. Importantly, portlets do not need to be complete HTML documents. Open the `portlet.jsp` file and replace the "Portlet1" text with "This is the Test Portlet". Save the file and reload the stockportal in the browser. The portlet should be re-rendered and appear as shown in Figure 5-5.

Figure 5-5 Edited Portlet



test

P1 Banner

P1 HEADER

This is the Test Portlet

P1 FOOTER

About Scriptlets and JSP Tags

A JSP may contain static HTML, as demonstrated in the preceding slides, and dynamic logic in the form of scriptlets and JSP tags. This section introduces some scriptlet and JSP fundamentals for those who may not be familiar with the servlet model.

A scriptlet is a section of valid Java code typically used to process input and generate output for the JSP. In most cases this is display logic, because business logic is best handled by components such as EJBs.

The Java code in a scriptlet is processed by the JSP compiler to create the body of a servlet that will be called when the URL for this JSP is requested. Servlets are based on a request-response architecture that is ideally suited for handling HTTP requests such as those from HTML browsers.

The generated servlet is Java source code that will in turn be compiled. The resulting Java class file will be stored in a location by the server and loaded when the servlet is accessed. Errors may occur in three stages, including JSP compilation, Java compilation, and execution of the resulting servlet class. These errors may be displayed in the browser as part of the portlet body, the console where the server is running, or the server log files.

For more information on servlets and JSPs, consult the J2EE specification at `http://java.sun.com/`.

Using Java Scriptlets Within a Portlet

This section outlines how to insert some simple code into the portlet.jsp file that handles a variable called username, and will display it in the body of the portlet.

Note: For this example, open a new browser and navigate to the following URL:

```
http://<hostname>:7501/stockportal/
```

1. In the native file system, navigate to the `/portlets/test` directory and open the `portlet.jsp` file in Notepad.
2. Append the following snippet to the end of the file:

   ```
   <%
    String userName = "";
    %>
   The user name is: <%= userName %>
   <br>
   <p>
   ```

3. Save the file and re-load the portal in your browser.
4. Notice that the phrase “`The user name is:`” appears in the window, but the userName object returns nothing. (In order to make use of this variable, this object must be populated with the current portlet visitor. Of course, if the visitor has not authenticated, we cannot return a username.)

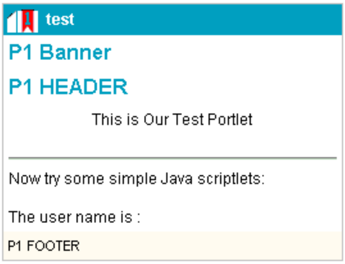
5. In the native file system, navigate to the `/portlets/test` directory and open the `portlet.jsp` file in Notepad.
6. Replace the contents of the file with the code in Listing 5-3:

Listing 5-3 Displaying the User Name

```
<center>

This is Our Test Portlet

</center>

<br>

<hr>

Now try some simple Java scriptlets:

<br>

<br>

<%

// Create a new String variable and set the value to an empty string.

  String userName = "";

/*

To get the actual user name, first get the java.security.Principal
object from the javax.servlet.http.HttpServletRequest object using
the getUserPrincipal() method.  The request object is available to
the JSP directly.  For the Principal object, use the full package
name.

*/

java.security.Principal principal = request.getUserPrincipal();

/*

If the principal object is null then the user has not logged in.
For this example, ignore the not-logged in case by using an if
statement to only process the value if it is not null.

*/

  if (principal != null)
  {

/*

Now get the user name as a String using the getName() method for
the java.security.Principal object.

*/

    userName = principal.getName();

  }

%>

<%--

Display the value of the userName variable using a scriptlet.  Note
that the scriptlet is embedded in the HTML and is denoted by the
<%= %> block. The "=" sign indicates that the scriptlet should be
evaluated as string and output in the HTML.  It can be any valid
Java expression, including a variable or method call.

--%>

The user name is : <%= userName %>

<br>
```

7. Save the file and re-load the portal in your browser.

Figure 5-6 Scriptlet without Authenticated Visitor



test

P1 Banner

P1 HEADER

This is Our Test Portlet

Now try some simple Java scriptlets:

The user name is :

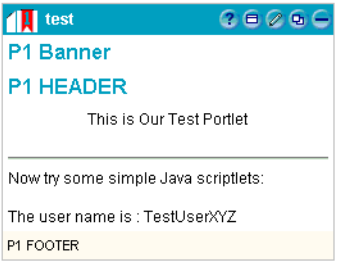
P1 FOOTER

8. The portlet should appear as it does in Figure 5-6.

9. Login as the following test user: **`acme/acme`**.

10. Notice that the phrase "The user name is:" is now followed by the name of the new user, as shown in Figure 5-7.

Figure 5-7 Scriptlet Showing Authenticated Username



P1 Banner

P1 HEADER

This is Our Test Portlet

Now try some simple Java scriptlets:

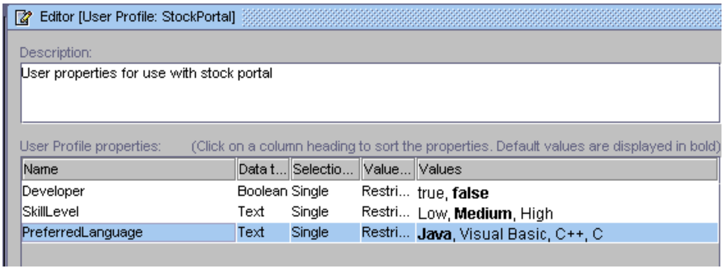
The user name is : TestUserXYZ

P1 FOOTER

Displaying User Profile Values

This section outlines how to use a simple tag to insert a value from a user profile into your portlet. Figure 5-8 shows that the property we're going to expose is already set in the stock portal sample data.

Figure 5-8 Viewing the PreferredLanguage Property



Description:

User properties for use with stock portal

User Profile properties: (Click on a column heading to sort the properties. Default values are displayed in bold)

Name	Data t...	Selectio...	Value...	Values
Developer	Boolean	Single	Restri...	true, **false**
SkillLevel	Text	Single	Restri...	Low, **Medium**, High
PreferredLanguage	Text	Single	Restri...	**Java**, Visual Basic, C++, C

Exposing a property such as PreferredLanguage only requires placing a few of lines of code inside a portlet. This is an example of the efficiency of using JSP tag libraries to develop reusable components.

1. First, use the E-Business Control Center to look at the user property:

   Select the Site Infrastructure tab at the bottom right corner of the E-Business Control Center.

   Click on User Profiles, then double-click on the StockPortal profile.

Notice the default setting for the PreferredLanguage property, shown in Figure 5-8.

2. Now, in the native file system, navigate to the `/portlets/test` directory and open the `portlet.jsp` file in Notepad.
3. Replace the contents of the file with the code in Listing 5-4.
4. Save the `portlet.jsp` and refresh the browser. (If not logged in, do so as the following test user: **`acme/acme`**.
5. Notice the PreferredLanguage is Java.

Listing 5-4 Getting a Property from User Profile

```
<center>

This is Our Test Portlet

</center>

<br>

<hr>

Now try some simple Java scriptlets:

<br>

<br>

<%

// The following code is from the earlier example minus the comments

  String userName = "";

  java.security.Principal principal = request.getUserPrincipal();

  if (principal != null)

  {

    userName = principal.getName();

  }

%>
```

```
The user name is : <%= userName %>

<br>

<hr>

Now try some simple JSP tags:

<br>

<br>

<%-- Import the User Management tag library. --%>

<%@ taglib uri="um.tld" prefix="um" %>

<%--

Get the PreferredLanguage property as a String from the StockPortal
user profile property set and store the value in the
preferredLanguage variable.

 --%>

<um:getPropertyAsString propertyName="PreferredLanguage"

 propertySet="StockPortal" id="preferredLanguage"/>

<%-- Display the preferredLanguage variable in a scriptlet. --%>

The preferred language is: <%= preferredLanguage %>

<br>
```

Note: About the code in Listing 5-4:

- The first line in bold imports the JSP tag library for the User Management tags.
- The second bolded section (um:getPropertyAsString tag) gets the PreferredLanguage property in the StockPortal property set, storing it in the preferredLanguage variable.
- The last line (also in bold) displays the value of the preferredLanguage variable.

Changing User Profile Values

Using the same scriptlet code in Listing 5-4, let’s change the value of the property for the test user **`acme/acme`**.

1. To change the property value for this user, navigate to the following URL:

   `http://<hostname>:7501/portalTools/index.jsp`

2. Login as **`Administrator/<password>`**
3. Click on the User Management home page, and select Users.
4. Search for acme, and click on the acme user from the list.
5. From the dropmenu on the left, select the StockPortal property set and click the red Search button.
6. On the right tab next to the PreferredLanguage property, click on Edit.
7. From the Edit Property Values page, set the value to “C” and click Save.
8. If a “Property set correctly” message appears, go back to the browser where the acme user is logged into stockportal.
9. Refresh the browser. The value of the PreferredLanguage should now be “C”.

For More Information on JSP Tags and User Profiles

WebLogic Portal 4.0 provides a number of other JSP tags you can use to bring rich functionality into your portlets quickly and easily. Most of the major functionality provided by the WebLogic Portal platform is available via JSP tags. For more a complete list of available portal-specific JSP tags, consult Chapter 6, “Portal Management JSP Tag Library Reference.”

To learn more about how user profiles are created and managed in the WebLogic Portal platform, consult Chapter 7, “Creating and Managing Users,” in the *Guide to Building Personalized Applications*.

Several examples that use personalization JSP tags are installed with the out of the box product. You can view these working examples by navigating to the following URL:

`http://<hostname>:7501/p13n`

Note: Login as **`acme/acme`**.

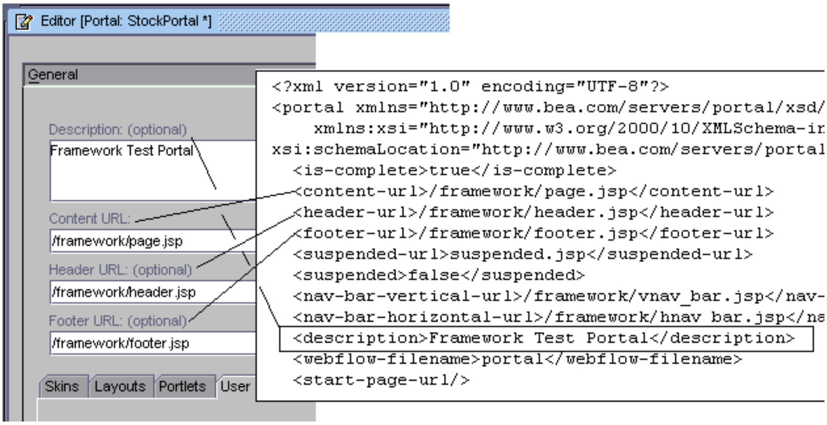
In What Ways Can Portal be Customized?

The out of the box installation of WebLogic Portal includes a sample portal Web application which is intended to be used as a building block for your custom portals. The most basic customization involves copying and modifying this sample portal. Even if you start from scratch, many of the supporting files are already created for you. This topic explains these simple tasks, as well as creating your own portlets, applying Webflows to individual portlets, and setting up a development environment where several Business Engineers can stage their work simultaneously on a single development instance of WebLogic Portal without overwriting each other's files.

How Portals Are Defined

Portals are constructed of JSPs, image files and whatever other HTML or Javascript is needed to achieve the desired functionality. Within the WebLogic Portal platform, a portal is defined using XML metadata files maintained by the Portal Module in the E-Business Control Center, as shown in Figure 5-9.

Figure 5-9 Portal.portal XML file



Description: (optional) Framework Test Portal

Content URL: /framework/page.jsp

Header URL: (optional) /framework/header.jsp

Footer URL: (optional) /framework/footer.jsp

```
<?xml version="1.0" encoding="UTF-8"?>
<portal xmlns="http://www.bea.com/servers/portal/xsd/
    xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-ir
xsi:schemaLocation="http://www.bea.com/servers/portal
  <is-complete>true</is-complete>
  <content-url>/framework/page.jsp</content-url>
  <header-url>/framework/header.jsp</header-url>
  <footer-url>/framework/footer.jsp</footer-url>
  <suspended-url>suspended.jsp</suspended-url>
  <suspended>false</suspended>
  <nav-bar-vertical-url>/framework/vnav_bar.jsp</nav-
  <nav-bar-horizontal-url>/framework/hnav bar.jsp</na
  <description>Framework Test Portal</description>
  <webflow-filename>portal</webflow-filename>
  <start-page-url/>
```

Comparing the entries in this portal definition file to those appearing in the E-Business Control Center Portal module, you can infer that the XML files are metadata, containing references to the J2EE application resources that make up the application itself.

The entries in this portal definition file tell the platform where to go for each of the following resources:

Entry	**Definition**
General Settings	
■ Description	Optional text string describing portal
■ Content URL	Location of Content portion of portal
■ Header URL	Location of optional portal header
■ Footer URL	Location of optional portal footer
■ Vertical Navigation Bar URL	Location of vertical navigation bar
■ Horizontal Navigation Bar URL	Location of horizontal navigation bar
■ Portal Webflow	Name of portal Webflow
Add and Remove portlet elements	Move portlet elements from Available (cannot be used) to Selected (can be used)
■ Skins ■ Layouts ■ Portlets	These design components can be excluded (marked as Available) from the current portal deployment or included in the current portal deployment (Selected)
■ User Profiles	These optional personalization mechanisms can be applied to individual portlets or to entire portals
Pages	Manage Pages within a Portal
■ Add	A new Portal page (sometimes called a page view) is designated by specifying a unique name and at least one layout and at least one portlet. Optionally, default, rollover and selected images can be designated here as well

Entry	Definition
■ Edit, Remove	The parameters of an existing page can be changed, and existing existing Portal pages can be deleted
Defaults	These portal parameters are set for each portal visitor, but are overwritten by any customization performed by the visitor.
■ Display Name	The name displayed in the browser's title bar - for example, "BEA Basic Portal"
■ Default Skin	The default look and feel template associated with this portal
■ Default Portal Page	The default portal page associated with this portal
■ Portal Pages	Subset of all existing portal pages can be made available
■ Nav-bar Orientation	Horizontal or Vertical
■ Portal Page Up/Down	Clicking Up or Down button re-orders selected page view
■ Portal Page Defaults	Set Rendering Parameters for selected Page View
■ Default Layout	Designate an existing layout as the default
■ Place Portlets	Use Page Layout tool to position portlets within selected layouts
■ Edit Portlet Defaults	Enter default values for the following settings: ■ Mandatory - overrides all customization settings such as "visible" as well as layout preferences ■ Maximizable - portlet can be maximized on visitor desktop ■ Minimizable - portlet can be minimized ■ Defaults minimized - portlet is initially rendered minimized

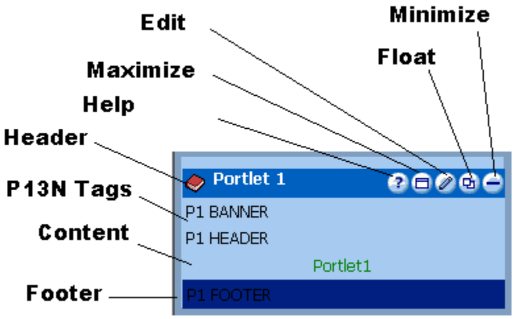
Note: Default settings are used to initialize the setup for a visitor. They are overwritten when the visitor chooses "customize my portal."

How Portlets Are Defined

Portlets can include some combination of the following presentation elements:

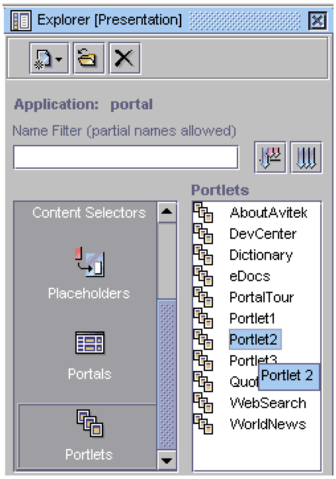
- header - Optional
- content - Either specified with URL or determined by Webflow
- footer - Optional
- edit - Optional
- help - Optional
- minimize - Optional
- maximize - Optional

Figure 5-10 Demo Portlet with Optional Controls



Using the E-Business Control Center, you can view a portlet definition by selecting Portlets in the Presentation tab, as shown in Figure 5-11.

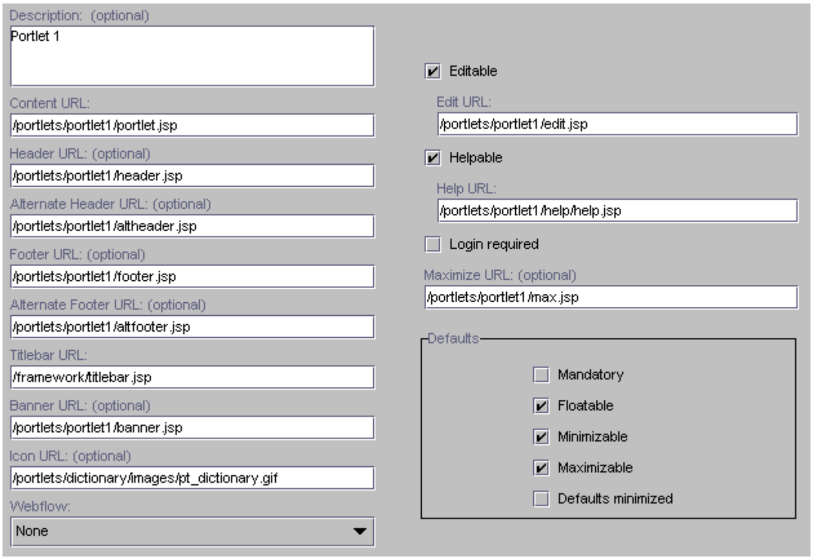
Figure 5-11 Editing a Portlet



Application: portal

When the portlet is selected, the definition editor appears, exposing the parameters that make up a portlet in WebLogic Portal. Figure 5-12 shows the definition settings for Portlet2 in the sample portal.

Figure 5-12 Portlet definition



Description: (optional)
Portlet 1

Content URL:
/portlets/portlet1/portlet.jsp

Header URL: (optional)
/portlets/portlet1/header.jsp

Alternate Header URL: (optional)
/portlets/portlet1/altheader.jsp

Footer URL: (optional)
/portlets/portlet1/footer.jsp

Alternate Footer URL: (optional)
/portlets/portlet1/altfooter.jsp

Titlebar URL:
/framework/titlebar.jsp

Banner URL: (optional)
/portlets/portlet1/banner.jsp

Icon URL: (optional)
/portlets/dictionary/images/pt_dictionary.gif

Webflow:
None

☑ Editable

Edit URL:
/portlets/portlet1/edit.jsp

☑ Helpable

Help URL:
/portlets/portlet1/help/help.jsp

☐ Login required

Maximize URL: (optional)
/portlets/portlet1/max.jsp

Defaults

☐ Mandatory
☑ Floatable
☑ Minimizable
☑ Maximizable
☐ Defaults minimized

Portlets, not unlike portals, are made up of JSPs, images and other resources, and are defined by XML metadata files. Where portal definitions determine which resources (skins and portlets) can be included, portlet definitions are more self-referential: They determine the components that make up the portlet, as well as certain presentation and navigation behaviors. Many of these definition elements are optional, and several can be determined within the portlet definition or within a Webflow specific to that portlet namespace.

Entry	Definition
■ Description	Optional text string describing portlet
■ Content URL	Location of content portion of portlet (can be declared within Webflow)
■ Header URL	Location of optional portal header
■ Alternate Header URL	Optional URL for Maximization
■ Footer URL	Location of optional portal footer
■ Alternate Footer URL	Optional URL for Maximization
■ Titlebar URL	Location of titlebar URL
■ Banner URL	Location of optional banner URL
■ Icon URL	Location of optional Icon URL
■ Portlet Webflow	Name of optional portlet Webflow
■ Editable	If checked, requires Edit URL (or declaration within Webflow)
■ Helpable	If checked, requires Help URL
■ Login required	Optional
■ Maximize URL	Optional URL for separate maximize layout (can be declared within Webflow)

Entry	Definition
■ Edit Portlet Defaults	Enter default values for the following settings: ■ Mandatory - overrides all personalization settings such as “visible” as well as layout preferences ■ Moveable - allows visitor to relocate portlet ■ Floatable - portlet can be viewed in its own window ■ Maximizable - portlet can be maximized on visitor desktop ■ Minimizable - portlet can be minimized ■ Defaults minimized - portlet is initially rendered minimized

How Portal Resources Are Defined

The JSPs, HTML pages, image files and directory structures for the portal web application may also be accompanied by CSS files and other sets of images used to provide skins for the new application. After these objects are installed on the server, the Business User can use the E-Business Control Center Portal Module to create references to these resources for the WebLogic Portal server.

The E-Business Control Center Portal module provides a graphical interface for specifying key parameters about a portal, and defining which resources are available to the portal, such as pages, layouts, skins and portlets. These resources can be edited offline, then synchronized later.

Define a Portal

Steps to define a portal resources are covered in Chapter 3, “Using the E-Business Control Center Portal Tool.”

Look and Feel Customization

The appearance of HTML pages can be customized in many ways. WebLogic Portal provides a number of access points to replace the out-of-the-box layouts, images and HTML page styles.

The workflow is similar to other tasks you perform using the portal framework: As with portlets and other portal Web application objects, the application files (in this case, layout JSPs, cascading style sheet files and images) are created and migrated into the server's native file system using your choice of tools. The metadata is then edited and migrated using the E-Business Control Center. Finally, the application resources you have created and made available on the portal server are administrated and managed using the WebLogic Portal Administration Tools.

This section explains the implementation of layouts and skins in the out-of-the-box installation, and outlines the basic procedures for creating custom layouts and skins. After reading this section, an HTML developer familiar with cascading style sheets should be able to customize the appearance, layout and navigation buttons for portals and portlets, and be able to manage the inheritance of these look and feel objects within portal web applications.

How Layouts Are Implemented

Out of the box, WebLogic Portal provides the following five default layouts:

- twcolumn
- threecolumn
- fourcolumn
- spanning
- spanningthreecolumn

These layouts consist of two elements: a simple `template.jsp` file and a thumbnail image. The JSPs declare a set of placeholders and then insert these placeholders within the HTML fragment. The JSP resides within a layout folder, and its corresponding thumbnail image is displayed by the EBCC while the metadata is being edited. The layout folders are located in the following directory:

```
<portal_home>\applications\portal-project\library\portal\layouts
```

Create a Custom Layout

Creating a custom layout is fairly straightforward. This example uses a copy of an existing layout, shown in Listing 5-5.

Listing 5-5 template.jsp for fourcolumn layout

```
<%@ taglib uri='ren.tld' prefix='layout' %>

<layout:placePortletsinPlaceholder
placeholders="left,leftcenter,rightcenter,right" />


<table width="100%" border="0" cellspacing="0" cellpadding="0">

  <TR>

    <TD WIDTH="25%" VALIGN="TOP">

      <layout:render section='left'/>

    </TD>

    <TD WIDTH="25%" VALIGN="TOP">

      <layout:render section='leftcenter'/>

    </TD>

    <TD WIDTH="25%" VALIGN="TOP">

      <layout:render section='rightcenter'/>

    </TD>

    <TD WIDTH="25%" VALIGN="TOP">

      <layout:render section='right'/>
      </TD>
    </TR>
</TABLE>
```

1. Make a copy of the fourcolumn layout folder and its contents.
2. Replace the contents of template.jsp with the code in Listing 5-6.
3. Rename the folder “fivecolumn”, and rename the image inside fivecolumn.gif. Use a simple graphic editor to alter the four-column thumbnail to it has five columns.
4. Open the template.jsp in notepad. Notice the reference to a JSP tag library called “ren.tld”. This tag library encapsulates portal rendering code, and takes the following parameters:

   `layout:placePortletsinPlaceholder`

   `placeholders=''<placeholder names>`

   With the layout tag called and its placeholders declared, all that remains is to place calls to the layout tag within the HTML table. Notice that placeholders are referred to as “section” when they are used inside the HTML.
5. Place the directory, including the GIF and the JSP file, into the following location:

   `<portal_home>applications\portal\stockportal\framework\layouts`

   Notice that a new placeholder has been declared called `center`. For more detail on how these placeholders are used by the framework, consult “The Auto Placer and “Homeless” Portlets.”

Note: Make sure the thumbnail is named the same as the layout directory.

Listing 5-6 template.jsp for fivecolumn layout

```
<%@ taglib uri='ren.tld' prefix='layout' %>

<layout:placePortletsinPlaceholder
placeholders="left,leftcenter,center,rightcenter,right" />
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
  <TR>
    <TD WIDTH="20%" VALIGN="TOP">
      <layout:render section='left'/>
    </TD>
    <TD WIDTH="20%" VALIGN="TOP">
      <layout:render section='leftcenter'/>
    </TD>
    <TD WIDTH="20%" VALIGN="TOP">
      <layout:render section='center'/>
    </TD>
    <TD WIDTH="20%" VALIGN="TOP">
      <layout:render section='rightcenter'/>
    </TD>
    <TD WIDTH="20%" VALIGN="TOP">
      <layout:render section='right'/>
    </TD>
  </TR>
</TABLE>
```

How Skins Are Implemented

Out of the box, WebLogic Portal provides the following five default skins:

- blueinfinity
- brightlight
- coologic

- futurism
- lumiereorange

Skins are made up of .css files and some images. The thumbnail images for the EBCC reside in the following directory:

```
<portal_home>applications\portal-project\library\portal\skins
```

The folders containing images and css files reside in the following directory:

```
<portal_home>applications\portal\stockportal\framework\skins
```

For stockportal, the location of these resources is stipulated in the following file:

```
<portal_home>\applications\portal\stockportal\framework\resourceU
RL.inc
```

Making Custom Skins Available to the Portal Framework

To create custom skins, have an HTML designer create cascading style sheet files and HTML image files, and package them according to the skins directories in the following folder:

```
<portal_home>applications\portal\<yourportal>\framework\skins
```

1. From the `<portal_home>applications\portal\<yourportal>\framework\skins` directory, copy an existing skin directory to use as a template. For example, copy `skins\coologic` to `skins\newskin`.
2. Modify `main.css` and the images, or copy in the images and css files created by a designer.
3. Run the portal.
4. Take a screenshot of the portal and update the thumbnail gif in the following directory:

```
portal-project\library\portal\skins
```

5. From the <portal_home>applications\portal-project\application-sync\webapps\`<yourportal>`, edit `<yourportal>.portal` file. Add the following reference to your skin in the <skin-pool> node:

```
<skin-pool>

     <skin-name>coologic</skin-name>

     <skin-name>futurism</skin-name>

     <skin-name>newskin</skin-name>

...

</skin-pool>
```

6. In the EBCC, open the portal, go to **`General > Skins`**. Notice that the new skin is visible in the Available skins list. Add the skin to the selected skin list. You can make it the default skin.

Note: These resources also exist within `<ebcc_home>\applications\portal\application-sync\webapps\<yourportal>`

When designing skins, note that CSS attributes take precedence over non-CSS attributes. For example, if the content heading color attribute is set within the JSP:

```
<span class="contentheading" color="green">Test</span>
```

But the contentheading style is set this way:

```
.contentheading{

color: blue;

}
```

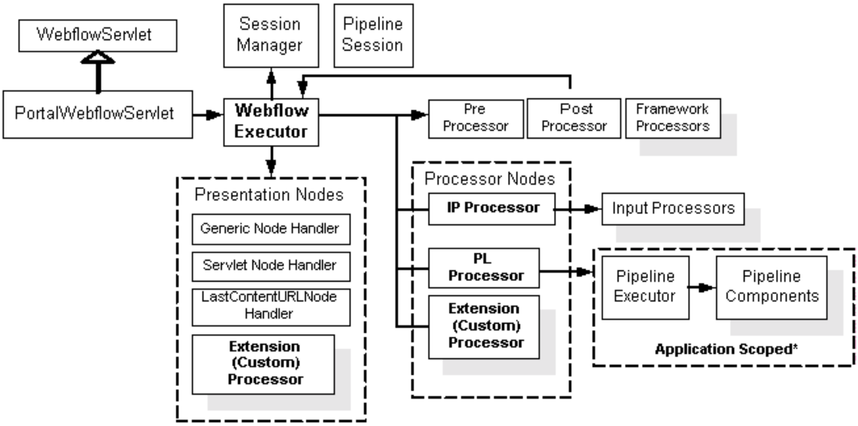
The text will be displayed blue instead of green.

Webflow and the Portal Platform

To customize the behavior of your portal Web applications, whether that means creating special navigation logic for a simple portlet, or developing complex business and presentation framework customizations that change the portal platform to suit your business case, an understanding of Webflow is required.

Figure 5-13 outlines the architecture of the portal platform, and gives insight into the way Webflow functionality is implemented.

Figure 5-13 Portal High Level Architecture



The Portal Framework as a Webflow Application

With a few exceptions, Webflow works the same way in portal web applications as it does within the rest of the platform. That is, you can use Webflow to coordinate different parts of your portal web application, enabling code reuse and maintainability.

Because its main function is to serve as a structure of reusable presentation and navigation modules, the Portal framework itself can be though of primarily as one large Webflow application. For this reason, the default Webflow underlying the sample portal included with WebLogic Portal, although it can be edited, should be thought of as a fundamental part of the Portal framework. As such, the portal.wf file included with the installation of the stockportal should not be edited. Unless you or your developers possess intimate knowledge of BEA Webflow functionality, the Portal Webflow should be reused as is. That said, a developer well versed in Webflow can customize the Webflow for a portal. For more information on Webflow, consult the *Guide to Managing Presentation and Business Logic: Using Webflow and Pipeline*.

For the most part, Webflow functionality is invoked the same way within the Portal framework as it is within the rest of the WebLogic Server platform, with the Webflow servlet (in this case the PortalWebflowServlet) performing most of the routing between validation, session logic and custom processor nodes. For an exhaustive discussion of Webflow, consult the *Guide to Managing Presentation and Business Logic: Using Webflow and Pipeline*. This section covers those aspects of Webflow functionality that are specific to the Portal framework.

Portal Presentation Logic and Webflow

Webflow and WebLogic Portal could also be thought of as two complementary navigation mechanisms. The portal platform provides an array of content placeholders, each able to function as a virtual browser that controls its own state without influencing the state of its neighbor portlets. The Webflow mechanism is designed to enable the navigation logic of a web application to be controlled by a multitude of factors, meaning that Personalization, Commerce and Campaign Services can all be coordinated to influence what is presented to the visitor at runtime. While portal Webflow is largely a part of the platform you are not expected to customize, each individual portlet can be assigned its own particular Webflow. The following section explains the basic requirements in using Webflows with portlets.

Note: For the most part, Webflow customization is to be performed using the Webflow Editor with the E-Business Control Center. This means you will not be editing the XML files in a text editor. This section exposes some fragments of these files for illustration only.

Portlets and Webflow

The Webflow associated with portals are generally not to be edited because it is fundamental to the application logic that makes up the portal platform. Individual portlets, however, can have individual Webflow files assigned to them, allowing practically all Webflow functionality to be applied to a portlet.

Portlet Webflow Parameters

Portal Webflow is always invoked with the following three parameters:

- **Portlet Webflow Namespace**: Portlet namespace handling allows the Webflow Servlet to discern individual portlets when handling events within a portal.

- **Event**: Portlet Webflows listen for as many as seven events related to user actions within a portlet.
- **Origin**: Unlike a typical web application using Webflow, the Portal server must keep track of the current node for each portlet Webflow being displayed. This current node information may be considered for each new event.

What Events Are Called on Portlet Webflow?

The out-of-the-box framework uses the following portlet events, all of which must be handled when Webflow is invoked:

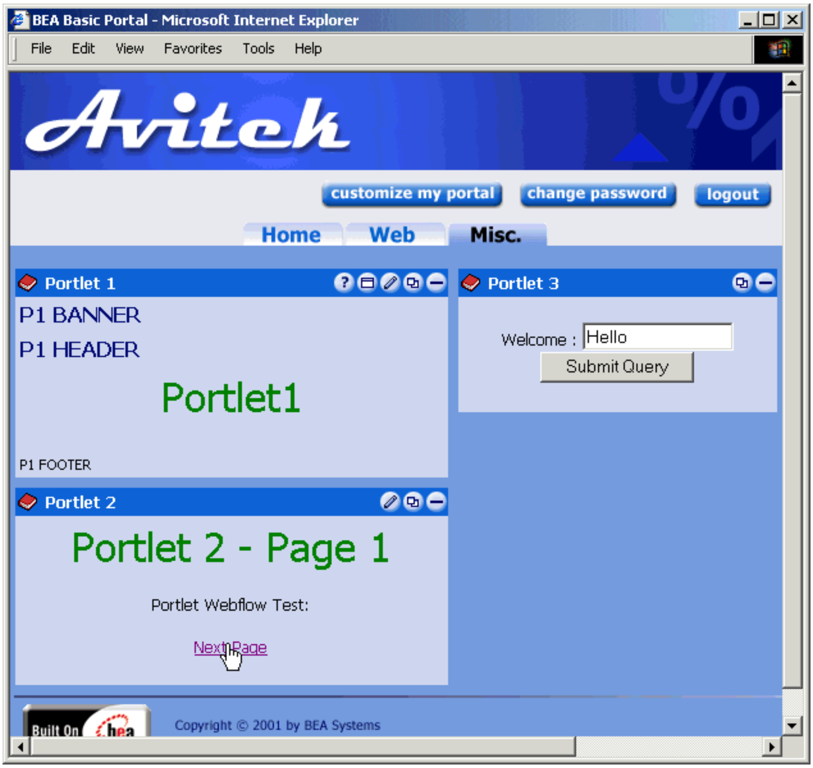
- Refresh – Required of all Webflows
- Maximize – Required if Maximizable and no Maximize URL specified in portlet definition
- unMaximize – Required if Maximizable and no Maximize URL specified in portlet definition
- Edit – Required if Editable and if no Edit URL specified in portlet definition
- unEdit – Required if Editable and if no Edit URL specified in portlet definition
- Minimize – Required if portlet is minimizable (No Minimize URL needed)
- unMinimize – Required if portlet is minimizable (No Minimize URL needed)

These events are called when the corresponding actions are performed, and handled by the out-of-the-box framework. You can use these events within your own portlets, and you can also define your own events. Your custom events can be triggered from within your custom portlets, or they can be triggered by portlet framework events.

The Refresh Event

Understanding Webflow in portals requires a grasp of the Refresh event. To illustrate the way this event is used, consider the following page view of the sample portal shown in Figure 5-14.

Figure 5-14 Three Portlets



P1 BANNER

P1 HEADER

Portlet1

P1 FOOTER

Welcome : Hello

Portlet 2 - Page 1

Portlet Webflow Test:

If a visitor minimized portlet2, a refresh event would be sent to both the other portlets via the PortalWebflowServlet, illustrated in Figure 5-13. It is this refresh event which would be used to make one portlet communicate with or respond to another portlet. (Remember that this refresh event is accompanied by the portlet namespace as well as the origin.)

The Origin Parameter

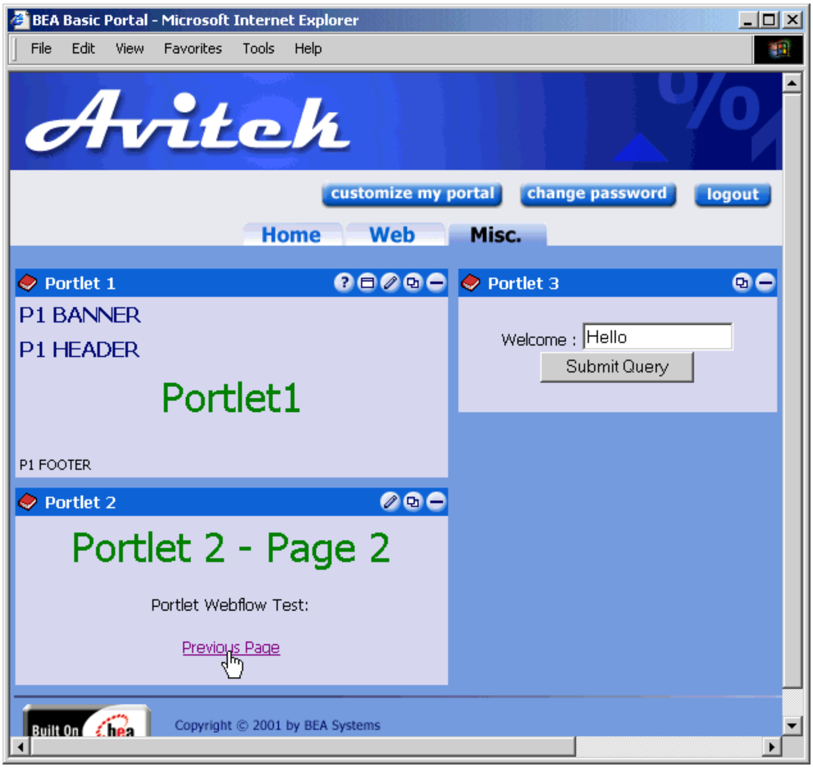
Portlet Webflows do not use Origin differently than other web applications. Consider what happens when a user clicks on the Next Page link in Portlet2 as shown in Figure 5-14. As you can see from looking at the Webflow fragment show in Figure 5-15, the presentation node defined as the origin is “portlet,” and the destination node declared for that origin is portletB.

Figure 5-15 Portlet2 Webflow Origin for Page 1

```
<presentation-origin node-name="portlet" node-type="jsp">
    <node-processor-info page-relative-path="/portlets/portlet2" page-name="portlet.jsp"/>
    <event-list>

        <!-- next -->
        <event event-name="switch">
            <destination namespace="portlet2"
                node-name="portletB"
                node-type="jsp"/>
        </event>

    </event-list>
</presentation-origin>
```

Sure enough, clicking on the “Next Page” link in page 1 of Portlet2 takes you to Page 2, as shown in Figure 5-16.

Figure 5-16 Portlet Origins and Webflow


BEA Basic Portal - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Avitek
customize my portal
change password
logout
Home
Web
Misc.
Portlet 1
Portlet 3
Portlet 2
Submit Query
Built On bea
Copyright © 2001 by BEA Systems

P1 BANNER

P1 HEADER

Portlet1

P1 FOOTER

Welcome : Hello

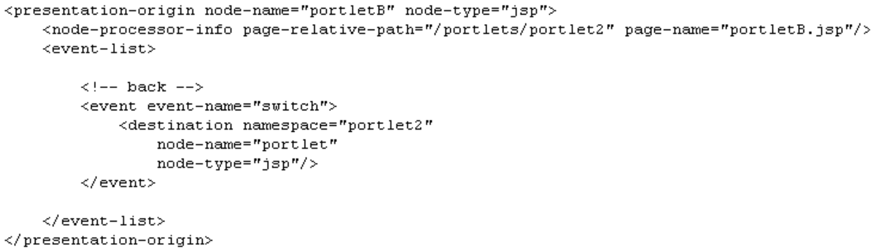
Portlet 2 - Page 2

Portlet Webflow Test:

Previous Page

To understand how the “Previous Page” link on Page2 is made to point to the initial presentation node designated for Portlet2, look at the Webflow fragment show in Figure 5-17.

Figure 5-17 Portlet2 Webflow Origin for Page 2

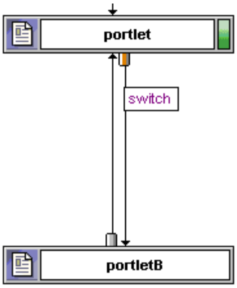


```
<presentation-origin node-name="portletB" node-type="jsp">
    <node-processor-info page-relative-path="/portlets/portlet2" page-name="portletB.jsp"/>
    <event-list>

        <!-- back -->
        <event event-name="switch">
            <destination namespace="portlet2"
                node-name="portlet"
                node-type="jsp"/>
        </event>

    </event-list>
</presentation-origin>
```

Considering this simple example illustrating the Origin parameter and the way Refresh events are handled in the portal framework, you can begin to see how much customization can be accomplished without writing any Java code. Keep in mind, however, that you will not need to edit the Webflow file directly - these fragments are shown for illustration only. The actual Webflow configuration is done using the Webflow Editor.

Figure 5-18 is an image from the Webflow editor, showing a very simple portlet Webflow which navigates from one JSP to another and back. You can see an example of a Webflow this simple in portlet 2 in the stock portal.

Figure 5-18 Simple Portlet Webflow Image



Using Webflow Within a Portlet

A few general reminders could save you some time:

- When using Webflow within a portlet, you need to use a portlet-specific JSP tag. Where a non-portal web application would invoke Webflow using `<wf:CreateWebflowURL/>` tag, you would use the following tag within a portlet: `<portlet:CreateWebflowURL/>`.
- To refer to the saved state of a portlet, use `LastContentURL`. to denote a special destination informs the portal engine to display the last/current JSP. For example: `<destination namespace="portal" node-name="lastContentUrl" node-type="portal"/>`
- Another special destination, "portalRefreshProcessor" is used to refresh the portal or return from a “regular” webflow back to the portal. For example: `<destination namespace="portal" node-name="portalRefreshProcessor" node-type="inputprocessor"/>`
- For all refresh events, use a wildcard event handler with last `LastContentURL`. This enables you to do back-end processing without changing the front-end state of the object.

When to Use Portlet Webflow

Webflow is not mandatory - it is an advanced feature of the WebLogic Portal platform. For more information on Webflow, consult the *Guide to Managing Presentation and Business Logic: Using Webflow and Pipeline*. Briefly, here are some reasons to use Webflow in individual portlets:

- To separate business logic from presentation
- To coordinate the navigation of a Web site with the execution of business logic
- To reuse existing InputProcessors and PipelineComponents
- To link business logic execution across portlets

What Portlet Webflows Can Specify

In order for them to use Webflows, portlets must observe the following contract: the Edit, Maximize and Content URLs described in the table above can either be declared within the portlet definition, or they can be handled within the Webflow for that individual portlet.

Note: By design, presentation nodes for a portlet must exist in the same namespace. Non-presentation nodes can exist in separate namespaces, but the portlet webflow must return to the originating namespace for all presentation nodes. This means that portlets may not have multiple namespaces.

CHAPTER

6 Portal Management JSP Tag Library Reference

WebLogic Portal includes a set of JSP tags designed to facilitate the development of portals and portlets. Use these predefined tags to reduce the amount of Java code required in your JSP page. This topic explains how to import each set of tags into your Web pages, and describes the purpose of each tag.

In the following tables, the Required column specifies if the attribute is required (yes) or optional (no). In the R/C column, C means that the attribute is a Compile time expression, and R means that the attribute can be either a Request time expression or a Compile time expression.

This topic includes the following sections:

- Importing Tag Libraries
- Portlet Tag Libraries
- Portal Tag Libraries
- Utility Tag Libraries

Importing Tag Libraries

To use these tags you must import them into each JSP page. Import the JSP tags described in this document by including the following code in your JSP page:

- `<%@ taglib uri="util.tld" prefix="util" %>`

  Imports the utility tag library.
- `<%@ taglib uri="portal.tld" prefix="ptl" %>`

  Imports the portal tag library.
- `<%@ taglib uri="portlet.tld" prefix="portlet" %>`

  Imports the portlet tag library.

Portlet Tag Libraries

The following section describes the portlet tags and their attributes.

<portlet:createWebflowURL>

The `<portlet:createWebflowURL>` tag is used in a portlet to dynamically create a Webflow URL in a JSP. The Webflow URL may include the protocol, domain name, port, Web application URI, `WebflowServlet` URI, and query string.

Table 6-1 <portlet:createWebflowURL>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the event is generated.	R
portletName	No	String	The relevant portlet ID. This is usually automatically generated, but can be modified.	R

Table 6-1 <portlet:createWebflowURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
event	Yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	C

Table 6-1 <portlet:createWebflowURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	C

Example

The following code sample illustrates how to use the `<portlet:createWebflowURL>` JSP tag:

```
<%@ taglib uri="portlet.tld" prefix="portlet" %>...

<center>
<font size="6" color="green">Portlet 2 - Page 1</font><BR>
<p>
<p>
Portlet Webflow Test:
<p>
<a href="<portlet:createWebflowURL event="switch"/>">Next Page</a>
<p>
</center>
```

<portlet:form>

The `<portlet:form>` tag is used in a JSP to dynamically generate an HTML form tag. This tag is not as sophisticated as the `<portlet:validatedForm>` tag, but is simpler. For more information about the `<portlet:validatedForm>` tag, refer to the next section.

Table 6-2 <portlet:form>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the event is generated.	R
portletName	No	String	The relevant portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
event	yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R

Table 6-2 <portlet:form> (Continued)

Tag Attribute	Required	Type	Description	R/C
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	C
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	C

Table 6-2 <portlet:form> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not except cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not except cookies.	C
hide	No	String	If set to `false`, the namespace, origin, and event will be displayed on the command line instead of as hidden form fields. Valid values are `true` and `false`. The default value is `true`.	C
method	No	String	The method to be used for the form. Valid values are `get` and `post`. The default value is `post`.	C
name	No	String	The name of the form.	C

<portlet:validatedForm>

The <`portlet:validatedForm`> tag is used to dynamically generate HTML forms that can be validated. When a Web site visitor enters invalid information, the visitor's input is preserved and redisplayed with an appropriate error message..

Table 6-3 <portlet:ValidatedForm>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the validated form URL is generated	R
portletName	No	String	The relevant portlet ID. This is usually automatically generated, but can be modified.	R

Table 6-3 <portlet:ValidatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
event	yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	C

Table 6-3 <portlet:ValidatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	C
hide	No	String	If set to `false`, the namespace, origin, and event will be displayed on the command line instead of as hidden form fields. Valid values are `true` and `false`. The default value is `true`.	C
method	No	String	The method to be used for the form. Valid values are `get` and `post`. The default value is `post`.	C
name	No	String	The name of the form.	C
validStyle	No	String	The style used to format the HTML field when it is valid.	R
invalidStyle	No	String	The style used to format the HTML field or the message when the field is invalid.	R
unspecifiedStyle	No	String	Used to specify the initial style of the HTML field before validation occurs.	R

Table 6-3 <portlet:ValidatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
styleId	No	String	Scripting variable that will be set to one of invalidStyle, unSpecifiedStyle, or validStyle, depending on the field's status: `valid`, `invalid`, `unspecified`. Can be used for finer control of formatting the HTML form.	R
applyStyle	No	String	Applies the associated style as indicated by the field status to the message, the field, or to none. Therefore, valid values are `message`, `field`, and `none`. The default value is `message`.	C
messageAlign	No	String	Indicates whether to align the error message above the field, to the right of the field, or below the field. Therefore, value values are `top`, `right`, and `bottom`. The default value is `right`.	C

Example

The following code sample illustrates how to use the `<portlet:validatedForm>` JSP tag:

```
<%@ taglib uri="portlet.tld" prefix="portlet" %>
<%@ taglib uri="webflow.tld" prefix="webflow" %>


<center>
<% String validStyle="color: black; font-family: Arial"; %>
<% String invalidStyle="color: darkred; font-style: Arial"; %>
<%-- If there was an InvalidFormDataException thrown display the
message --%>
<font size="3" color="darkred"><portlet:getException/></font>
<br>
<webflow:getProperty id="welcomeStr" property="WELCOME_STRING"
type="java.lang.String" scope="request" namespace="portlet3"/>
<% if (welcomeStr == null || welcomeStr.length() < 1)
   {
%>
     <portlet:validatedForm event="button.go" applyStyle="message"
messageAlign="right"
```

```
       validStyle="<%=validStyle%>"
invalidStyle="<%=invalidStyle%>"
unspecifiedStyle="<%=validStyle%>" >
          <p>
           Welcome :
          <webflow:text name="welcome" value="Hello" size="15"
maxlength="30" /> <br>
          <input type="submit"  name="Submit"/>
</portlet:validatedForm>
     <br>
<%
   }
   else
   {
%>
       <font size="5" color="green"><%= welcomeStr %></font>
<%
   }
%>
</center>
<p>
```

<portlet:createPortletEditURL>

The <portlet:createPortletEditURL> tag generates a webflow URL that represents editing a portlet.

Table 6-4 <portlet:createPortletEditURL>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the edit URL is generated.	R
portletName	No	String	The relevant portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R

Table 6-4 <portlet:createPortletEditURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-4 <portlet:createPortletEditURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portlet:createPortletEditURL>` JSP tag:

```
        }
        if (PortletRenderHelper.isEditable(request))
        {
%>

    <td width="1%" valign="middle"><a 
href="<portlet:createPortletEditURL httpsInd='http'/>">
      <img src="<webflow:createResourceURL 
resource='<%=imagesPath+"portlet_edit.gif"%>'/>" hspace="1" 
vspace="2" border="0" alt="Edit" align="absmiddle"></a></td>
<%
```

<portlet:createPortletUneditURL>

The `<createPortletUneditURL>` JSP Tag generates a webflow URL that represents leaving the edit URL and moving to another page.

Table 6-5 <portlet:createPortletUneditURL>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the unedit URL is generated.	R
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R

Table 6-5 <portlet:createPortletUneditURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

<portlet:createPortletMinimizeURL>

The `<portlet:createPortletMinimizeURL>` JSP Tag generates a webflow URL that represents minimizing a portlet.

Table 6-6 <portlet:createPortletMinimizeURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	`The page from which the minimize URL is generated.`	R

Table 6-6 <portlet:createPortletMinimizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R

Table 6-6 <portlet:createPortletMinimizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portlet:createPortletMinimizeURL>` JSP tag:

```
        }
        if (PortletRenderHelper.isMinimizable(request))
        {
%>

    <td width="1%" valign="middle"><a
href="<portlet:createPortletMinimizeURL httpsInd='http'/>">
      <img src="<webflow:createResourceURL
resource='<%=imagesPath+"portlet_min.gif"%>'/>" hspace="1"
vspace="2" border="0" alt="Minimize" align="absmiddle"></a></td>
<%
```

<portlet:createPortletUnminimizeURL>

The `<portlet:createPortletUnminimizeURL>` tag generates a webflow URL that represents unminimizing a portlet.

Table 6-7 <portlet:createPortletUnminimizeURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	The page from which the unminimize URL is generated.	R
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-7 <portlet:createPortletUnminimizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-7 <portlet:createPortletUnminimizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portlet:createPortletUnminimizeURL>` JSP tag:

```
%>
    <%--  Create a link to unminimize the portlet --%>
    <td width="1%" valign="middle"><a
href="<portlet:createPortletUnminimizeURL/>"><img
src="<webflow:createResourceURL
resource='<%=imagesPath+"portlet_unmin.gif"%>'/>" hspace="1"
vspace="2" border="0" alt="Restore" align="absmiddle"></a></td>
```

<portlet:createPortletMaximizeURL>

The `<portlet:createPortletMaximizeURL>` JSP Tag generates a webflow URL that represents maximizing a portlet.

Table 6-8 <portlet:createPortletMaximizeURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	`The page from which the maximize URL is generated.`	R
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-8 <portlet:createPortletMaximizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-8 <portlet:createPortletMaximizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portlet:createPortletMaximizeURL>` JSP tag:

```
<%
        }
        if (PortletRenderHelper.isMaximizable(request))
        {
%>

    <td width="1%" valign="middle">
      <%-- Create the link to maximize the portlet --%>
      <a href="<portlet:createPortletMaximizeURL
httpsInd='http'/>"><img src="<webflow:createResourceURL
resource='<%=imagesPath+"portlet_max.gif"%>'/>" hspace="1"
vspace="2" border="0" alt="Maximize" align="absmiddle"></a></td>
<%
```

<portlet:createPortletUnmaximizeURL>

The `<portlet:createPortletUnmaximizeURL>` JSP Tag generates a webflow URL that represents unmaximizing a portlet.

Table 6-9 <portlet:createPortletUnmaximizeURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	The page from which the unmaximize URL is generated.	R
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-9 <portlet:createPortletUnmaximizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application’s `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-9 <portlet:createPortletUnmaximizeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portal:createPortletUnamximizeURL>` JSP tag:

```
    <%--  Create a link to unmaximize the portlet --%>
    <td width="1%" valign="middle"><a
href="<portlet:createPortletUnmaximizeURL/>"><img
src="<webflow:createResourceURL
resource='<%=imagesPath+"portlet_unmax.gif"%>'/>" hspace="1"
vspace="2" border="0" alt="Unmaximize"></a></td>
```

<portlet:createPortletFloatURL>

The `<portlet:createPortletFloatURL>` tag generates a webflow URL that represents creating a ‘floating’ portlet, which is a portlet that appears in an independent window..

Table 6-10 <portlet:createPortletFloatURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	The page from which the float URL is generated.	R
portletName	No	String	The portlet ID. This is usually automatically generated, but can be modified.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-10 <portlet:createPortletFloatURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-10 <portlet:createPortletFloatURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portal:createPortletFloatURL>` JSP tag:

```
<%
        }
        if (PortletRenderHelper.isFloatable(request))
        {
            url = portletState.getUrl(Portlet.URL_CONTENT);
%>
            <util:validURL url="<%=url%>">
<% request.setAttribute( "fullscreenPortletURL", url); %>


    <td width="1%" valign="middle"><a
href="javascript:submitForm();"

onClick="openBrWindow('<portlet:createPortletFloatURL
httpsInd='http'/>',
                                       '<%= portletWindowName%>',
'scrollbars=yes,width=780,height=550'); return false;">
      <img src="<webflow:createResourceURL
resource='<%=imagesPath+"portlet_float.gif"%>'/>" hspace="1"
vspace="2" border="0" alt="Float" align="absmiddle"></a></td>
            </util:validURL>
<%
```

<portlet:getException>

The `<portlet:getException>` tag is used to retrieve the exception or message thrown by a webflow processor. This can be the message associated with a InvalidFormFieldException or ProcessingException. This tag can be inlined in which it calls getMessage() on the exception or return a scripting variable representing the exception itself.

Table 6-11 <portlet:getException>

Tag Attribute	Required	Type	Description	R/C
id	no	String	java scripting variable, can be used to retrieve an instance of the exception.	R

Table 6-11 <portlet:getException> (Continued)

Tag Attribute	Required	Type	Description	R/C
type	No	String	java class name, can be used to cast your exception.	R

Portal Tag Libraries

The following sections describe the portal tags and their attributes.

<portal:createWebflowURL>

The `<portal:createWebflowURL>` tag is used in a JSP to dynamically create a Webflow URL in a JSP. The Webflow URL may include the protocol, domain name, port, Web application URI, `WebflowServlet` URI, and query string.Example

Table 6-12 <portal:createWebflowURL>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	The page for which the Webflow URL is being generated.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
event	yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-12 <portal:createWebflowURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-12 <portal:createWebflowURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

<portal:form>

The `<portal:form>` tag is used in a JSP to dynamically generate an HTML form tag. This tag is not as sophisticated as the `<portal:validatedForm>` tag, but is simpler. For more information about the `<portal:validatedForm>` tag, refer to the next section.

Table 6-13 <portal:form>

Tag Attribute	Required	Type	Description	R/C
pageName	no	String	`The page for which the form is being generated.`	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R
event	yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R

Table 6-13 <portal:form> (Continued)

Tag Attribute	Required	Type	Description	R/C
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here.	R

Table 6-13 <portal:form> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not except cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not except cookies.	R
hide	No	String	If set to `false`, the namespace, origin, and event will be displayed on the command line instead of as hidden form fields. Valid values are `true` and `false`. The default value is `true`.	R
method	No	String	The method to be used for the form. Valid values are `get` and `post`. The default value is `post`.	R
name	No	String	The name of the form.	R

<portal:validatedForm>

The `<portal:validatedForm>` tag is used to dynamically generate HTML forms that can be validated. When a Web site visitor enters invalid information, the visitor's input is preserved and redisplayed with an appropriate error message.

Table 6-14 <portal:validatedForm>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page from which the validated form is being generated.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R

Table 6-14 <portal:validatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
event	yes	String	Webflow will use this in combination with the origin to resolve a destination in the supplied namespace XML file.	R
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R

Table 6-14 <portal:validatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here	R
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R
hide	No	String	If set to `false`, the namespace, origin, and event will be displayed on the command line instead of as hidden form fields. Valid values are `true` and `false`. The default value is `true`.	R
method	No		The method to be used for the form. Valid values are `get` and `post`. The default value is `post`.	R
name	No	String	The name of the form.	R
validStyle	No	String	The style used to format the HTML field when it is valid.	R
invalidStyle	No	String	The style used to format the HTML field or the message when the field is invalid.	R
unspecifiedStyle	No	String	Used to specify the initial style of the HTML field before validation occurs.	R

Table 6-14 <portal:validatedForm> (Continued)

Tag Attribute	Required	Type	Description	R/C
styleId	No	String	Scripting variable that will be set to one of invalidStyle, unSpecifiedStyle, or validStyle, depending on the field's status: `valid`, `invalid`, `unspecified`. Can be used for finer control of formatting the HTML form.	R
applyStyle	No	String	Applies the associated style as indicated by the field status to the message, the field, or to none. Therefore, valid values are `message`, `field`, and `none`. The default value is `message`.	R
messageAlign	No	String	Indicates whether to align the error message above the field, to the right of the field, or below the field. Therefore, value values are `top`, `right`, and `bottom`. The default value is `right`.	R

<portal:createPortalPageChangeURL>

The `<portal:createPortalPageChangeURL>` tag generates a webflow URL for a page change event.

Table 6-15 <portlet:createPortalPageChangeURL>

Tag Attribute	Required	Type	Description	R/C
pageName	No	String	The page for which the event will generated.	R
origin	No	String	The node where the event will be coming from. Origins follow the `node-name.node-type` format. This may or may not be equal to the page name. If omitted, then the JSP page name is used.	R

Table 6-15 <portlet:createPortalPageChangeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
domainName	No	String	Used to change the domain name or IP address of the URL. This may be used if Webflow is fronted by a proxy server and that server resides on another machine.	R
pathprefix	No	String	Used to prefix the path information. This string will be placed in front of the Web application URI. This can be used when the proxy server is located on the same machine. **Note:** The proxy must strip the path prefix before forwarding the request to Webflow.	R
pathsuffix	No	String	Used to suffix the path information. The additional path information will be placed after the `WebflowServlet` URI. This information can then be retrieved via the `request.getPathInfo()` method.	R
extraParams	No	String	Used to supply additional request parameters as name/value pairs.	R
doRedirect	No	String	Causes the `WebflowServlet` to perform a redirect instead of a forward to a presentation node. Valid values are `true` and `false`. The default is `false` (not to redirect).	R
httpsInd	No	String	Tells Webflow to calculate the protocol or use HTTPS or HTTP. Therefore, valid values are `calculate`, `http`, and `https`. The default value is `http`. `Calculate` will yield HTTPS if any node in the origin/event branch chain list is matched under the `HTTPS_URL_PATTERNS` context parameter in the application's `META-INF/web.xml` file. `Calculate` is more dynamic and expensive, but if the protocol needs to be forced you can specify it here	R

Table 6-15 <portlet:createPortalPageChangeURL> (Continued)

Tag Attribute	Required	Type	Description	R/C
encode	No	String	Tells Webflow to encode the URL. URLs need to be encoded if you wish to maintain session state and the browser does not accept cookies. Valid values are `true` and `false`. The default value is `true`, as URLs will only need to be encoded if the browser does not accept cookies.	R

Example

The following code sample illustrates how to use the `<portal:createPortalPageChangeURL>` JSP tag:

```
     if (DISPLAY_TEXT_LINKS)
              {
                    if (portalPageName.equals(selectedPage))

              {
%>
<td class="tabselected" nowrap align="center">
<b>&nbsp;
<a href="<ptl:createPortalPageChangeURL pageName='<%=
portalPageName %>'/>"><span
class="tabselected"><%=portalPageName%></span></a></b></td>
                <td class="tabselected"
WIDTH="2"><b>&nbsp;</b></td>
<%                          }
              else
                          {
%>                          <td class="tabunselected" nowrap
align="center">

<b>&nbsp;
<a href="<ptl:createPortalPageChangeURL    pageName='<%=
portalPageName %>'/>"><span
class="tabunselected"><%=portalPageName%></span></a></b></td>
                <td class="tabunselected"
WIDTH="2"><b>&nbsp;</b></td>
<%            }
```

Utility Tag Libraries

The following section describes the utility tags and their attributes.

<utility:validURL>

Processes the body if the supplied url is not null and greater then 3 characters long.

Table 6-16 <utility:validURL>

Tag Attribute	Required	Type	Description	R/C
URL	yes	String	The supplied URL that is processed if it is valid.	R

<utility:invalidURL>

The `<utility:invalidURL>` tag processes the body if the supplied url is null or less then 4 characters long.

Table 6-17 <utility:invalidURL>

Tag Attribute	Required	Type	Description	R/C
URL	yes	String	The supplied URL.	R

Example

The following code sample illustrates how to use the `<utility:invalidURL>` as well as the <`utility:validURL`> JSP tag:

```
<%

       // First try for an alternate header if one doesn't exist use
the

       // regular header

       url =  portletState.getUrl(Portlet.URL_ALTERNATE_HEADER);

       debug.out("Alternate header: " + url);

%>

       <util:invalidURL url="<%=url %>">

       <% url = portletState.getUrl(Portlet.URL_HEADER); %>

       </util:invalidURL>

       <util:validURL url="<%=url %>">

       <table cellpadding="4" cellspacing="0" width="100%"
class="portletheader">

       <tr>

       <td width="100%">

       <jsp:include page="<%=url%>"/>

        </td>

        </tr>

        </table>

       </util:validURL>

<%
```

CHAPTER

7 Portal Management Database Schema

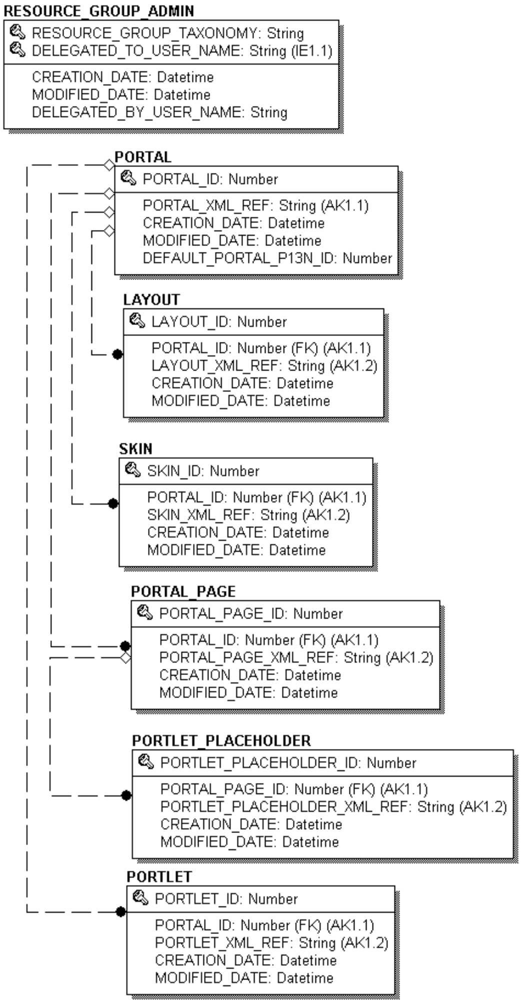
This topic documents the database schema for the WebLogic Portal package. This topic includes the following sections:

- The Entity-Relation Diagram
- List of Tables Comprising the Portal Management Package
- The Portal Management Data Dictionary
- The SQL Scripts Used to Create the Database
- Defined Constraints

The Entity-Relation Diagram

Figure 7-1 shows the logical Entity-Relation diagram for the WebLogic Commerce Server Portal and Portlet tables in the Commerce database. See the subsequent sections in this chapter for information about the data type syntax.

Figure 7-1 Entity-Relation Diagram for the Portal and Portlet Tables



RESOURCE_GROUP_ADMIN
RESOURCE_GROUP_TAXONOMY: String
DELEGATED_TO_USER_NAME: String (IE1.1)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
DELEGATED_BY_USER_NAME: String
PORTAL
PORTAL_ID: Number
PORTAL_XML_REF: String (AK1.1)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
DEFAULT_PORTAL_P13N_ID: Number
LAYOUT
LAYOUT_ID: Number
PORTAL_ID: Number (FK) (AK1.1)
LAYOUT_XML_REF: String (AK1.2)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
SKIN
SKIN_ID: Number
PORTAL_ID: Number (FK) (AK1.1)
SKIN_XML_REF: String (AK1.2)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
PORTAL_PAGE
PORTAL_PAGE_ID: Number
PORTAL_ID: Number (FK) (AK1.1)
PORTAL_PAGE_XML_REF: String (AK1.2)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
PORTLET_PLACEHOLDER
PORTLET_PLACEHOLDER_ID: Number
PORTAL_PAGE_ID: Number (FK) (AK1.1)
PORTLET_PLACEHOLDER_XML_REF: String (AK1.2)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime
PORTLET
PORTLET_ID: Number
PORTAL_ID: Number (FK) (AK1.1)
PORTLET_XML_REF: String (AK1.2)
CREATION_DATE: Datetime
MODIFIED_DATE: Datetime

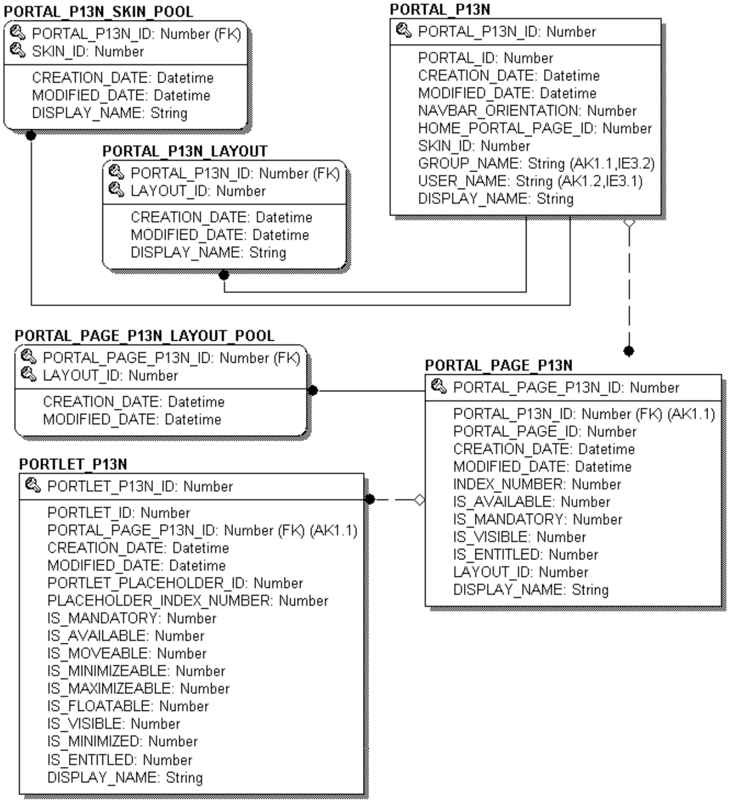
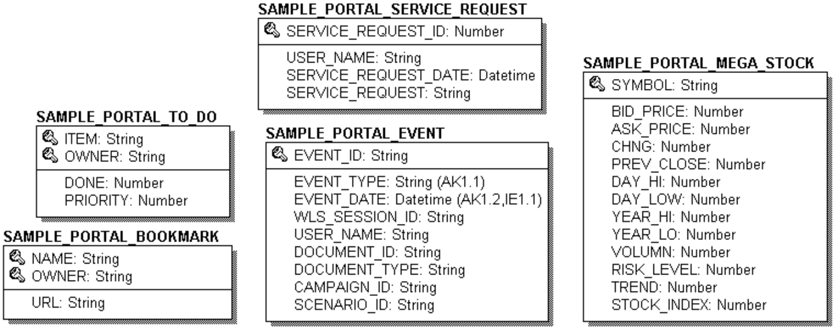


Figure 7-2 lists the tables used for the example portal.

Figure 7-2 Tables Used for the Sample Portal



List of Tables Comprising the Portal Management Package

Portal Management tables:

The LAYOUT Database Table
The PORTAL Database Table
The PORTAL_P13N Database Table
The PORTAL_P13N_LAYOUT Database Table
The PORTAL_P13N_SKIN_POOL Database Table
The PORTAL_PAGE Database Table
The PORTAL_PAGE_P13N Database Table
The PORTAL_PAGE_P13N_LAYOUT_POOL Database Table
The PORTLET Database Table
The PORTLET_P13N Database Table
The PORTLET_PLACEHOLDER Database Table
The RESOURCE_GROUP_ADMIN Database Table
The SKIN Database Table

The Portal Management Data Dictionary

In this section, the Portal, Portlet, and Sample Portal schema tables are arranged alphabetically as a data dictionary.

The LAYOUT Database Table

Table 7-1 describes the LAYOUT table. This table stores all of the defined layouts.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is LAYOUT_ID.

Table 7-1 LAYOUT Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
LAYOUT_ID	NUMBER(15)	NOT NULL	PK— a unique, system-generated number used as the record identifier.
PORTAL_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with the layout. FK to the PORTAL table.
LAYOUT_XML_REF	VARCHAR(254)	NOT NULL	The identifier for this layout as specified in the portal definition XML file.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.

The PORTAL Database Table

Table 7-2 describes the PORTAL table. This table stores all of the defined portals.

The Primary Key is `PORTAL_ID`.

Table 7-2 PORTAL Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.
PORTAL_XML_REF	VARCHAR(254)	NOT NULL	The identifier for this portal as specified in the portal definition XML file.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
DEFAULT_PORTAL_P13N_ID	NUMBER(15)	NULL	The record identifier for the personalized portal associated with the particular portal.

The PORTAL_P13N Database Table

Table 7-3 describes the PORTAL_P13N table. This table stores all of the personalized portal records.

The Primary Key is `PORTAL_P13N_ID`.

Table 7-3 PORTAL_P13N Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_P13N_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.
PORTAL_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with the personalized portal. FK to the `PORTAL` table.

Table 7-3 PORTAL_P13N Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
NAVBAR_ORIENTATION	NUMBER(1)	NOT NULL	This flag determines the orientation of the navigation bar. 0 equates to `horizontal` and 1 equates to `vertical`.
HOME_PORTAL_PAGE_ID	NUMBER(15)	NOT NULL	The record identifier for the home portal page associated with the personalized portal. FK to the `PORTAL_PAGE` table.
SKIN_ID	NUMBER(15)	NOT NULL	The record identifier for the skin associated with the personalized portal. FK to the `SKIN` table.
GROUP_NAME	VARCHAR(200)	NOT NULL	The group name associated with the personalized portal.
USER_NAME	VARCHAR(200)	NOT NULL	The user name associated with the personalized portal.
DISPLAY_NAME	VARCHAR(254)	NOT NULL	The personalized portal display name.

The PORTAL_P13N_LAYOUT Database Table

Table 7-4 describes the PORTAL_P13N_LAYOUT table. This table stores all available layouts for each of the personalized portals.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is comprised of both `LAYOUT_ID` and `PORTAL_P13N_ID`.

Table 7-4 PORTAL_P13N_LAYOUT Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
LAYOUT_ID	NUMBER(15)	NOT NULL	PK—combined with PORTAL_P13N_ID to form a unique record identifier. FK to the LAYOUT table.
PORTAL_P13N_ID	NUMBER(15)	NOT NULL	PK—combined with PORTAL_P13N_ID to form a unique record identifier. FK to the PORTAL_P13N table.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
DISPLAY_NAME	VARCHAR(254)	NULL	The display name for the layout associated with the personalized portal.

The PORTAL_P13N_SKIN_POOL Database Table

Table 7-5 describes the PORTAL_P13N_SKIN_POOL table. This table stores all available skins for a personalized portal.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is comprised of both `PORTAL_P13N_ID` and `SKIN_ID`.

Table 7-5 PORTAL_P13N_SKIN_POOL Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_P13N_ID	NUMBER(15)	NOT NULL	PK—combined with `SKIN_ID` to form a unique record identifier. FK to the `PORTAL_P13N` table.
SKIN_ID	NUMBER(15)	NOT NULL	PK—combined with `PORTAL_P13N_ID` to form a unique record identifier. FK to the `SKIN` table.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
DISPLAY_NAME	VARCHAR(254)	NULL	The display name for the skin associated with the personalized portal.

The PORTAL_PAGE Database Table

Table 7-6 describes the PORTAL_PAGE table. This table stores all of the defined pages for each portal.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is `PORTAL_PAGE_ID`.

Table 7-6 PORTAL_PAGE Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_PAGE_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.

Table 7-6 PORTAL_PAGE Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with the portal page. FK to the PORTAL table.
PORTAL_PAGE_XML_REF	VARCHAR(254)	NOT NULL	The identifier for this portal page as specified in the portal definition XML file.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.

The PORTAL_PAGE_P13N Database Table

Table 7-7 describes the PORTAL_PAGE_P13N table. This table stores information for each personalized portal page.

For information on defined constraints in this table, see “Defined Constraints” on page 7-23.

The Primary Key is PORTAL_PAGE_P13N_ID.

Table 7-7 PORTAL_PAGE_P13N Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTAL_PAGE_P13N_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.
PORTAL_P13N_ID	NUMBER(15)	NOT NULL	The record identifier for the personalized portal associated with this personalized portal page. FK to the PORTAL_P13N table.
PORTAL_PAGE_ID	NUMBER(15)	NOT NULL	The record identifier for the portal page associated with this personalized portal page. FK to the PORTAL_PAGE table.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.

Table 7-7 PORTAL_PAGE_P13N Table Metadata (Continued)

Column Name	Data Type	Null Value	Description and Recommendations
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
INDEX_NUMBER	NUMBER(4)	NOT NULL	The index number associated with this piece of the personalized portal page.
IS_AVAILABLE	NUMBER(1)	NOT NULL	This flag determines whether or not the portal page is available. 0 equates to `false`, 1 equates to `true`, and -1 equates to `unspecified`.
IS_ENTITLED	NUMBER(1)	NOT NULL	This flag determines whether or not entitlement rulesets have been defined for the portal page. 0 equates to `false`, 1 equates to `true`, and -1 equates to `unspecified`.
IS_MANDATORY	NUMBER(1)	NOT NULL	This flag determines whether or not the portal page is mandatory. 0 equates to `false`, 1 equates to `true`, and -1 equates to `unspecified`.
IS_VISIBLE	NUMBER(1)	NOT NULL	This flag determines whether or not the portal page is visible. 0 equates to `false`, 1 equates to `true`, and -1 equates to `unspecified`.
LAYOUT_ID	NUMBER(15)	NOT NULL	The record identifier for the layout associated with this personalized portal page. FK to the `LAYOUT` table.
DISPLAY_NAME	VARCHAR(254)	NOT NULL	The display name associated with the personalized portal page.

The PORTAL_PAGE_P13N_LAYOUT_POOL Database Table

Table 7-8 describes the PORTAL_PAGE_P13N_LAYOUT_POOL table. This table stores all available layouts for each of the personalized portal pages.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is comprised of both `PORTAL_PAGE_P13N_ID` and `LAYOUT_ID`.

Table 7-8 PORTAL_PAGE_P13N_LAYOUT_POOL Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
`PORTAL_PAGE_P13N_ID`	`NUMBER(15)`	`NOT NULL`	PK—combined with `LAYOUT_ID` this is a unique record identifier. FK to the PORTAL_PAGE_P13N table.
`LAYOUT_ID`	`NUMBER(15)`	`NOT NULL`	PK—combined with `PORTAL_PAGE_P13N_ID` this is a unique record identifier. FK to the `LAYOUT` table.
`CREATION_DATE`	`DATE`	`NOT NULL`	The date and time the record was created.
`MODIFIED_DATE`	`DATE`	`NOT NULL`	The last date and time the record was modified.

The PORTLET Database Table

Table 7-9 describes the PORTLET table. This table stores all of the defined portlets.

For information on defined constraints in this table, see “Defined Constraints” on page 7-23.

The Primary Key is `PORTLET_ID`.

Table 7-9 PORTLET Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTLET_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.
PORTAL_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with this particular portlet. FK to the `PORTAL` table.
PORTLET_XML_REF	VARCHAR(254)	NOT NULL	The identifier for this portlet as specified in the portal definition XML file.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.

The PORTLET_P13N Database Table

Table 7-10 describes the PORTLET_P13N table. This table holds all personalized portlet records.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is `PORTLET_P13N_ID`.

Table 7-10 PORTLET_P13N Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTLET_P13N_ID	NUMBER(15)	NOT NULL	PK—a unique, system-generated number used as the record identifier.
PORTLET_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with this particular personalized portlet. FK to the `PORTLET` table.
PORTAL_PAGE_P13N_ID	NUMBER(15)	NOT NULL	The record identifier for the portal associated with this particular personalized portlet. FK to the `PORTAL_PAGE_P13N` table.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
PORTLET_PLACEHOLDER_ID	NUMBER(15)	NOT NULL	The record identifier for the placeholder selected for this particular personalized portlet. FK to the `PORTLET_PLACEHOLDER` table.
PLACEHOLDER_INDEX_NUMBER	NUMBER(4)	NOT NULL	An index specifying the order that this particular personalized portlet appears within the specified placeholder.

Table 7-10 PORTLET_P13N Table Metadata (Continued)

Column Name	Data Type	Null Value	Description and Recommendations
IS_MANDATORY	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet is mandatory. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_AVAILABLE	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet is available. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_MOVEABLE	NUMBER(1)	NOT NULL	*This column is not being used and is reserved for future use.*
IS_MINIMIZEABLE	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet can be minimized. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_MAXIMIZEABLE	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet can be maximized. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_FLOATABLE	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet can float. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_VISIBLE	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet is visible. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.
IS_MINIMIZED	NUMBER(1)	NOT NULL	This flag determines whether or not the personalized portlet is minimized at startup. 0 equates to `false,` 1 equates to `true`, and -1 equates to `unspecified`.

Table 7-10 PORTLET_P13N Table Metadata (Continued)

Column Name	Data Type	Null Value	Description and Recommendations
IS_ENTITLED	NUMBER(1)	NOT NULL	This flag determines whether or not entitlement rulesets have been defined for the portlet. 0 equates to `false`, 1 equates to `true`, and -1 equates to `unspecified`.
DISPLAY_NAME	VARCHAR(254)	NOT NULL	The display name of the personalized portlet.

The PORTLET_PLACEHOLDER Database Table

Table 7-11 describes the PORTLET_PLACEHOLDER table.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is `PORTLET_PLACEHOLDER_ID`.

Table 7-11 PORTLET_PLACEHOLDER_ID Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
PORTLET_PLACEHOLDER_ID	NUMBER(15)	NOT NULL	PK—stores all of the defined placeholder records for each portlet.
PORTAL_PAGE_ID	NUMBER(15)	NOT NULL	The record identifier for the portal page associated with this particular portlet placeholder. FK to the `PORTAL_PAGE` table.
PORTLET_PLACEHOLDER_XML_REF	VARCHAR(254)	NOT NULL	The identifier for this placeholder as specified in the portal definition XML file.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.

Table 7-11 PORTLET_PLACEHOLDER_ID Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.

The RESOURCE_GROUP_ADMIN Database Table

Table 7-12 describes the RESOURCE_GROUP_ADMIN table. This table is used to store administrative privileges for various users.

The Primary Key is comprised of both RESOURCE_GROUP_TAXONOMY and DELEGATED_TO_USER_NAME.

Table 7-12 RESOURCE_GROUP_ADMIN Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
RESOURCE_GROUP_TAXONOMY	VARCHAR(254)	NOT NULL	PK—combined with DELEGATED_TO_USER_NAME this is used as a unique record identifier.
DELEGATED_TO_USER_NAME	VARCHAR(200)	NOT NULL	PK—combined with RESOURCE_GROUP_TAXONOMY this is a unique record identifier.
CREATION_DATE	DATE	NOT NULL	The date and time the record was created.
MODIFIED_DATE	DATE	NOT NULL	The last date and time the record was modified.
DELEGATED_BY_USER_NAME	VARCHAR(200)	NULL	The user who administered priviliges to the DELEGATED_TO_USER_NAME.

The SKIN Database Table

Table 7-13 describes the SKIN table. This table holds all of the defined skins.

For information on defined constraints in this table, see "Defined Constraints" on page 7-23.

The Primary Key is comprised of `SKIN_ID`.

Table 7-13 SKIN Table Metadata

Column Name	Data Type	Null Value	Description and Recommendations
`SKIN_ID`	`NUMBER(15)`	`NOT NULL`	PK—a unique, system-generated number used as the record identifier.
`PORTAL_ID`	`NUMBER(15)`	`NOT NULL`	The record identifier for the portal associated with this particular skin. FK to the `PORTAL` table.
`SKIN_XML_REF`	`VARCHAR(254)`	`NOT NULL`	The identifier for this skin as specified in the portal definition XML file.
`CREATION_DATE`	`DATE`	`NOT NULL`	The date and time the record was created.
`MODIFIED_DATE`	`DATE`	`NOT NULL`	The last date and time the record was modified.

The SQL Scripts Used to Create the Database

The database schemas for WebLogic Portal and WebLogic Personalization Server are all created by executing the `create_all` script for the target database environment.

Scripts

Regardless of your database, execute one of the following to generate the necessary database objects for the modules desired (WebLogic Portal, WebLogic Personalization Server, Commerce services, Campaign services and Sample Portal):

- `PORTAL_HOME\db\create_all.bat` (Windows)
- `PORTAL_HOME/db/create_all.sh` (UNIX)

The following are the various directories underneath `WL_COMMERCE_HOME/db` (as seen in a UNIX environment):

```
PORTAL_HOME/db/cloudscape/351

PORTAL_HOME/db/oracle/817
```

Note: In this documentation,`PORTAL_HOME` is used to designate the directory where the product is installed.

Each of the databases supported have the same number of scripts in each of their sub-directories. The scripts are listed and described in Table 7-14 below.

Table 7-14 The Scripts Supporting the Databases

Script Name	Description
`create_all.bat`	Windows script used to connect to the database and create the necessary database objects for the modules desired (e.g., WebLogic Portal, WebLogic Personalization Server, Commerce services, Campaign services and Sample Portal)

Table 7-14 The Scripts Supporting the Databases (Continued)

Script Name	Description
`create_all.sh`	Unix script used to connect to the database and create the necessary database objects for the modules desired (e.g., WebLogic Portal, WebLogic Personalization Server, Commerce services, Campaign services and Sample Portal)
`campaign_create_fkeys.sql`	SQL script used to create all foreign keys associated with the Campaign services.
`campaign_create_indexes.sql`	SQL script used to create all indexes associated with the Campaign services.
`campaign_create_tables.sql`	SQL script used to create all tables associated with the Campaign services.
`campaign_create_triggers.sql`	SQL script used to create all database triggers associated with the Campaign services.
`campaign_create_views.sql`	SQL script used to create all views associated with the Campaign services.
`campaign_drop_constraints.sql`	SQL script used to drop all constraints (other than foreign keys) associated with the Campaign services.
`campaign_drop_fkeys.sql`	SQL script used to drop all foreign key constraints associated with the Campaign services.
`campaign_drop_indexes.sql`	SQL script used to drop all indexes associated with the Campaign services.
`campaign_drop_tables.sql`	SQL script used to drop all tables associated with the Campaign services.
`campaign_drop_views.sql`	SQL script used to drop all views associated with the Campaign services.
`p13n_create_fkeys.sql`	SQL script used to create all foreign keys associated with the WebLogic Personalization Server.
`p13n_create_indexes.sql`	SQL script used to create all indexes associated with the WebLogic Personalization Server.
`p13n_create_tables.sql`	SQL script used to create all tables associated with the WebLogic Personalization Server.
`p13n_create_triggers.sql`	SQL script used to create all database triggers associated with the WebLogic Personalization Server.

Table 7-14 The Scripts Supporting the Databases (Continued)

Script Name	Description
`p13n_create_views.sql`	SQL script used to create all views associated with the WebLogic Personalization Server.
`p13n_drop_constraints.sql`	SQL script used to drop all constraints (other than foreign keys) associated with the WebLogic Personalization Server.
`p13n_drop_fkeys.sql`	SQL script used to drop all foreign key constraints associated with the WebLogic Personalization Server.
`p13n_drop_indexes.sql`	SQL script used to drop all indexes associated with the WebLogic Personalization Server.
`p13n_drop_tables.sql`	SQL script used to drop all tables associated with the WebLogic Personalization Server.
`p13n_drop_views.sql`	SQL script used to drop all views associated with the WebLogic Personalization Server.
`portal_create_fkeys.sql`	SQL script used to create all foreign keys associated with the WebLogic Portal.
`portal_create_indexes.sql`	SQL script used to create all indexes associated with the WebLogic Portal.
`portal_create_tables.sql`	SQL script used to create all tables associated with the WebLogic Portal.
`portal_create_triggers.sql`	SQL script used to create all database triggers associated with the WebLogic Portal.
`portal_create_views.sql`	SQL script used to create all views associated with the WebLogic Portal.
`portal_drop_constraints.sql`	SQL script used to drop all constraints (other than foreign keys) associated with the WebLogic Portal.
`portal_drop_fkeys.sql`	SQL script used to drop all foreign key constraints associated with the WebLogic Portal.
`portal_drop_indexes.sql`	SQL script used to drop all indexes associated with the WebLogic Portal.
`portal_drop_tables.sql`	SQL script used to drop all tables associated with the WebLogic Portal.
`portal_drop_views.sql`	SQL script used to drop all views associated with the WebLogic Portal.

Table 7-14 The Scripts Supporting the Databases (Continued)

Script Name	Description
sample_portal_create_fkeys.sql	SQL script used to create all foreign keys associated with the Sample Portal.
sample_portal_create_indexes.sql	SQL script used to create all indexes associated with the Sample Portal.
sample_portal_create_tables.sql	SQL script used to create all tables associated with the Sample Portal.
sample_portal_create_triggers.sql	SQL script used to create all database triggers associated with the Sample Portal.
sample_portal_create_views.sql	SQL script used to create all views associated with the Sample Portal.
sample_portal_drop_constraints.sql	SQL script used to drop all constraints (other than foreign keys) associated with the Sample Portal.
sample_portal_drop_fkeys.sql	SQL script used to drop all foreign key constraints associated with the Sample Portal.
sample_portal_drop_indexes.sql	SQL script used to drop all indexes associated with the Sample Portal.
sample_portal_drop_tables.sql	SQL script used to drop all tables associated with the Sample Portal.
sample_portal_drop_views.sql	SQL script used to drop all views associated with the Sample Portal.
wlcs_create_fkeys.sql	SQL script used to create all foreign keys associated with the Commerce services.
wlcs_create_indexes.sql	SQL script used to create all indexes associated with the Commerce services.
wlcs_create_tables.sql	SQL script used to create all tables associated with the Commerce services.
wlcs_create_triggers.sql	SQL script used to create all database triggers associated with the Commerce services.
wlcs_create_views.sql	SQL script used to create all views associated with the Commerce services.
wlcs_drop_constraints.sql	SQL script used to drop all constraints (other than foreign keys) associated with the Commerce services.

Table 7-14 The Scripts Supporting the Databases (Continued)

Script Name	Description
`wlcs_drop_fkeys.sql`	SQL script used to drop all foreign key constraints associated with the Commerce services.
`wlcs_drop_indexes.sql`	SQL script used to drop all indexes associated with the Commerce services.
`wlcs_drop_tables.sql`	SQL script used to drop all tables associated with the Commerce services.
`wlcs_drop_views.sql`	SQL script used to drop all views associated with the Commerce services.

Defined Constraints

Various constraints are defined and used in the Portal database schema. These constraints can be found in the following scripts:

`portal_create_fkeys.sql`—contains the Foreign Keys

`portal_create_tables.sql`—contains the Check Constraints

Table 7-15 Constraints Defined on Portal Database Tables

Table Name	Constraints
`LAYOUT`	**Column**—PORTAL_ID **Constraint**—FK1_LAYOUT **Constraint Type**—FOREIGN KEY Ensures that each `LAYOUT` references an existing `PORTAL` via the `PORTAL_ID` column.
`PORTAL_P13N_SKIN_POOL`	**Column**—PORTAL_P13N_ID **Constraint—**FK1_SKIN_POOL **Constraint Type—**FOREIGN KEY Ensures that each `PORTAL_P13N_SKIN_POOL` references an existing `PORTAL_P13N` via the `PORTAL_P13N_ID` column.

Table 7-15 Constraints Defined on Portal Database Tables (Continued)

Table Name	Constraints
PORTAL_PAGE	**Column**—PORTAL_ID **Constraint—**FK1_PORTAL_PAGE **Constraint Type—**FOREIGN KEY Ensures that each PORTAL_PAGE references an existing PORTAL via the PORTAL_ID column
PORTAL_PAGE_P13N	**Column**— PORTAL_P13N_ID **Constraint—**FK3_PL_PAGE_P13N **Constraint Type—**FOREIGN KEY Ensures that each PORTAL_PAGE_P13N references an existing PORTAL_P13N via the PORTAL_P13N_ID column.
PORTAL_PAGE_P13N_LAYOUT_POOL	**Column**—PORTAL_PAGE_P13N_ID **Constraint—** FK2_LAYOUT_POOL **Constraint Type—**FOREIGN KEY Ensures that each PORTAL_PAGE_P13N_LAYOUT_POOL references an existing PORTAL_PAGE_P13N via the PORTAL_PAGE_P13N_ID column.
PORTLET	**Column**—PORTAL_ID **Constraint—**FK1_PORTLET **Constraint Type—**FOREIGN KEY Ensures that each PORTLET references an existing PORTAL via the PORTAL_ID column.
PORTLET_P13N	**Column**—PORTAL_PAGE_P13N_ID **Constraint—**FK3_PORTLET_P13N **Constraint Type—**FOREIGN KEY Ensures that each PORTAL_P13N references an existing PORTAL_PAGE_P13N via the PORTAL_PAGE_P13N_ID column.
PORTLET_PLACEHOLDER	**Column**— PORTAL_PAGE_ID **Constraint—**FK1_PORTLET_PHD **Constraint Type—**FOREIGN KEY Ensures that each PORTLET_PLACEHOLDER references an existing PORTAL_PAGE via the PORTAL_PAGE_ID column.

Table 7-15 Constraints Defined on Portal Database Tables (Continued)

Table Name	Constraints
SKIN	**Column**—PORTAL_ID **Constraint—** FK1_SKIN **Constraint Type—**FOREIGN KEY Ensures that each SKIN references an existing PORTAL via the PORTAL_ID column.
PORTAL_P13N_LAYOUT	**Column**—PORTAL_P13N_ID **Constraint—**FK1_PL_LAYOUT **Constraint Type—**FOREIGN KEY Ensures that each PORTAL_P13N_LAYOUT references an existing PORTAL_P13N via the PORTAL_P13N_ID column.
PORTAL_PAGE_P13N	**Column**—IS_AVAILABLE **Constraint—**CC1_PL_PAGE_P13N **Constraint Type—**CHECK Ensures the value of the IS_AVAILABLE column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_MANDATORY **Constraint—**CC2_PL_PAGE_P13N **Constraint Type—**CHECK Ensures the value of the IS_MANDATORY column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_VISIBLE **Constraint—**CC3_PL_PAGE_P13N **Constraint Type—**CHECK Ensures the value of the IS_VISIBLE column is either -1 (undefined), 0 (false) or (true). **Column**—IS_ENTITLED **Constraint—**CC4_PL_PAGE_P13N **Constraint Type—**CHECK Ensures the value of the IS_ENTITLED column is either -1 (undefined), 0 (false) or 1 (true).

Table 7-15 Constraints Defined on Portal Database Tables (Continued)

Table Name	Constraints
PORTLET_P13N	**Column**—IS_MINIMIZED **Constraint—**CC1_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_MINIMIZED column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_MANDATORY **Constraint**—CC2_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_MANDATORY column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_AVAILABLE **Constraint—**CC3_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_AVAILABLE column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_MOVEABLE **Constraint—**CC4_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_MOVEABLE column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_MINIMIZEABLE **Constraint—**CC5_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_MINIMIZEABLE column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_MAXIMIZEABLE **Constraint—**CC6_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_MAXIMIZEABLE column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_FLOATABLE **Constraint—**CC7_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the IS_FLOATABLE column is either -1 (undefined), 0 (false) or 1 (true).

Table 7-15 Constraints Defined on Portal Database Tables (Continued)

Table Name	Constraints
`PORTLET_P13N` (continued)	**Column**—IS_VISIBLE **Constraint—**CC8_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the `IS_VISIBLE` column is either -1 (undefined), 0 (false) or 1 (true). **Column**—IS_ENTITLED **Constraint—**CC9_PORTLET_P13N **Constraint Type—**CHECK Ensures the value of the `IS_ENTITLED` column is either -1 (undefined), 0 (false) or 1 (true).

Index